HELSINGIN YLIOPISTO

TIETOJENKÄSITTELYTIETEEN LAITOS

Ohjelmistotuotantoprojekti

PAKETTI

Projektisuunnitelma

Marko Buuri
Lasse Gustafsson
Tommi Rantala
Pekka Rousu
Jaakko Saaristo

16. joulukuuta 2003

Versiohistoria

Versio	Päivämäärä	Muutokset
1.5	16.12.2003	Siirretty seurantapalaveri, tarkastettu ulkoasu
1.4	19.11.2003	Aikataulua tarkistettu, lisätty dokumentteihin muutosdokumentti sekä ylläpitodokumenttiin liitteeksi englanninkielinen protokollan kuvaus
1.3	19.10.2003	Suunnittelua pidennetty viikolla
1.2	9.10.2003	Aikataulu tarkistettu
1.1	18.9.2003	Asiakkaan palautteen perusteella korjattu versio
1.0	15.9.2003	Ensimmäinen asiakkaalle toimitettu versio

Sisältö

1 Johdanto

Projekti kuuluu Helsingin yliopiston Tietojenkäsittelytieteen laitoksen kuuden opintoviikon laajuiseen Ohjelmistotuotantoprojekti-kurssiin. Projektin tavoitteena on tuottaa ohjelmisto Linux-käyttöjärjestelmän pakettien vuorottajan hallintaan. Ohjelmisto on suunnattu ensisijaisesti Tietojenkäsittelytieteen laitokselle tutkimuskäyttöön. Ohjelma julkaistaan mahdollisesti GPL-lisenssin alla. Projektin ohjaajana toimii Turjo Tuohiniemi, vastuuhenkilönä Juha Taina ja asiakkaana Jukka Manner. Projekti toteutetaan aikavälillä 2.9. - 31.12.2003.

1.1 Tavoite

Linux-ytimessä on kehittyneet ominaisuudet IP-pakettiliikenteen ohjaamiseen. Näiden ominaisuuksien avulla järjestelmän ylläpitäjä voi toteuttaa erilaisia protokolla- tai yhteyskohtaisia laatutakuita (engl. Quality of Service, QoS). Ytimessä on rajapinta laatutakuiden hallinnointiin, minkä lisäksi ylläpitäjälle on tarjolla rajapintaa käyttävä komentorivipohjainen työkalu *tc*. Tämän projektin tavoitteena on tuottaa Linuxin laatutakuiden hallintaan tc-työkalua helppokäyttöisempi graafinen ohjelmisto.

1.2 Henkilöt

Paketti-ryhmä koostuu viidestä tietojenkäsittelytieteen opiskelijasta. Ryhmän jäsenet ja heidän vastuualueensa on esitetty taulukossa 1. Projektipäällikkö Jaakko Saaristo toimii ryhmän kokousten puheenjohtajana ja sihteerin tehtävät on jaettu viikoittaisesti muiden jäsenten kesken.

1.3 Resurssit

Projektiryhmän käytössä on Tietojenkäsittelytieteen laitoksen ATK-palvelut, joihin kuuluu ryhmähakemisto, sähköpostilista sekä CVS-versionhallintajärjestelmä. Ryhmä käyttää Tietojenkäsittelytieteen laitoksen saleja ja tietoliikennelaboratorion tietokoneita, sekä oppimiskeskus Aleksandrian [HEL03] ryhmätyötiloja.

2 Ohjelmisto

Ohjelmisto on tarkoitettu Linux-käyttöjärjestelmän laatutakuiden hallinnointiin. Sen oletettuja käyttäjiä ovat Tietojenkäsittelytieteen laitoksen tutki-

Nimi	Sähköposti	Vastuualue
Jaakko Saaristo	jaakko.saaristo<at>helsinki.fi	Projektipäällikkö
Marko Buuri	buuri<at>cs.helsinki.fi	XML Ohjelmiston laatu Testaus
Pekka Rousu	prousu<at>cs.helsinki.fi	C-ohjelmointi Versionhallinta
Tommi Rantala	tommi.rantala<at>cs.helsinki.fi	Dokumentaatio Tc-työkalu Web-sivut
Lasse Gustafsson	lasse.gustafsson<at>helsinki.fi	Käyttöliittymä Java-ohjelmointi Tietoliikenne
Turjo Tuohiniemi	turjo.tuohiniemi<at>cs.helsinki.fi	ohjaaja
Jukka Manner	jukka.manner<at>cs.helsinki.fi	asiakas
Juha Taina	juha.taina<at>cs.helsinki.fi	vastuuhenkilö

Taulukko 1: *Projektin henkilöt*

musryhmien jäsenet, ennen kaikkea järjestelmäylläpitäjät, sillä laatutakuiden hallinta vaatii Linux-järjestelmässä pääkäyttäjän (root) käyttöoikeudet.

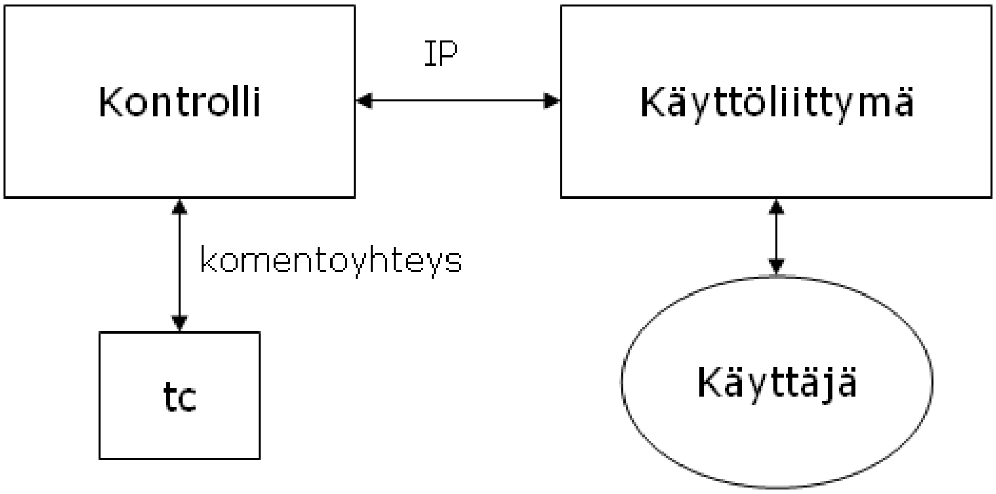
2.1 Rakenne

Ohjelmiston toiminnallisuus hajautetaan seuraaviin komponentteihin:

Kontrollikomponentti: on palvelu, joka tarjoaa rajapinnan Linux-ytimen pakettiskeduleriin. Komponentti käyttää suoraan tc-ohjelmaa ja se suoritetaan siinä tietokoneessa, minkä IP-pakettiliikennettä halutaan muokata.

Käyttöliittymä: mahdollistaa kontrollikomponentin käytön graafisesti. Esittää pakettiskedulerin tilan esimerkiksi puurakenteena ja sallii asetusten muuttamisen havainnollisella tavalla. Voi sijaita tietoliikenneverkossa eri tietokoneella kuin kontrollikomponentti.

Ohjelmiston rakenne on esitetty kuvassa 1. Komponenttien välinen tietoliikenne toteutetaan IP-protokollalla käyttäen rajapinnan toteutuksessa XML-kuvauskieltä. Komponenttijako on suunniteltu huomioiden sovelluksen jatkokehittäminen. Kontrollikomponentti on mahdollista tulevaisuudessa korvata esimerkiksi sellaisella, joka käyttää suoraan Linux-ytimen rajapintaa ilman tc-ohjelmaa. Myös käyttöliittymä on mahdollista korvata esimerkiksi WWW-pohjaisella ratkaisulla.



Kuva 1: Ohjelmiston rakenne

2.2 Ympäristö

Ohjelmisto suunnitellaan toimimaan CSL Linux-ympäristössä [KAL03]. Käyttöliittymän toteutuskieli on Java ja ympäristö JDK 1.4.1 [SUN03]. Kontrollikomponentin toteutuskieli on C ja ympäristö GCC 3.2.2 [GNU03]. Kontrollikomponentti käyttää verkkoliikenteeseen xinetd-palvelua [BRO03].

2.3 Kokoarvio

Ohjelmiston laajuus on arvioitu SLOC-arvona, joka ilmaisee lähdekoodin rivimäärän ilman kommentteja tai tyhjiä rivejä. Ohjelmistotuotantoprojektin ohjeellinen SLOC-määrä henkilöä kohden projektin aikana on 450-700, joka viiden hengen ryhmässä tarkoittaa projektin kokoluokkaa 2250-3500 SLOC. Java- ja C-kielten ilmaisuvoimien erot on otettu huomioon komponenttien kokoarviossa.

SLOC-arviot projektin pääkomponenteille ja niiden alikomponenteille on eritelty taulukossa 2. Ohjelmiston kokoarvio on yhteensä 3300 SLOC eli keskimäärin 660 SLOC henkilöä kohden.

Komponentti	Alikomponentti	**SLOC**
Kontrollikomponentti	Tietoliikennekomponentti	300
(C-kieli)	Tietorakennetoiminnot	500
	tc-komentotyökaluun kytkentä	100
	Levyjärjestelmätoiminnot	100
	Muita komponentteja	200
	Yhteensä	**1200**
Käyttöliittymä	Käyttöliittymä	1400
(Java)	Tietoliikennekomponentti	200
	Tietorakennetoiminnot	200
	Levyjärjestelmätoiminnot	100
	Muita komponentteja	200
	Yhteensä	**2100**

Taulukko 2: *Projektin pääkomponenttien kokoarviot*

Kokoarvio perustuu ryhmän jäsenten aikaisempaan ohjelmointikokemukseen.

3 Projektin eteneminen

Projekti etenee laaditun vaihejaon ja aikataulun mukaisesti. Kukin projektiryhmän jäsen on sitoutunut käyttämään projektin eteen vähintään 20 tuntia viikottaista työaikaa. Aikataulua muutetaan vain pakottavista syistä.

3.1 Työvaiheet

Projekti toteutetaan käyttäen lineaarista projektimallia (vesiputousmalli). Sen osat ovat määrittely, suunnittelu, toteutus, testaus ja ylläpito. Ohjelmiston ylläpito rajataan projektin ulkopuolelle, ja sen sijaan asiakkaalle toimitetaan luovutusvaiheessa ylläpitodokumentti. Toteutus- ja testausvaiheet limittyvät osittain päällekkäin.

Määrittely
Määrittelyvaiheessa kartoitetaan ohjelmiston vaatimukset, päätoiminnot, liittymät ja rajoitukset, ja tuloksista laaditaan määrittelydokumentti. Dokumentti toimitetaan asiakkaalle, joka esittää siihen vaatimansa muutokset ja korjaukset vaiheen lopussa järjestettävässä FTR-palaverissa (Formal Technical Review). Dokumentti päivitetään asiakkaan toiveita vastaavaksi suunnitteluvaiheen alussa. Määrittelyvaiheen alussa ryhmä laatii projektisuunnitelman, mikä toimii runkona projektin eri vaiheissa.

Suunnittelu
Suunnitteluvaiheessa määritellään tarkasti ohjelmiston arkkitehtuuri: tieto-

rakenteet, tietoliikenne, käyttöliittymä ja rajapinnat. Kuvauksen tarkkuudessa pyritään siihen, että ohjelmointi on sen pohjalta suoraviivaista. Tuloksena syntynyt suunnitteludokumentti tarkistetaan asiakkaan kanssa samalla tavalla kuin määrittelydokumentti, ja esiin tulleet muutokset päivitetään toteutusvaiheen alussa.

Toteutus
Varsinainen ohjelmointi alkaa toteutusvaiheen alussa ja pyritään saattamaan loppuun koodin tarkistuspalaveriin mennessä. Testaussuunnitelman laatiminen aloitetaan heti ohjelmoinnin alettua ja varsinainen testaus aloitetaan kaksi viikkoa ohjelmoinnin alettua, jo ennen varsinaisen testausvaiheen alkamista. Toteutusvaiheen aikana toimitetaan asiakkaalle esiversio käyttöohjeesta.

Testaus
Testaus saatetaan loppuun testausvaiheen aikana, jolloin ohjelmointi on jo päättynyt. Testauksen menetelmät ja tulokset dokumentoidaan testausdokumenttiin. Mahdolliset jäljelle jääneet ohjelmistovirheet dokumentoidaan tarkasti testausdokumenttiin. Testausvaiheen lopussa viimeistellään ohjelmiston käyttöohje sekä ylläpitodokumentti.

Luovutus
Projektin ehdoton takaraja on 31.12.2003, mutta projekti aikataulutetaan päättymään 16.12.2003. Luovutusvaiheessa järjestetään asiakkaalle demotilaisuus, missä esitellään valmista ohjelmistoa. Asiakkaalle luovutetaan käyttöohje, ohjelma ja sen lähdekoodi, sekä loppuraportti. Tammikuussa järjestetään projektin palautetilaisuus, jossa ohjaaja, ohjelmistotuotantoprojektien vastuuhenkilö ja asiakas arvioivat projektin onnistumista sekä annettua arvosanaa.

3.2 Aikataulu

Projektin aikataulu on esitetty taulukossa 3 sekä liitteenä olevassa Gantt-kaaviossa.

3.3 Työnjako

Projektipäällikkö Jaakko Saaristo vastaa projektiryhmän johtamisesta. Hän laatii projektin alussa projektisuunnitelman ja seuraa projektin edistymistä suunnitelman mukaisesti. Projektipäällikkö hoitaa pääosan kommunikaatiosta asiakkaan ja projektiryhmän välillä sekä raportoi projektin vastuuhenkilöille projektin edistymisestä.

Dokumentaatio: Tommi Rantala huolehtii projektin dokumentaation saattamisesta sisällöllisesti kattavaan ja ulkoisesti siistiin muotoon. Hän luo ryhmän jäsenille yhtenäisen säännöstön materiaalin kirjoittamisesta ja tallentamisesta projektin aikana, kokoaa yhteen eri jäsenten tuottamat dokumentit ja huolehtii niiden viimeistelystä.

Testaus: Marko Buuri vastaa testauksesta, kirjoittaa testaussuunnitelman sekä valvoo sen noudattamista.

Ohjelmiston laatu: Marko Buuri vastaa suunnittelun ja dokumentaation sekä ohjelmointitapojen laadusta.

Käyttöliittymä: Lasse Gustafsson vastaa käyttöliitymän suunnittelusta. Käyttöliittymävastaava luo käyttöliittymästä prototyypin jonka hän esittelee asiakkaalle. Käytettävyystestausta ei tiukan aikataulun vuoksi ehditä suorittaa. Hyväksytystä prototyypistä käyttöliittymävastaava ohjelmoi käyttöliittymän.

XML: Marko Buuri vastaa XML-komponenttien suunnittelusta ja toteutuksesta.

Tietoliikenne: Kontrollikomponentin ja käyttöliittymän välisen tiedonvälityksen suunnittelu ja toteutus tehdään Lasse Gustafssonin johdolla.

C-ohjelmointi: Pekka Rousu vastaa C-kielisten komponenttien toimivuudesta ja koodin luettavuudesta.

Java-ohjelmointi: Lasse Gustafsson vastaa Java-kielisten komponenttien toimivuudesta ja koodin luettavuudesta.

Tc-työkalu: Tommi Rantala on Linuxin tc-työkalun asiantuntija. Työkalun käyttöön liittyy sen syntaksin, parametrien ja paluuviestien hallinta.

Versionhallinta: Pekka Rousu huolehtii CVS-versionhallintajärjestelmään tallennetun sisällön järjestyksestä sekä päivittäisistä varmuuskopioista.

4 Dokumentointi

Projektin aikana ryhmä tuottaa taulukossa 4 esitetyt dokumentit.

Projektisuunnitelma
Hyväksymismenettely: Asiakas hyväksyy
Projektisuunnitelma pitää projektin aikataulussa.

Määrittelydokumentti
Hyväksymismenettely: Asiakas hyväksyy
Vaatimusanalyysin tuloksena saadaan määrittelydokumentti. Siinä kuvataan tuotteelta vaadittavat ominaisuudet. Vaatimusanalyysi sisältyy määrittelydokumenttiin. Dokumentissa kuvataan myös validointitason testit.

Suunnitteludokumentti
Hyväksymismenettely: Asiakas hyväksyy
Suunnitteludokumentti kuvaa tuotettavan ohjelmiston rakenteen. Dokumentti sisältää tietosuunnitelman, arkkitehtuurisuunnitelman, liittymäsuunnitelman ja moduulisuunnitelman.

Testaussuunnitelma
Sisältää kuvauksen tehtävistä moduuli-, luokka- ja integrointitason testeistä. Jokaisella tasolla tehdään sekä rakenteellista että toiminnallista testausta. Validointitason testit kuvataan määrittelydokumentissa.

Testausdokumentti
Sisältää testaussuunnitelman mukaisesti tehdyt testiaineistot ja testien tulokset.

Käyttöohje
Hyväksymismenettely: Asiakas hyväksyy
Sisältää loppukäyttäjälle tarkoitetun asennus- ja käyttöohjeen, sekä kuvauksen järjestelmän toiminnoista.

Muutosdokumentti
Sisältää tiedot suunnitteluun toteutusvaiheen aikana tehdyistä muutoksista.

Ylläpitodokumentti
Sisältää kuvauksen ohjelmiston toimintaperiaatteista ylläpidon avuksi. Sisältää liitteenä englanninkielisen protokollan kuvauksen.

Loppuraportti
Sisältää yhteenvedon projektista. Tähän dokumenttiin liitetään projektin sisäiset dokumentit: työtuntilistat, pöytäkirjat ja sisäiset muistiot.

5 Riskianalyysi

Tässä luvussa on arvioitu projektin riskejä ja niiden todennäköisyyttä asteikolla pieni, mahdollinen, todennäköinen.

5.1 Projektiryhmään liittyvät riskit

Riski: Henkilön sairastuminen.
Ennaltaehkäisy: -
Hallinta: Mikäli ryhmän jäsenen poissaolo kestää yhtäjaksoisesti yli viikon, projektipäällikkö jakaa henkilön tehtävät muiden jäsenten suoritettavaksi.
Todennäköisyys: Todennäköinen

Riski: Henkilö keskeyttää.

Ennaltaehkäisy: Projektipäällikkö jakaa tehtävät tasaisesti ryhmän kesken niin, ettei yksittäisiä henkilöitä rasiteta liialla työmäärällä.
Hallinta: Korjataan projektisuunnitelmaa ja projektin vaatimuksia.
Todennäköisyys: Pieni

5.2 Projektin hallintaan liittyvät riskit

Riski: Aikataulu ei pidä.
Ennaltaehkäisy: Projektipäällikkö seuraa ryhmän ajankäyttöä ja huolehtii aikataulussa pysymisestä.
Hallinta: Ryhmä keskustelee asiakkaan ja ohjaajan kanssa jatkosta.
Todennäköisyys: Pieni

Riski: Valittu prosessimalli ei sovellu projektin tarpeisiin.
Ennaltaehkäisy: Ryhmä käyttää prosessimallina tunnettua vesiputousmallia, joka on helpohko jokaisen ryhmän jäsenen sisäistää.
Hallinta: Projektipäällikkö keskustelee asiasta ohjaajan ja vastuuhenkilön kanssa.
Todennäköisyys: Pieni

5.3 Tekniikkaan liittyvät riskit

Riski: Työvälineitä ei hallita.
Ennaltaehkäisy: Ryhmä tutustuu työvälineiden dokumentaatioon ja ohjeisiin. Projektipäällikkö jakaa tehtävät osaamisen mukaan. Työvälineen hallitseva käyttäjä pitää muille ryhmän jäsenille aiheesta esitelmän.
Hallinta: Henkilö lähettää ilmoituksen ryhmän sisäiselle sähköpostilistalle. Työvälineen hallitseva käyttäjä opastaa työvälineen käytössä. Opastusta voidaan pyytää myös ryhmän ohjaajalta sekä asiakkaalta. Tarvittaessa projektipäällikkö korjaa projektisuunnitelmaa ja vaihtaa työvälinettä.
Todennäköisyys: Mahdollinen

Riski: Käytetyt työvälineet tai tekniikat eivät sovellu projektin tarpeisiin.
Ennaltaehkäisy: Käytetään sellaisia työvälineitä ja tekniikoita, jotka ovat tuttuja ainakin osalle ryhmän jäsenistä. Laatuvastaava valvoo, että käytetyt työvälineet ja tekniikat soveltuvat projektin tarpeisiin.
Hallinta: Vaihdetaan työvälineitä tai tekniikoita laatuvastaavan johdolla seurantakokouksissa ja korjataan projektisuunnitelmaa.
Todennäköisyys: Pieni

Riski: Tiedot katoavat palvelimilta tai versionhallintajärjestelmä vioittuu.
Ennaltaehkäisy: Varmuuskopiointiskripti tekee varmuuskopioita ryhmäha-

kemistosta ATK-osaston palvelimelle kerran vuorokaudessa.
Hallinta: CVS-vastaava palauttaa tiedot varmuuskopiolta.
Todennäköisyys: Pieni

5.4 Tuotteeseen ja asiakkaaseen liittyvät riskit

Riski: Asiakkaan vaatimukset ovat ristiriitaiset tai puutteelliset.
Ennaltaehkäisy: Vaatimukset selvitetään asiakkaan kanssa mahdollisimman tarkasti.
Hallinta: Projektipäällikkö keskustelee asiasta asiakkaan kanssa.
Todennäköisyys: Pieni

Riski: Asiakas muuttaa vaatimuksia määrittelyvaiheen jälkeen.
Ennaltaehkäisy: Sovitaan asiakkaan kanssa ettei vaatimuksia voi muuttaa määrittelyvaiheen jälkeen.
Hallinta: Projektipäällikkö keskustelee asiasta asiakkaan kanssa.
Todennäköisyys: Pieni

Riski: Ohjelmisto ei valmistu ohjelmistotuotantoprojektin loppuun mennessä.
Ennaltaehkäisy: Suunnitellaan ohjelmisto mahdollisimman hyvin ja pysytään aikataulussa.
Hallinta: Ryhmä keskustelee asiasta ohjaajan ja asiakkaan kanssa. Puuttuneet ja kesken jääneet ominaisuudet dokumentoidaan.
Todennäköisyys: Pieni

Riski: Asiakas on tyytymätön tuotteeseen.
Ennaltaehkäisy: Tuotetaan ohjelmisto hyväksi havaittujen periaatteiden mukaan. Pidetään asiakas tietoisena siitä mitä ollaan tekemässä. Ei tehdä liian optimistista projektisuunnitelmaa.
Hallinta: Kuunnellaan asiakkaan muutosehdotukset ja tehdään tarvittavat muutokset ohjelmistoon jos aikataulu sen sallii.
Todennäköisyys: Pieni

Määrittely	Projekti alkaa	2.9.2003
	Projektisuunnitelman laatiminen alkaa	2.9.2003
	Projektisuunnitelman toimitus asiakkaalle	15.9.2003
	Projektisuunnitelman katselmus	16.9.2003
	Määrittelydokumentin laatiminen alkaa	16.9.2003
	Määrittelydokumentin toimittaminen asiakkaalle	26.9.2003
	Määrittelydokumentin katselmus (FTR)	30.9.2003
Suunnittelu	Suunnitteludokumentin laatiminen alkaa	30.9.2003
	Seurantapalaveri	3.10.2003
	Seurantapalaveri	10.10.2003
	Suunnitteludokumentin toimittaminen asiakkaalle	17.10.2003
	Suunnitteludokumentin katselmus (FTR)	21.10.2003
	Seurantapalaveri	24.10.2003
	Suunnitteludokumentti valmis	28.10.2003
Toteutus	Ohjelmointi alkaa	28.10.2003
	Testaussuunnitelman laatiminen alkaa	28.10.2003
	Käyttöohjeen laatiminen alkaa	28.10.2003
	Ylläpitodokumentin laatiminen alkaa	28.10.2003
	Seurantapalaveri	6.11.2003
	Testaussuunnitelma valmis	13.11.2003
	Testaus alkaa	13.11.2003
	Käyttöohjeen toimittaminen asiakkaalle	21.11.2003
	Seurantapalaveri	21.11.2003
	Koodin katselmus (FTR)	28.11.2003
Testaus	Ohjelmointi päättyy	1.12.2003
	Testaus valmis	8.12.2003
	Loppuraportin kirjoittaminen alkaa	8.12.2003
	Seurantapalaveri	9.12.2003
	Projektien yhteinen demopäivä	11.12.2003
Luovutus	Käyttöohje valmis	16.12.2003
	Ylläpitodokumentti valmis	16.12.2003
	Ohjelmiston luovutus asiakkaalle	16.12.2003
	Testausdokumentti valmis	16.12.2003
	Loppuraportti valmis	16.12.2003
	Projektimapin luovutus ohjaajalle	16.12.2003
	Projektin ehdoton takaraja	31.12.2003

Taulukko 3: *Projektin aikataulu*

Dokumentti	**Kieli**	**Takaraja**
Projektisuunnitelma	Suomi	15.9.
Määrittelydokumentti	Suomi	26.9.
Suunnitteludokumentti	Suomi	28.10.
Testaussuunnitelma	Suomi	13.11.
Muutosdokumentti	Suomi	16.12.
Testausdokumentti	Suomi	16.12.
Käyttöohje	Englanti	16.12.
Ylläpitodokumentti	Englanti	16.12.
Loppuraportti	Suomi	16.12.

Taulukko 4: *Projektin aikana tuotettavat dokumentit*

Viitteet

[BRO03] Brown, R., Xinetd-palvelun kotisivu. http://www.xinetd.org, 14.9.2003

[GNU03] GNU-projekti, GCC-projektin kotisivu, 2003. http://gcc.gnu.org,14.9.2003

[HEL03] Helsingin yliopisto, Oppimiskeskus Aleksandrian kotisivu, 2003. http://www.helsinki.fi/aleksandria,14.9.2003

[KAL03] Kalliokoski, P., CSL-ympäristön erityisominaisuudet, 2003. http://www.ling.helsinki.fi/atk/admin/csl_features.xhtml,14.9.2003

[SUN03] Sun Microsystems, Java 2 -alustan v1.4.1 verkkodokumentaatio, 2003. http://java.sun.com/j2se/1.4.1/docs/index.html,14.9.2003

Liite 1: Gantt-kaavio