

# **Yhteenvetodokumentti**

## **Labra**

Helsinki 26.8.2008

Ohjelmistotuotantoprojekti

HELSINGIN YLIOPISTO

Tietojenkäsittelytieteen laitos

**Kurssi**

581260 Ohjelmistotuotantoprojekti (9+1op)

**Projektiryhmä**

Henrik Naakka, projektipäällikkö, suunnitteluvastaava

Tuomo Tilli, vaatimusmäärittelyvastaava

Anssi Kapanen, testaus- ja dokumenttivastaava

Antti Laaksonen, koodivastaava

**Asiakas**

Raimo Timonen

**Ohjaaja**

Paula Kuosmanen

**Vastuhenkilö**

Kimmo Simola

**Kotisivu**

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

**Versiohistoria**

Versio	Päiväys	Tehdyt muutokset
0.1	21.8.2008	Ensimmäinen versio
0.2	26.8.2008	Pieniä korjauksia

# Sisällysluettelo

1 Johdanto.....	4
2 Lopputuotteen esittely.....	4
3 Dokumenttien tiivistelmät.....	4
3.1 Projektisuunnitelma.....	4
3.2 Vaatimusmäärittelydokumentti.....	4
3.3 Suunnitteludokumentti.....	4
3.4 Testaussuunnitelma.....	5
3.5 Käyttöohje.....	5
3.6 Ylläpitodokumentti.....	5
4 Projektin päättöanalyysi.....	5
4.1 Ryhmän toiminta.....	5
4.2 Lopputuote.....	6

# 1 Johdanto

Tässä ohjelmistotuotantoprojektissa luotiin järjestelmä, jonka avulla voidaan varata aikoja harjoitustöiden tekoon ja palauttaa työraportit assistenttien tarkistettaviksi. Järjestelmän avulla pyritään vähentämään manuaalista työtä ja papereiden käsittelyä.

Tämä dokumentti sisältää yhteenvedon tehdystä tuotteesta ja projektin muista dokumenteista sekä projektin päättöanalyysin. Päättöanalyysissä kerrotaan, mikä projektissa oli hyvää ja mikä huonoa.

## 2 Lopputuotteen esittely

Labra-projektissa luotiin www-pohjainen järjestelmä harjoitustilaisuuksien ja laboratoriotöiden hallintaan sekä työraporttien palauttamiseen ja arvostelemiseen. Järjestelmän käyttäjinä ovat kemian laitoksen opiskelijat, assistentit ja vastuuhenkilöt. Heillä kaikilla on eri oikeudet järjestelmän eri osiin. Järjestelmä on räätälöity hyvin tarkasti asiakkaan tarpeisiin, mutta muokattavuuteen on pyritty mahdollistamalla eri asetusten muuttaminen käyttöliittymässä.

Järjestelmä on luotu täysin puhtaalta pöydältä asiakkaan tarpeisiin. Tietokannan ja käyttöliittymän suunnittelussa on menty vaatimusten perusteella. Tämä saattaa vaikuttaa järjestelmän muokattavuuteen.

Käyttöliittymässä on pyritty helppokäyttöisyyteen. Erityisesti opiskelijapuolesta on pyritty tekemään selkeä, koska he ovat järjestelmän suurin käyttäjäryhmä. Assistentti- ja vastuuhenkilöpuolella on enemmän toimintoja, ja sen käyttämisen oppimisessa on myös enemmän työtä. Järjestelmän käytön pitäisi kuitenkin onnistua ilman ohjeiden lukemista.

## 3 Dokumenttien tiivistelmät

### 3.1 Projektisuunnitelma

Projektisuunnitelma on projektin ensimmäinen dokumentti. Sitä kuitenkin on päivitetty koko projektin ajan. Dokumentti sisältää tiedot vastuualueiden jakautumisesta, analyysin projektin riskeistä, laitteisto- ja ohjelmistovaatimukset, kokoarvion projektista sekä projektin aikataulun. Dokumentin tärkein tehtävä on varmistaa, että projekti pysyy aikataulussa.

### 3.2 Vaatimusmäärittelydokumentti

Vaatimusmäärittelydokumentti sisältää asiakkaan sekä projektiryhmän laatimat vaatimukset järjestelmälle, jotka toimivat suunnittelun perustana. Dokumentti toimii myös asiakkaan ja projektiryhmän välisenä sopimuksena tuotettavan ohjelmiston ominaisuuksista. Dokumentissa on lueteltu järjestelmän käyttäjäryhmät, käyttötapaukset, käyttäjävaatimukset, järjestelmävaatimukset sekä ympäristövaatimukset. Käyttäjävaatimukset sekä järjestelmävaatimukset on jaettu toiminnallisiin ja ei-toiminnallisiin vaatimuksiin. Lisäksi dokumentissa on kerrottu järjestelmän elinkaaresta.

### 3.3 Suunnitteludokumentti

Suunnitteludokumentissa kerrotaan yksityiskohtaisesti, miten järjestelmä toteutetaan. Dokumentissa on ensiksi kuvattu järjestelmän yleisarkkitehtuuri ja järjestelmän arkkitehtuurikerrosten väliset

rajapinnat. Rajapinnoista on kuvattu, mitä funktiota mikäkin funktio kutsuu. Lisäksi dokumentissa on käyty läpi arkkitehtuurin suunnitelmassa esitettyjen kerrosten komponentit. Staattisen käyttöliittymäkerroksen komponentit on esitelty käyttöliittymäkuvien avulla, ja sivujen toiminnot on selitetty sanallisesti. Dynaamisen kerroksen komponentit on käyty läpi esittämällä kunkin tiedoston sisältämät funktiot ja muu toiminnallisuus taulukoiden ja selitysten avulla. Lopuksi dokumentissa on esitelty järjestelmän käyttämä tietokanta tietokantakaavion ja taulukohtaisten selitysten avulla.

### **3.4 Testaussuunnitelma**

Testaussuunnitelmassa kerrotaan miten järjestelmää testataan. Se koostuu kolmesta päävaiheesta, jotka ovat yksikkötestaus, integrointitestaus ja järjestelmätestaus. Jokaisesta vaiheesta käydään läpi lähestymistapa (miten testataan), testattavat kohteet (mitä testataan) ja hyväksymiskriteerit (milloin on testattu tarpeeksi).

### **3.5 Käyttöohje**

Käyttöohje toimii ohjeena järjestelmän käyttöön. Dokumentissa kerrotaan, miten järjestelmä asennetaan ja miten sitä käytetään. Dokumentin ohjeet järjestelmän eri ominaisuuksien käyttöön ovat melko suppeat, koska yksityiskohtaisemmat ohjeet löytyvät itse järjestelmästä. Järjestelmän käyttöliittymästä on myös pyritty tekemään niin selkeä, että varsinaista käyttöohjetta ei välttämättä edes tarvittaisi.

### **3.6 Ylläpidodokumentti**

Ylläpidodokumentissa kerrotaan asioita, jotka mahdollisten järjestelmän jatkokehittäjien ja ylläpitäjien on hyvä tietää. Kaikki poikkeavuudet suunnitteludokumentin ja varsinaisen lopullisen järjestelmän välillä on kuvattu tässä dokumentissa. Siksi ylläpidodokumenttia tuleekin lukea suunnitteludokumentin ja ohjelmakoodin kanssa samanaikaisesti.

## **4 Projektin päättöanalyysi**

### **4.1 Ryhmän toiminta**

Ohjelmistotuotantoprojektin normaali ryhmäkoko on kuusi henkilöä. Ryhmässämme oli alussa viisi henkilöä ja yksi ryhmän jäsen keskeytti jo projektin alussa vaatimusmäärittelyvaiheessa. Näin ollen ryhmään jäi vain neljä jäsentä. Aikaisempien kokemusten perusteella kurssin vastuuhenkilö kertoi, että neljä jäsentä on liian vähän ohjelmistotuotantoprojektiin ja että lähes aina neljän hengen projektit olivat epäonnistuneet. Emme kuitenkaan antaneet tämän lannistaa itseämme ja uskoimme, että saamme projektin vietyä kunnialla läpi.

Ryhmämme kokoontui kaksi kertaa viikossa koko projektin ajan. Toteutusvaiheessa sovimme lisäksi erillisiä koodaustapaamisia. Kommunikoinnissa oli alussa hieman puutteita, mutta mitään ongelmia siitä ei syntynyt. Tapaamisten lisäksi olimme yhteydessä sähköpostin välityksellä.

Vaikka ryhmämme jäsenmäärä oli pieni, ei missään vaiheessa tuntunut tulevan kiire. Ryhmän jäsenillä oli aikaa panostaa kurssiin tarvittava määrä. Vain aivan lopussa, toteutusvaiheessa, oli oltava tarkkana, että kaikki toiminnallisuudet saatiin toteutettua. Vaikka projektille olisi ollut mahdollisuus käyttää viikko lisäaikaa, päätimme, että se ei ole tarpeen.

Vastuualueiden jaossa ei ollut mitään ongelmia ja ne olivat hyvin selvillä alusta lähtien. Vastuualueet olisi kuitenkin pitänyt jakaa pienempiin osiin, jotta kaikille olisi ollut tasaisesti tehtäviä koko kurssin ajan. Tästä johtuen työtunnit eivät jakaantuneet kaikille täysin tasaisesti, varsinkaan kurssin alkuvaiheessa. Projektissa oli kaksi henkilöä, jotka olivat ohjelmoineet PHP:llä ennenkin, ja kaksi henkilöä, jotka eivät olleet ennen käyttäneet PHP:tä. Tästä johtuen koodausvastuut jakautuivat epätasaisesti. Työjako mahdollisti mahdollisimman tehokkaan työskentelynopeuden, joka loppujen lopuksi johti onnistuneeseen lopputuotteeseen.

## **4.2 Lopputuote**

Asiakkaalla oli hyvin selvillä, minkälaisen järjestelmän hän halusi. Näin ollen ryhmä sai jo aikaisin selville järjestelmän vaatimukset. Vaatimuksia tosin tarkennettiin vielä asiakkaalta sähköpostin avulla. Lopputuotteessa on toteutettuna lähes kaikki vaatimusmäärittelydokumentissa mainitut vaatimukset. Vain yksi alhaisimman prioriteetin vaatimus jäi toteuttamatta. Uskoimme projektin alkuvaiheessa, että ainakin alhaisimman prioriteetin vaatimukset jäisivät toteuttamatta ryhmän pienestä koosta johtuen. Näin ollen on pakko olla tyytyväinen lopputuotteeseen.

Järjestelmän käyttöliittymä on mielestämme onnistunut ja selkeä. Sitä ei kuitenkaan päästy testaamaan ryhmän ulkopuolisilla jäsenillä, mikä olisi ollut tärkeää käyttöliittymän suunnittelussa. Asiakas sai kuitenkin kommentoida järjestelmän käyttöliittymää ja sitä muokattiin asiakkaan kommenttien perusteella. Lopullista kaikki toiminnot sisältävää käyttöliittymää ei kuitenkaan enää ehditty muokkaamaan asiakkaan kommenttien perusteella, koska järjestelmä esiteltiin asiakkaalle projektin toiseksi viimeisenä päivänä.