

Ohjelmistotuotanto 2013, Periodi III  
Miniprojektin raportti

Tuukka Peuraniemi  
Mikko Rautiainen  
Mikko Tamminen  
Mika Viinamäki  
Ilkka Vähämaa

26. huhtikuuta 2013

# 1 Projektin aloitus

Ryhmämme muodostui spontaanisti yhden kurssin luennon lopussa – suurin osa ryhmästä ei ollut aiemmin työskennellyt yhdessä.

Pääsimme melko vaivatta yhteisymmärrykseen projektin toteutustavasta – konso-lisovelluksen teko oli luonteva valinta, koska Swing ei herättänyt suuria intohimon tunteita ryhmäläisissä eikä toisaalta web-sovelluksen kehittämiseen Javalla ollut kenties tarpeeksi kokemusta ryhmän sisällä. Muita vaihtehtoja näiden lisäksi ei mietitty.

# 2 Projektin kulku

Ensimmäisessä sprintissä työskentelimme koko viikon tunnit neljän tunnin yhtenäisessä pätkässä, mutta muissa sprinteissä aikatauluongelmien vuoksi jouduimme työskentelemään kahden tunnin erillisissä pätkissä. Molemmat työskentelytavat osoittautuivat kuitenkin toimiviksi, jälkimmäinen ehkä jopa paremmaksi – oli helpompaa kunnolla keskittyä tekemiseen vain kahden tunnin ajan. Kaikki nämä pätkät työskentelimme laitoksella niin, että koko ryhmä oli samassa tilassa.

Vaikka ryhmämme koko oli melko suuri, suurimmaksi osaksi ajasta kaikille löytyi jotain tuottavaa tekemistä. Ryhmän koko yhdistettynä suhteellisen pieneen aiheeseen ja lyhyihin sprintteihin kuitenkin aiheutti ongelmia kehitystehtävien koordinoinnin suhteen. Välillä esimerkiksi kaksi eri henkilöä muokkasi samaa koh-taa koodista yhtäaikaan. Lisäksi testit tuppasivat hajoamaan kokonaan pienistäkin muutoksista.

Käytössä olevan ajan vähyiden vuoksi emme oikein malttaneet alussa jäädä kunnolla miettimään miten ohjelmisto olisi järkevää toteuttaa, vaan siirryimme suoraan koodaamiseen – aineksia olisi ollut suurempaankin katastrofiin, mutta oikeastaan emme joutuneet mihinkään varsinaiseen integraatiohelvettiin. Vähän kattavampi suunnittelu etukäteen olisi kuitenkin saattanut näkyä ohjelmakoodin laadussa positiivisella tavalla.

Etenkin ensimmäisessä sprintissä aikataulu osoittautuikin aika tiukaksi – siinä ja muissakin sprinteissä onnistuimme kyllä toteuttamaan kaikki vaaditut ominaisuudet, mutta emme paljonkaan ylimääraistä. Toisaalta tuhlasimme ensimmäisessä sprintissä aikaa arkkitehtuurin rakentamiseen tulevia sprinttejä varten, mitä ei pitäisi tehdä.

Prosessin noudattaminen ei noussut kovin keskeiseksi asiaksi projektin aikana, ja projekti tuntuikin ensisijaisesti ohjelmointiprojektilta ketterän prosessin harjoit-telemisen sijaan. Yleensä yksi ryhmän jäsen hoiti prosessiin liittyvät asiat kun-toon kussakin sprintissä. Syntyneissä artefakteissa ei ollut ilmeisesti kuitenkaan puutteita.

### 3 Tekniset haasteet

Jenkinsin kanssa värkkääminen kulutti yllättävän paljon aikaa, enemmän kuin laskuharjoituksissa. Aikaa ei kuitenkaan mennyt juurikaan hukkaan mihinkään hölmöihin ongelmiin.

Päätös käyttää XML:ää viitteiden tallentamiseen osoittautui huonoksi valinnaksi - käyttämämme, Javan standardikirjastoon kuuluva XML-kirjasto osoittautui epäintuitiiviseksi käyttää. Parempi vaihtoehto olisi kenties ollut käyttää jotain tietokantaa, esimerkiksi SQLiteä ja jotain yksinkertaista ORM-kirjastoa.

BibTeX-formaatti osoittautui myös vähän kummalliseksi eikä dokumentaatio ollut hirveän selkeää etenkään erikoismerkkien tallentamisen osalta, joten oikea tapa tallentaa tiedot jouduttiin osittain selvittämään yrityksen ja erehdyksen kautta. Tähän ei kuitenkaan mennyt kokonaisuudessaan hirveän paljon aikaa, koska LaTeX oli ennestään tuttu osalle ryhmästä.

Muutamaan otteeseen myös Javan standardikirjaston puutteet häiritsivät – toisaalta, koska käytimme Mavenia, olisimme voineet helposti lisätä esimerkiksi Apache Commons tai Google Guava -kirjastoja projektiimme. Jätimme sen kuitenkin tekemättä.

Kurssin laskuharjoituksissa esitellyille tekniikoille (mm. Mockito) olisi ollut käyttöä projektin kuluessa, mutta ne esiteltiin miniprojektin kannalta turhan myöhään eikä niitä tuotu projektiin enää mukaan.

Keskimäärin pääsimme varmaankin keskimääräistä vähäisemmillä teknisillä murheilla, koska emme toteuttaneet esimerkiksi graafista käyttöliittymää, web-sovellusta tai datan synkronointia pilveen, joista kaikista olisi tullut epäilemättä esiin monenlaisia ongelmia.

### 4 Yleisesti positiivista

Versionhallinnan käytöstä koko ryhmä sai arvokasta kokemusta – osalle ryhmästä versionhallinnan käyttö muiden kanssa oli uutta.

Tiimin yhteishenki oli erinomainen ja opimme paljon ohjelmointitekniikoista toisiltamme. Muun muassa enumien käyttö tuli osalle ryhmästä uutena asiana.

Oli myös mukava päästä kokeilemaan kaikenlaisia ohjelmistotuotannon tekniikoita käytännössä – osaa niistä tulee todennäköisesti käytettyä jatkossakin.