

Ohjelmistotuotantoprojekti

Muutos- ja korjauspyyntöjen priorisointityökalu

Ryhmä Muppett

YHTEENVETODOKUMENTTI

Helsinki 1.9.2008

HELSINGIN YLIOPISTO
Tietojenkäsittelytieteen laitos

Kurssi:
Ohjelmistotuotantoprojekti, kesä 2008

Projekti:
Muutos- ja korjauspyyntöjen priorisointityökalu

Asiakas:
Oodi-konsortio/ Sampo Lehtinen

Ryhmä:
Arto Chydenius
Laura Haverinen
Merja Lindén
Topi Musto
Laura Ojala
Toni Sormunen

Ohjaaja:
Marko Lehtimäki

Dokumentin versiohistoria

<u>Versio</u>	<u>Päiväys</u>	<u>Muutokset</u>	<u>Muuttaja</u>
0.0	09.06.2008	Pohja	LH
0.1	01.09.2008	Sisältöä	LH
0.2	04.09.2008	Korjauksia	LO

Sisältö

1 Johdanto	1
2 Sanasto	1
3 Lopputuotteen esittely	1
4 Dokumenttien tiivistelmä	2
4.1 Projektisuunnitelma.....	2
4.2 Vaatimusmäärittelydokumentti.....	2
4.3 Suunnitteludokumentti.....	3
4.4 Testausdokumentti.....	4
4.5 Ylläpitodokumentti.....	4
5 Projektin päättöanalyysi	4
6 Yhteenveto	5

1 Johdanto

Tässä yhteenvetodokumentissa kuvataan järjestelmää, joka on tuotettu osana Helsingin yliopiston tietojenkäsittelytieteen laitoksen kesän 2008 Ohjelmistotuotantoprojekti -kurssin suoritusta. Järjestelmä tuotettiin ryhmätyönä ja nimitettiin ryhmän mukaisesti Muppett-järjestelmäksi.

Kurssin ensisijainen tavoite on tuottaa opiskelijoille oppimisprosessi. Tämä oppimisprosessi konkretisoituu asiakkaalta toimeksiantona saadun järjestelmän tuotoksena, johon liittyvät myös dokumentaatio niin tuotetusta järjestelmästä kuin myös valittuun prosessimalliin liittyvä projektin kuvaus.

Tässä dokumentissa esittelemme sanaston (luku 2) jälkeen luvussa 3 tuotteen, eli Muppett-järjestelmän. Tuote koostuu ohjelmakoodin lisäksi siihen liittyvästä dokumentaatiosta, ja siksi Muppett-järjestelmään liittyvistä dokumenteista on koostettu tiivistelmät lukuun 4. Projektin päättöanalyysi, luku 5, on suunnattu Ohjelmistotuotanto -kurssin järjestäjille ja se sisältää kurssilaisten ajatuksia kurssityöskentelystä.

Tässä dokumentissa viitataan kaikkiin Muppett-ryhmän dokumentteihin. Muiden dokumenttien lukeminen tämän dokumentin yhteydessä on suositeltavaa.

2 Sanasto

Lopputuote

Asiakkaan tilaama ohjelma dokumentteineen sellaisena kuin se on projektin päättyessä

Muppett-järjestelmä

Oodi-konsortion tilaama ohjelma muutos- ja korjauspyyntöjen priorisointiin

3 Lopputuotteen esittely

Ryhmä toteutti ohjelmistotuotantoprojektin yhteydessä Muppett-järjestelmän. Järjestelmä tarjoaa asiakkaalle web-pohjaisen järjestelmän, jolla asiakkaana ollevan konsortion jäsen yliopistot voivat nykyistä helpommin äänestää erilaisista muutoksista. Lisäksi järjestelmä helpottaa projektipäälliköiden työtä, joka liittyy äänestyksen luomiseen ja äänestysten lopputulosten laskemiseen. Muppett-järjestelmä toimii alustana äänestyksille ja laskee reaaliajassa Oodi-konsortiolle korjaus- ja muutospyyntöjen tärkeysjärjestyksen.

Muppett-järjestelmän toimintalogiikka on toteutettu Javalla, jotta se on helppo siirtää ympäristöstä toiseen asiakkaan niin halutessa. Käyttäjälle näkyvä osa järjestelmää on puolestaan JSP-sivuista muodostuva kokonaisuus, joka hyödyntää XHTML-standardia.

4 Dokumenttien tiivistelmä

Tässä luvussa esitellään Muppett-ryhmän projektin aikana tuottamia dokumentteja. Ensimmäisenä ryhmä tuotti projektisuunnitelman, jonka jälkeen ryhmä tuotti asiakkaan tilaamaa tuotetta käsitteleviä dokumentteja. Nämä dokumentit ovat vaatimusmäärittelydokumentti, suunnitteludokumentti, testausdokumentti ja ylläpitodokumentti. Käyttöohje on tuotettavan ohjelmiston osana verkossa. Lopuksi kurssin anti kootaan tähän yhteenvedodokumenttiin. Kaikki dokumentit ovat esillä ryhmän verkkosivuilla.

4.1 Projektisuunnitelma

Projektisuunnitelmassa esitellään ryhmän jäsenten vastualueet ja kerrotaan, mitä kukin vastuualue pitää sisällään. Projektisuunnitelmadokumentti esittelee lyhyesti projektin tehtävänannon ja yleisiä toteutusperiaatteita, esimerkiksi käytettävät kielet. Toteutuskieliksi valittiin Java 1.6, XHTML 1.0 ja SQL-kyselykieli.

Projektisuunnitelmassa esitetään arvio tuotettavan ohjelmiston koosta. Toimipisteanalyysin mukaan projektin kooksi tulisi reilut 7273 koodiriviä ja LOC-menetelmän mukaan koko olisi 2980 riviä koodia. Projektisuunnitelmaan kuuluu myös aikataulun suunnittelu koko projektin ajalle. Lisäksi oleellisena osana projektisuunnitelmaa on riskianalyysi,

jossa arvioidaan projektin aikana mahdollisesti projektin edistymistä hidastavien tai estävien tapahtumien todennäköisyyttä ja vakavuutta.

4.2 Vaatimusmäärittelydokumentti

Vaatimusmäärittelydokumentissa esitellään lähtötilanne, tässä tapauksessa Oodi-konsortion aikaisempi äänestyskäytäntö, ja järjestelmän tarkoitus. Dokumentissa kootaan yhteen erilaiset tuotteelle annetut vaatimukset. Vaatimusmäärittelydokumentti esittää myös järjestelmän sidosryhmät.

Vaatimusmäärittelyn osana tuotettiin käyttötapauskuvaukset. Tärkeimmistä käyttötapauksista esitetään vaatimusmäärittelydokumentissa sekä sanalliset kuvaukset että teknisemmät määrittelyt. Kaikki käyttötapaukset priorisoitiin. Järjestelmän toteutus pohjautuu näihin prioriteetteihin siten, että toteutusvaiheessa on pyritty ensin toteuttamaan kaikki ensimmäisen ja toisen prioriteetin käyttötapaukset.

Vaatimusmäärittelydokumentissa on listattu myös ei-toiminnalliset vaatimukset. Nämä vaatimukset koskevat lähinnä tuotteen ylläpidettävyyttä. Dokumentissa on myös kuvattu tarvittavien osien tietosisällöt.

4.3 Suunnitteludokumentti

Suunnitteludokumentti esittää järjestelmän toteutussuunnitelman. Suunnitelma etenee ylhäältä alaspäin: ensin esitellään arkkitehtuurit eli yleisen tason kuvaukset ja näistä jatketaan alas aina luokkakuvauksiin asti. Suunnitteludokumentti on pyritty toteuttamaan niin tarkasti, että mikä tahansa muukin toteutusryhmä voisi vaatimusmäärittelydokumentin ja suunnitteludokumentin pohjalta toteuttaa Muppett-järjestelmän.

Muppett-järjestelmässä käytetään useita eri tasoisia arkkitehtuureja ja sovellettu näissä suunnittelumalleja. Ylimmällä tasolla on MVC-eli malli, näkymä, ohjain -arkkitehtuuri, joka erottaa toiminnallisuuden käyttöliittymästä. Kussakin MVC-mallin osassa on suunnitelmaa vielä tarkennettu. Mallissa tietokannan hallintaan käytetään DAO/DTO-arkkitehtuuria ja DAO/DTO – olioiden luomiseen käytetään abstraktia tehdasta.

Ohjaimessa on käytössä Front Controller ja komentomallit, joiden mukaisesti järjestelmässä on yksi ainoa servletti hallinnoimassa palveluita ja niiden välistä kommunikaatiota.

Näkymä esittää järjestelmän tilan ja kommunikoi Front Controllerin välityksellä mallin ja ohjaimen kanssa.

4.4 Testausdokumentti

Testausdokumentissa esitellään testauksen yleiset periaatteet. Dokumentissa käsitellään yksikkötestausta, integrointitestausta ja järjestelmätestausta. Dokumentin ohessa esitetään testausraportit eri testeistä.

4.5 Ylläpitodokumentti

Ylläpitodokumentti kertoo, mitä oikeastaan tulikaan tehtyä. Ylläpitodokumentin päätarkoitus on esitellä poikkeamat suunnitelmasta ja kuvata muut ylläpitoon vaikuttavat seikat. Ylläpitodokumentissa kerrotaan myös esimerkiksi, kuinka järjestelmään luodaan ensimmäinen käyttäjä järjestelmän ulkopuolelta ja kuinka järjestelmään voidaan luoda uusia äänestystyyppejä.

Ylläpitodokumentissa raportoidaan myös toteuttamatta jääneet osat sekä havaitut puutteet ja virheet. Tämä helpottaa tuotteen jatkokehittelyä.

5 Projektin päättöanalyysi

Projektin päättöanalyysin on koottu kaikilta ryhmäläisiltä mielipiteitä. Analyysissa on arvioitu ainoastaan kurssityöskentelyä ja sen onnistumista, ryhmäläisiä ei ole arvioitu tässä yhteydessä.

Projekti sujui suhteellisen hyvin, vaikka kaikki olimme ensikertalaisia ja amatöörejä tämän kaltaisissa projekteissa. Suurin vaikeus ehkä olikin, että kukaan ryhmäläisistä ei oikein tiennyt mitä pitää tehdä missäkin vaiheessa. Projektin varsinainen alkuun saaminen oli hankalaa ja aikataulujen teko ei ollut kenelläkään hallussa. Kun päästiin vauhtiin, ryhmätyöskentely sujui varsin jouhevasti.

Kurssin aikataulu on tiukka, kuten erittäin usein myös todellisen elämän aikataulut. Aikataulun ollessa tiukka olisi kuitenkin hyvä määritellä kurssin tavoitteet tarkemmin. Koska kurssin tarkoitus on oppiminen, olisi hyvä tietää, mihin osa-alueeseen panostetaan, jos ja kun aikataulu pettää. Esimerkiksi olisiko tärkeämpää suunnitella kaikkien osien testaus, vai toteuttaa huolellisesti vain osa testeistä ja jättää loput tekemättä ja suunnittelematta. Onko tärkeämpää saada aikaan valmis järjestelmä heikommin dokumentoituna vai osa järjestelmästä ja dokumentit viimeisen päälle?

Kurssin oppimistavoite jäi avonaiseksi: Käytännössä vaaditaan monen vasta laudaturkurssin asioita. Onko kurssilla siis tarkoitus ottaa 'luulot' pois ja valmistaa tulevien kursien tärkeyden ymmärtämistä vai soveltaa vanhaa jo opittua? Kenties molempia, mutta ahneella ei ole kovin ruusuinen loppu. Realiteettien ymmärtäminen oppimistavoitteiden suhteen lienee kurssijärjestäjille haaste, johon tulee tulevaisuudessa vastata.

Kaiken kaikkiaan voimme olla tyytyväisiä projektiimme. Saavutimme suuren osan itsellemme asettamistamme tavoitteista, teimme asiakkaan ainakin osittain onnelliseksi ja heitimme paljon huonoa läppää. Ainut, mikä jäi kaivertamaan oli testauksen vaillinaisuus. Ensi kerralla koko prosessi menisikin jo paljon helpommin, ja ehkä testauksellekin jäisi hieman enemmän aikaa.

6 Yhteenveto

Tämä dokumentti päättää kesän 2008 ohjelmistotuotantoprojektin Muppett-ryhmän osalta. Dokumenttissa on luotu katsaus päättyvän projektin dokumentteihin, aikaansaatuun lopputuotteeseen ja ryhmän jäsenten tuntemuksiin projektista.