

# **Projektisuunnitelma**

**OPEAPURI – Opetuutorin apuväline**

**Ohjelmistotuotantoprojekti**  
**Helsinki**  
**HELSINGIN YLIOPISTO**  
**Tietojenkäsittelytieteen laitos**

# Sisälllys

1	Projektin tiedot ja muutoshistoria.....	1
2	Johdanto.....	2
3	Sanasto.....	2
4	Projektiorganisaatio.....	3
4.1	Projektiin osallistuvat henkilöt.....	3
4.1.1	Asiakas: Juha Taina (juha.taina [at] cs.helsinki.fi).....	3
4.1.2	Projektin vastuuhenkilö: Kimmo Simola (kimmo.simola [at] cs.helsinki.fi).....	3
4.1.3	Projektin ohjaaja: Sampo Yrjänäinen (sampo.yrjanainen [at] cs.helsinki.fi).....	3
4.1.4	Projektiryhmä.....	3
4.2	Tapaamiset ja kommunikointi.....	4
5	Riskianalyysi.....	5
6	Projektin laitteisto- ja ohjelmistovaatimukset.....	6
6.1	Projektin toteuttamisen vaatimukset.....	6
6.2	Lopullisen tuotteen vaatimukset.....	6
6.2.1	Käyttäjävaatimukset.....	6
6.2.2	Ohjelmiston laitteistovaatimukset.....	6
7	Koko- ja kustannusarviot.....	6
7.1	Työaika-arvio.....	6
7.2	Ohjelmiston kokoarviot.....	6
7.2.1	LOC-analyysi.....	6
7.2.2	FP-analyysi.....	7
7.2.3	Kokoarvioiden luotettavuudesta.....	8
7.3	Kustannusarvio.....	8
8	Työn ositus.....	8
8.1	Projektin osat ja osatehtävät.....	8
8.1.1	Projektisuunnitelma.....	8
8.1.2	Vaatimusmäärittely.....	8
8.1.3	Suunnittelu.....	9
8.1.4	Tarkastukset.....	9
8.1.5	Toteutus + yksikkötestaus.....	9
8.1.6	Integrointi ja järjestelmätestaus.....	9
8.1.7	Tarkistuspisteet.....	9
8.1.8	Erilliset dokumentit.....	9
8.1.9	Lopputoimet.....	9
9	Projektin aikataulu.....	10
10	Seuranta- ja raportointimenetelmät.....	10
11	Liitteet.....	11

# 1 Projektin tiedot ja muutoshistoria

<b>Kurssi</b>	581260 Ohjelmistotuotantoprojekti (10 op)
<b>Projektiryhmä</b>	Krister Eklund Kim Ervasti Miina Kilpikivi Timo Korkama Hanna Leväniemi Santeri Pietilä
<b>Ohjaaja</b>	Sampo Yrjänäinen
<b>Asiakas</b>	Juha Taina
<b>Vastuhenkilö</b>	Kimmo Simola
<b>Kotisivu</b>	<a href="http://www.cs.helsinki.fi/group/opeapuri">http://www.cs.helsinki.fi/group/opeapuri</a>

OHJE: Versiointi tapahtuu siten, että viimeistä numeroa kasvatetaan, kun dokumenttia työstetään. Versionumeroa kasvatetaan aina. Keskimmäistä numeroa kasvatetaan, kun dokumentti jaetaan kommentoitavaksi. Ensimmäinen hyväksytty versio on 1.0.0.

Henkilö	Päivämäärä	Versio	Kommentti
Miina Kilpikivi	02.02.2007	0.0.1	Dokumenttipohja luotu
Miina Kilpikivi	02.02.2007	0.0.2	Leipätekstiasetuksia muokattu.
Kim Ervasti	05.02.2007	0.1.0	Sisältö siirretty pohjaan.
Kim Ervasti	07.02.2007	1.0.0	Julkaistu viimeistelty versio.
Kim Ervasti	25.02.2007	1.0.1	Lisätty riski
Kim Ervasti	18.03.2007	1.0.2	Tarkennettu riskejä. Muutettu aikataulun tehtäviä. Lisätty raportointimenetelmä.

## 2 Johdanto

Opeapuri on Helsingin yliopiston tietojenkäsittelytieteen laitoksen opiskelijoiden muodostama ryhmä, jonka tehtävänä on suunnitella ja toteuttaa projekti: opetuutorin apuväline.

Opetuutorin apuväline on työkalu, jonka avulla opettajatuutorit voivat hallita henkilökohtaisia tapaamisia ja ryhmätapaamisia. Työkaluun tulee kalenteri, jonne opettajatuutori voi merkitä hänelle sopivia henkilökohtaisten tapaamisten aikoja, ja joista tuutoroivat voivat sitten varata haluamansa ajan. Lisäksi työkalu tarjoaa keinot kirjata vapaamuotoisesti ylös tuutoroinnissa käsitellyt asiat. Ryhmätapaamisia varten opetuutorin apuvälineeseen tulee nimilista ja keino kirjata läsnäolot ylös tapaamisen jälkeen.

Tuutoroitavat voivat varata aikoja henkilökohtaiselle tapaamiselle ja tarkistaa heistä kirjatut asiat. Sekä tuutorit että tuutoroitavat kirjautuvat järjestelmään järjestelmän sisäisellä käyttäjätunnuksella ja salasananalla. Järjestelmän superkäyttäjä perustaa käyttäjätunnukset ja salasanat järjestelmässä.

## 3 Sanasto

XHTML 1.0	Ohjelmointikieli, jolla toteutetaan ohjelmiston Web-pohjainen käyttöliittymä.
CSS	Cascade Style Sheets, tyylimuotoilu.
PHP	Ohjelmointikieli, jolla toteutetaan Web-käyttöliittymän komentosarjat (toiminnot).
SVN	Subversion. Versionhallintajärjestelmä, johon talletetaan kollektiivisesti projektin tuotokset.
Wiki	Verkkosivusto, jolla oikeutettu jäsenistö voi lisätä tekstipohjaista tietoa sivulle kaikkien nähtäväksi.
TRAC	Kehitysympäristö joka tarjoaa rajapinnan SVN:lle ja Wiki:lle.
SQL	Tietokannan hallitsemisessa käytettävä kuvauskieli.
Oracle	SQL-tietokantaohjelmisto.

## 4 Projektiorganisaatio

### 4.1 Projektiin osallistuvat henkilöt

Projektiorganisaation kuuluvat projektin asiakas, projektin vastuhenkilö, projektiryhmän ohjaaja sekä projektiryhmä.

#### 4.1.1 Asiakas: Juha Taina (juha.taina [at] cs.helsinki.fi)

Projektiryhmän asiakas on projektin tuotteena syntyvän ohjelmiston tilaaja ja loppukäyttäjä. Asiakas osallistuu projektin toimintaan vaatimusmäärittelyvaiheessa.

#### 4.1.2 Projektin vastuhenkilö: Kimmo Simola (kimmo.simola [at] cs.helsinki.fi)

Vastuhenkilö vastaa projektin hallinnoinnista sekä toimii ylimääräisenä prosessiasiantuntijana.

#### 4.1.3 Projektin ohjaaja: Sampo Yrjänäinen (sampo.yrjanainen [at] cs.helsinki.fi)

Ohjaaja toimii ohjelmistotuotantoprosessin asiantuntijana. Ohjaajan tehtävänä on avustaa ryhmä alkuun sekä toimia laaduntarkkailijana.

#### 4.1.4 Projektiryhmä

Projektiryhmässä on kuusi jäsentä. Kullakin ryhmän jäsenellä on oma päävastuualueensa, joka kattaa jonkin projektin ydintoiminnoista (projektin johtaminen, vaatimusmäärittely, suunnittelu, koodaus, testaus ja dokumentointi). Lisäksi kullekin ryhmän jäsenelle on määrätty myös varavastuualue poissaolojen tai sairastumisten varalta.

Ryhmän jäsenet, heidän pää- ja varavastuualueensa sekä yhteystietonsa ovat:

Kim Ervasti  
Projektipäällikkö, varasuunnitteluvastaava  
kim.ervasti [at] cs.helsinki.fi  
050-3683332

Krister Eklund  
Suunnitteluvastaava, varaprojektipäällikkö  
krister eklund [at] cs.helsinki.fi  
044-0381310

Miina Kilpikivi  
Dokumenttivastaava, varavaatimusvastaava  
kilpikiv [at] cs.helsinki.fi  
040-7401317

Timo Korkama  
Testausvastaava, varadokumenttivastaava  
timo.korkama [at] cs.helsinki.fi  
040-5431774

Hanna Leväniemi  
Koodivastaava, varatestausvastaava  
levaniem [at] cs.helsinki.fi  
044-5315081

Santeri Pietilä  
Vaatimusvastaava, varakoodivastaava  
sjpietil [at] mappi.helsinki.fi  
050-5834355

## 4.2 Tapaamiset ja kommunikointi

Projektiryhmä kokoontuu kaksi kertaa viikossa, maanantaisin klo 16 ja torstaisin klo 9.15. Tapaamisten kesto on noin 1,5 tuntia. Tapaamisissa käydään läpi projektin etenemistä sekä tulevia tehtäviä. Tapaamisten puheenjohtajana toimii projektipäällikkö ja sihteerinä joku ryhmän jäsenistä. Tapaamisten kokouspöytäkirjat tallennetaan ryhmän kotisivulle. Kokouksia järjestetään 15.1. - 3.5.2007. Yliopiston lukukausiperiodien vaihtuessa viikolla 10 (5.-11.3.2007) tapaamisia ei järjestetä.

Viikoittaisten kokousten lisäksi ryhmän jäsenet järjestävät tapaamisia tarvittaessa. Lisätapaamisissa eivät välttämättä ole läsnä kaikki ryhmän jäsenet.

Projektiryhmän välinen pääasiallinen kommunikointi tapahtuu ryhmän tapaamisissa (joihin osallistuvat aina ryhmän ohjaaja sekä projektiryhmä) sekä sähköpostin välityksellä. Lisäksi projektiryhmän jäsenet voivat käyttää halutessaan IRC / Messenger -yhteydenpitoa projektin eri työvaiheissa. Kiireisissä asioissa ryhmän jäsenet ovat tavoitettavissa myös puhelimitse.

## 5 Riskianalyysi

Seuraavassa taulukossa on listattuna joitain riskejä jotka voisivat hankaloittaa projektin valmistumista. Riskejä seurataan aktiivisesti viikottaisissa kokouksissa. Taulukossa käytetyt todennäköisyydet ovat pieni, huomioitava ja suuri.

Riski	Todennäköisyys	Vakavuus	Varautuminen
PHP:n opettelu tuottaa ongelmia.	Pieni. Kaikki eivät osaa PHP:tä entuudestaan, mutta PHP on helppo kieli oppia.	Ei vakavaa, osaavimmat henkilöt voivat auttaa muita.	Ongelma yritetään havaita heti alussa, jolloin toteutuskielen vaihto Java-servletteihin vielä onnistuu.
Joku ryhmäläinen saattaa sairastua, tietokone saattaa hajota, mikä estää työskentelyä ennalta tuntemattoman pituisena ajanjaksona.	Todennäköinen.	Ei kovin vakavaa, koska ryhmässä riittää jäseniä.	Sairastumisiin ei varauduta. Ryhmähenkeä yritetään pitää yllä mm. projektin ulkopuolisilla tapaamisilla. Projektin aikataulussa pyritään pitämään liukumavaraa.
Joku ryhmäläinen saattaa olla estynyt kurssin loppuunsaatokselta.	Pieni.	Ei kovin vakavaa, koska ryhmässä on paljon jäseniä.	Hyvää ryhmähenkeä yritetään pitää yllä, ettei keskeytyksen syy tule ryhmän sisältä. Tapauksen toteutuminen vaatii aikataulun uudelleen miettimisen.
Aikataulu ei pidä.	Huomioitava.	Vakavaa, lisäaikaa ei myönnetä.	Projektipäällikkö seuraa, että projekti etenee aikataulun mukaisesti. Projektipäällikkö hoputtaa muita ryhmän jäseniä tarvittaessa. Projektin aikataulussa on liukumavaraa. Jos viivästyminen uhkaa, tiputetaan vähemmän tärkeät ominaisuudet pois ohjelmistosta.
Tiedostoja/tietoja tuhoutuu tahattomasti, esim kiintolevyrikossa.	Huomioitava.	Vakavaa – Hyvin vakavaa, riippuen menetetyistä tiedosto-/tietomäärästä.	Varaudutaan pitämällä tiedostot kaiken aikaa keskitetyssä versionhallinnassa, josta tiedostot edelleen kopioituvat jokaisen ryhmäläisen käyttökoneelle. Lisäksi versiohallinnan tiedostot varmuuskopioidaan koulun palvelimelle joka yö.

## 6 Projektin laitteisto- ja ohjelmistovaatimukset

Laitteisto ja ohjelmistovaatimukset jakautuvat kahteen osaan: projektin toteuttamisen vaatimukset sekä lopullisen tuotteen vaatimukset.

### 6.1 Projektin toteuttamisen vaatimukset

- Projektissa tullaan käyttämään TRAC-kehitysympäristöä.
- Kommunikointitavat tapaamisten ulkopuolella on vapaavalintaisia, pääasiassa käytetään ryhmälle luotua sähköpostilistaa, tai IRC / Messenger -protokollia. Kiireellisissä asioissa soitetaan.
- Projektissa käytetään yliopiston Oracle-SQL-palvelinta.
- Projektissa käytetään kokeiluluontoisesti myös Googlen tarjoamaa "Docs & Spreadsheets"-järjestelmää, jonka avulla useampi henkilö voi kirjoittaa samaa dokumenttia yhtä aikaa.

### 6.2 Lopullisen tuotteen vaatimukset

#### 6.2.1 Käyttäjävaatimukset

Ohjelmisto toteutetaan toimimaan tavallisella Web-selaimella. Käyttäjän kannalta riittää siis, että hänellä on Internet-yhteys ja nykyaikainen Web-selain.

#### 6.2.2 Ohjelmiston laitteistovaatimukset

Ohjelmisto pyörii palvelin-tietokoneella, jonka vaatimuksena on Oracle-tietokantajärjestelmä sekä PHP-tuella varustettu Apache-palvelinohjelmisto.

## 7 Koko- ja kustannusarviot

### 7.1 Työaika-arvio

Projektin toteuttamiseen laskettu aika on 240 tuntia / henkilö. Kuuden hengen projektiryhmällä tämä tarkoittaa siis yhteensä 1440 tunnin työmäärää.

### 7.2 Ohjelmiston kokoarviot

Ohjelmistojen kokoja voidaan arvioida LOC- tai FP-parametreilla. Tässä vaiheessa kokoarviot ovat vielä alustavia ja ne tarkentuvat projektin edetessä.

#### 7.2.1 LOC-analyysi

LOC (Lines Of Code) määrittää ohjelmiston koon koodiriveinä. Valmiin ohjelmiston voidaan arvioida



koostuvan 10-12 nettisivusta, joissa kussakin on noin 300 riviä koodia. Lisäksi tarvitaan tietokannan luomiseksi noin 100 riviä SQL-koodia sekä n. 200 riviä CSS-koodia. Tällöin kokonaissummaksi tulisi arviolta noin 3900 koodiriviä ( $12 * 300 + 100 + 200$ ).

## 7.2.2 FP-analyysi

Toinen menetelmä ohjelman koon arvioimiseksi on laskea sen toimintapistearvio eli FP-pisteet (FP = Function Point). FP-menetelmä perustuu ohjelman ulkoisten ominaisuuksien luokitteluun ja arviointiin: järjestelmän ulkoiset toimintapisteen (ohjelma syötteiden, tulosteiden, kyselyiden, tiedostojen ja ulkoisten liittymien lukumäärä) luokitellaan kolmeen eri luokkaan (yksinkertainen, normaali, vaikea) ja pisteytykset summataan erilaisilla painokertoimilla painotettuna sekä erillisen kompleksisuuskertoimen avulla.

FP:ssä arvioitavat tekijät sekä niiden pisteytys on esitetty Taulukossa 1.

	yksinkertaisia	*k	normaaleja	*k	vaikeita	*k	pisteet
syötteitä	7	* 3 +	7	* 4 +	0	* 6 =	49
tulosteita	12	* 4 +	0	* 5 +	0	* 7 =	48
kyselyjä	2	* 3 +	2	* 4 +	0	* 6 =	14
tiedostoja	4	* 7 +	0	* 10 +	0	* 15 =	21
liittymiä	0	* 5 +	0	* 7 +	0	* 10 =	0
							yht: 132

Taulukko 1. Järjestelmän FP-analyysi.

Järjestelmän ulkoinen rajapinta on tässä tarkastelussa käyttäjän ja www-selaimen välinen rajapinta. Järjestelmä on näin käsitettynä osittain hyvin itsenäinen, mistä johtuen esim. ulkoisia liittymiä eli järjestelmän käyttämiä sen ulkopuolella säilytettäviä aineistoja tai rajapintoja muihin järjestelmiin ei ole.

Pisteytyksen lisäksi toimintapisteanalyysissä lasketaan kompleksisuuskertoimen, jonka avulla tulosta säädetään kaavan

$$FP = FP_0 \times (0.65 + 0.01 \times \text{sum}(F_i))$$

avulla, missä  $FP_0$  on taulukossa 1 esiintyvien pisteiden summa sekä  $\text{sum}(F_i)$  kompleksisuuskertoimen, joka lasketaan summaamalla neljäntoista vaikeusasteikolla 1-5 arvostellun tekijän pisteet yhteen.

Järjestelmälle määritettiin kompleksisuustekijäksi  $\text{sum}(F_i)$  arvo 32. Tällöin saadaan toimintapisteanalyysin tulokseksi

$$FP = 132 \times (0.65 + 0.01 \times 32) \approx 128$$

### 7.2.3 Kokoarvioiden luotettavuudesta

LOC-arviossa lopputulos riippuu ohjelmointikielestä - tässä raportissa koodirivien määrä on arvioitu PHP-kielen mukaan. FP-analyysi puolestaan ei riipu ohjelmointikielestä, mutta luokitteluvaiheessa saattaa syntyä luokittelijasta johtuvia subjektiivisia epätarkkuuksia.

Molemmista luokittelumenetelmistä on todettava, että ne ovat suunnittelun tässä vaiheessa alustava arvio. Ohtu-kurssin materiaaleissa (Juha Taina: A brief introduction to function points, <http://www.cs.helsinki.fi/group/ohtu/resurssit/fp.html>) annetaan PHP-kielille karkea suuruusluokkasuhde LOC/FP = 67. Tämän järjestelmän osalta suhde on 30, eli ainakin lähdemateriaalin arvioon nähden liian pieni. Onkin erittäin mahdollista, että esim. koodirivien määrä on arvioitu liian alhaiseksi, mikä selittäisi analyysipisteiden alhaisen suhdeluvun.

## 7.3 Kustannusarvio

Projektilla ei ole kustannuksia. Asiakas saa ohjelmiston ilmaiseksi käyttöönsä.

## 8 Työn ositus

### 8.1 Projektin osat ja osatehtävät

Projektille on ennalta määrätty ulkoapäin yksi tarkistuspiste: projektien demopäivä. Koska käytämme vesiputousmallia, ja suunnitteludokumentti ja vaatimusdokumentti näyttelevät hyvin tärkeitä osia projektissamme, päätimme, että niille molemmille on syytä pitää tarkastus, ennen niiden hyväksymistä jäädytettäväksi. Vaatimusdokumentille pidetään asiakkaan kanssa epävirallinen tarkastus, ja suunnitteludokumentille virallinen tarkastus ryhmän kesken.

Projektin eri osat on jaettu niiden sisäisten osien (pienen kokonaisuuksien) mukaan osatehtäviksi siten, että niillä on joku looginen riippuvuussuhde jonkin toisen projektin vaiheen osan kanssa. Seuraavassa on projektin osat ja osatehtävät luetteloituina. Osatehtävän yhteydessä on esitelty aikataulun yhteydessä käytetty osatehtävän koodi.

#### 8.1.1 Projektisuunnitelma

- Projektisuunnitelma 1 = Alustava projektisuunnitelman laatiminen ja aikatauluttaminen
- Projektisuunnitelma 2 = Projektisuunnitelmadokumentti

#### 8.1.2 Vaatimusmäärittely

- Vaatimusmäärittely 1 = Vaatimusten keruu kokouksessa asiakkaan kanssa
- Vaatimusmäärittely 2 = Vaatimusten analyysi + käyttötapaukset
- Vaatimusmäärittely 3 = Käyttöliittymädemon tekeminen asiakkaalle + asiakkaan palaute
- Vaatimusmäärittely 4 = Vaatimusdokumentin kirjoittaminen
- Vaatimusmäärittely 5 = Vaatimusdokumentin mahdolliset korjaukset asiakkaan palautteen jälkeen + viimeistely

### 8.1.3 Suunnittelu

- Suunnittelu 1 = Arkkitehtuurisuunnittelu, Abstraktisuunnittelu
- Suunnittelu 2 = Rajapintasuunnittelu, Komponenttisuunnittelu
- Suunnittelu 3 = Algoritmien suunnittelu, Tietorakenteiden suunnittelu
- Suunnittelu 4 = Suunnitteludokumentin viimeistely
- Suunnittelu 5 = Käyttöliittymäsuunnittelu. Pidettäneen hiukan erillään muusta suunnittelusta. Vaatii kuitenkin hyvää yhteistyötä muun suunnittelun kanssa.

### 8.1.4 Tarkastukset

- Tarkastus 1 = Vaatimusdokumentin epävirallinen tarkastus asiakkaan kanssa
- Tarkastus 2 = Suunnitteludokumentin virallinen tarkastus

### 8.1.5 Toteutus + yksikkötestaus

- Toteutus 1 = Alustavaa sivupohjien ja rajapintojen ohjelmointia tähän asti suunnitellun järjestelmän pohjalta
- Toteutus 2 = Tietokantaa käsittelevien funktioiden ohjelmointi (tietokantakerros)
- Toteutus 3 = Ohjelman eri komponenttien toteutus + testaus yksi kerrallaan (dynaaminen kerros)
- Toteutus 4 = Ohjelman toteutuksen (koodin) viimeistely, sivujen ulkoasun hiominen

### 8.1.6 Integrointi ja järjestelmätestaus

- Testaus 1 = Luento testauksesta
- Testaus 2 = Dokumentin valmistelu (tästä tehtävästä sovittu dokumenttivastaavan kanssa)
- Testaus 3 = Testaussuunnittelun laatiminen
- Testaus 4 = Integrointi
- Testaus 5 = Järjestelmätestaus

### 8.1.7 Tarkistuspisteet

- Tarkistuspiste = Demotilaisuus

### 8.1.8 Erilliset dokumentit

- Käyttöohje
- Ylläpidodokumentti
- Yhteenvedodokumentti

### 8.1.9 Lopputoimet

- Lopputoimet 1 = Ohjelmiston viimeistely
- Lopputoimet 2 = Ohjelmiston palautus asiakkaalle

## 9 Projektin aikataulu

Projektille on määritetty ulkoapäin kiinteitä päivämääriä ja aikarajoja. Projekti alkaa 15.1.2007 ja päättyy 6.5.2007. Projektin aikaraja on joustamaton ja on palautettava joka tapauksessa määräaikaan mennessä. Tuote joudutaan siis palauttamaan keskeneräisenä, mikäli projekti ei ole aikarajaan mennessä valmis. Projektien demopäivän tarkka päivämäärä saadaan tietää myöhemmin, mutta se pidetään noin viikkoa ennen projektin päättymistä. Testauksesta pidetään kaikille yhteinen luento 23.2, jota ennen ei testausta ehkä kannata aloittaa.

Itse aikataulu on liitteenä.

## 10 Seuranta- ja raportointimenetelmät

Projektin seuranta suoritetaan jatkuvasti koko projektin keston ajan. Viikkotapaamisissa pidettyjen etenemiskatsausten lisäksi kukin projektin jäsen kirjaa käyttämänsä työtuntimäärät sekä tarvittavat metriikkasarjat Ohjelmistotuotantoprojektin tietojärjestelmä-työkalun avulla ([http://db.cs.helsinki.fi/~tkt\\_ohju/metrics/v0/](http://db.cs.helsinki.fi/~tkt_ohju/metrics/v0/)).

Projektipäällikkö seuraa projektin etenemistä ja huolehtii aikataulussa pysymisestä eri työvaiheiden vastuuhenkilöiden avulla. Projektin eri osa-alueista vastuussa olevat henkilöt raportoivat projektipäällikölle ennen viikkotapaamista kyseisen osa-alueen etenemisestä ja sen hetkisestä tilasta: mitä on tehty edellisen viikkotapaamisen jälkeen, missä vaiheessa ollaan nyt, mitä pitäisi tehdä seuraavaksi, ja mistä asioista omaan vastuualueeseen liittyen pitäisi seuraavassa kokouksessa keskustella.

Viikkotapaamisten lisäksi projektin aikana järjestetään kaksi tarkastusta. Ensimmäinen epävirallisempi tarkastus tehdään vaatimusdokumentille juuri ennen sen jäädytystä. Toinen, virallinen tarkastus tehdään suunnitteludokumentille sen ollessa viimeistelyä vaille valmis.

Projektissa tuotetut dokumentit jaetaan sisäisiin ja ulkoisiin dokumentteihin. Kaikki valmiit dokumentit kootaan ryhmän www-sivulle <http://www.cs.helsinki.fi/group/opeapuri/>. Tärkeimpiä dokumentteja ovat raportit, joita kootaan kunkin työvaiheen päättämisen yhteydessä. Raportit voivat olla sisäisiä tai ulkoisia. Projektin ulkoisia, asiakkaalle näkyviä raportteja ovat vaatimusdokumentti, jossa määritetään järjestelmälle asetettavat vaatimukset ja rajoitteet sekä lopullisen ohjelmiston käyttöohje ja ylläpidodokumentti. Näiden lisäksi ryhmä käyttää projektin seurannassa ja toteutuksessa apuna sisäisiä raportteja, joita ovat projektisuunnitelma, suunnitteludokumentti, testausuunnitelma sekä yhteenvetodokumentti.

Raporttien lisäksi ryhmä tekee viikkotapaamisistaan pöytäkirjat. Muuta dokumentaatiota ovat tarkastusten tarkastuslistat, mahdolliset välivaihemuistiot, ohjelmistokommentit (toteutuksen dokumentointi) sekä testiskriptien kommentit (testitapausten dokumentointi).

## 11 Liitteet