

## **Projektin yhteenvetodokumentti**

Viski-ryhmä

Helsinki 31.8.2006

Ohjelmistotuotantoprojekti

HELSINGIN YLIOPISTO

Tietojenkäsittelytieteen laitos

**Kurssi**

581260 Ohjelmistotuotantoprojekti (6 ov)

**Projektiryhmä**

Esa Elovaara  
Suvi Hiltunen  
Tomi Jylhä-Ollila  
Riku Louhimo  
Samuli Sairanen  
Juho Vuori

**Asiakas**

CSC / Aleks Kallio

**Johtoryhmä**

Juha Taina  
Jaakko Saaristo

**Kotisivu**

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

**Versiohistoria**

Versio	Päiväys	Tehdyt muutokset
1.0	31.8	Dokumentin valmis versio.

# Sisältö

<b>1</b>	<b>Johdanto</b>	<b>1</b>
<b>2</b>	<b>Sanasto</b>	<b>2</b>
<b>3</b>	<b>Lopputuotteen esittely</b>	<b>4</b>
3.1	Erot vaatimusmäärittelyyn ja suunnitteluun . . . . .	4
<b>4</b>	<b>Dokumenttianalyysi</b>	<b>5</b>
<b>5</b>	<b>Projektin päättöanalyysi</b>	<b>6</b>
5.1	Työvaiheet . . . . .	6
5.2	Toteutuneet riskit . . . . .	6
5.3	Ryhmäläisten kommentit . . . . .	6
<b>6</b>	<b>Yhteenveto</b>	<b>8</b>

# 1 Johdanto

CSC on opetusministeriön omistama tieteen tietotekniikan keskus, joka tarjoaa tutkijoille Suomen laajinta tieteellisten ohjelmistojen ja tietokantojen valikoimaa sekä Suomen tehokkainta superlaskentaympäristöä Funet-tietoliikenneyhteyksien kautta. CSC kehittää korkean tason analyysisovelluksia eri tieteenalojen käyttöön. Yksi konkreettinen esimerkki on bioinformatiikan alaan kuuluva DNA-mikrosirudataa varten kehitetty NAMI-analyysiympäristö. Järjestelmä koostuu graafisesta asiakasohjelmistosta sekä useista palvelinohjelmistoista. Tässä ohjelmistotuotantoprojektissa kehitetään visualisointikirjasto analyysiympäristöä varten.

Yhteenvetodokumentin tarkoitus on kerätä yhteen tuotteen kuvaus, lyhyet kuvaukset ja analyysit projektin tuottamista dokumenteista sekä analysoida jonkin verran ryhmän toimintaa ja onnistumista.

## 2 Sanasto

**Arkkitehtuurikäyttäjä** Ohjelmoija, joka käyttää visualisointikirjastoa sovelluksessaan.

**Bittikartta** Kuvien tallennusmuoto tietokoneessa. Jokainen kuvan pikseli omaa väritiedon.

**Datajoukko** Joukko datapisteitä.

**Datapiste** Visualisoitavan datan pienin yksikkö, joka voi koostua useasta erillisestä arvosta. Pisteiden arvot voidaan tulkita esimerkiksi kolmiulotteisen avaruuden koordinaateiksi.

**Entiteetti** Visualisoinnin interaktiivinen osa.

**Event** Ohjelmointitekhninen konstruktio, jonka avulla kirjasto kommunikoi sen sisäisistä tapahtumista kirjaston ulkopuolisille ohjelmiston komponenteille.

**Hierarkkinen klusterointi** Datajoukon jakaminen osajoukkoihin ja näiden edelleen jakaminen osiin tietyn yhtäläisen ominaisuuden perusteella.

**Hierarkkinen puu ja lämpökartta** Visualisointi, jolla kuvataan hierarkkisesti klusteroitua dataa kahdessa ulottuvuudessa värityksen avulla. Tätä käytetään esimerkiksi geenianalyyseissa.

**Interaktiivisuus** Interaktiokäyttäjän mahdollisuus vaikuttaa visualisoinnin ulkonäköön.

**Interaktiokäyttäjä** Arkkitehtuurikäyttäjän tekemän sovelluksen käyttäjä.

**JFreeChart** Avoimen lähdekoodin visualisointikirjasto, jolla voidaan piirtää tavallisimpia graafisia esityksiä.

**JPanel** Swing-kirjaston käyttöliittymäkomponentti, johon sijoitetaan muita käyttöliittymäkomponentteja.

**Kirjasto** Tässä dokumentissa sanaa kirjasto käytetään toisinaan viittaamaan tässä toteutettaviin ohjelmakomponentteihin. Tosiasiassa komponentit ovat ainoastaan JFreeChart-kirjaston laajennus.

**Klusteri** Datapisteiden joukko, jonka alkiot muistuttavat toisiaan.

**Kolmiulotteinen hajontakuvi** Kolmiulotteisen pistejoukon projektio kaksiulotteiselle tasolle.

**Kontekstivalikko** Valitun käyttöliittymäkomponentin ja sen osan mukaan muokkautuva valikko.

**MVC-arkkitehtuurimalli** on yleisesti käytetty tapa jakaa sovellus kolmeen erilliseen komponenttiin. Mallikomponentti toimii sovelluksen datavarastona, näkymäkomponentti esittää tiedon graafisesti tai muuten käyttäjälle, ja ohjainkomponentti kontrolloi näitä.

**Kuuntelija** Ohjelmistokomponentti, joka on rekisteröitynyt vastaanottamaan eventejä toiselta komponentilta.

**Pikseli** Näyttö- tai tulostuslaitteen erottelukyvyn yksikkö.

**Solu** SOM-kartan solu, yksittäinen datapiste.

**SOM-kartta** Self-organizing map. Neuroverkkoihin perustuva oppimisalgoritmi. Algoritmia käytetään kaksiulotteisten visualisointien tuottamiseen moniulotteisesta datasta.

**Suljettu klusteri** Hierarkkisen puun visualisoinnissa piilotettu alipuu.

**Swing** Javan graafisten käyttöliittymien luontiin tarkoitettu luokkakirjasto.

**Työkaluvihje** Tekstidialogi, joka kertoo jotain hyödyllistä tietoa hiirisoittimen osoittamasta käyttöliittymäkomponentista.

**Visualisointi** Annetusta datasta tietyllä visualisointimenetelmällä tuotettu kuva.

**Visualisointikokoelma** Yhteen JPanel-käyttöliittymäkomponenttiin liitettyjen visualisointien joukko.

**Visualisointimenetelmä** Algoritmi, jolla numeerinen  $n$ -ulotteinen data muutetaan kaksiulotteiseksi kuvaksi.

### 3 Lopputuotteen esittely

Viski-projektin tuottamat kaksi visualisointia sekä useamman visualisoinnin yhdistämisen mahdollistava komponentti ovat osa suurempaa JFreeChart-kirjastoa, joka on vapaa visualisointikirjasto erilaisten tieteellisten visualisointien ja kaavioiden piirtämistä varten.

Toteutimme kaksi visualisointia: SOM-kartan sekä hierarkkisen puun ja lämpökartan. Vaatimusdokumentissa oli lisäksi määritelty kolmaskin visualisointi — kolmiulotteinen hajontakuvio — mutta aikataulusyistä tätä ei ehditty toteuttaa. Näiden lisäksi projektimme muokkasi JFreeChart-kirjastoa lisäosilla niin, että useamman visualisoinnin yhdistäminen yhtäaikaista tulostusta ja tallennusta varten tuli mahdolliseksi.

#### 3.1 Erot vaatimusmäärittelyyn ja suunnitteluun

Suurin ero suhteessa vaatimusdokumenttiin oli kolmiulotteisen hajontakuvion visualisoinnin jättäminen suunnitteluvaiheessa pois projektista. Tähän päädyttiin, kun ymmärrettiin visualisointien toteuttamisen suuritoisuus, varsinkin, kun palautettavan tuotteen oli oltava mahdollisimman laadukkaasti toteutettua. Olimme kuitenkin toteuttaneet tiettyjä hajontakuvion ominaisuuksia teknologiademoon, jonka asiakkaan pyynnöstä palautimme lopullisen työn mukana, vaikka se ei olekaan osa lopputuotetta.

SOM-kartta pysyi miltei täysin uskollisena sekä vaatimus- että suunnitteludokumentille visualisoinnin suoraviivaisuuden vuoksi. Lisäksi toteutus ja suunnittelu tapahtuivat melkein samanaikaisesti, jolloin pystyimme huijaamaan vesiputousmallia ja päivittämään suunnitelmaa ennen sen jäädytystä.

Hierarkkinen puu ja lämpökartta -visualisointi ei sekään kokenut muutoksia vaatimusdokumenttiin nähden. Ainoastaan muutama alimman prioriteetin sekä yksi ylemmän prioriteetin vaatimus jäivät toteuttamatta. Syynä ykkösprioriteetin vaatimuksen toteuttamattomuuteen olivat JFreeChart-kirjastossa ilmenneet kankeudet käsitellä käyttöliittymäelementtejä, tässä tapauksessa kontekstivalikoita, jotka on valitettavasti ohjelmoitu muuttamattomaksi osaksi keskeisiä luokkia. Suunnittelu ei onnistunut yhtä tarkasti, vaan luokkia ja metodeja lisättiin toteutusvaiheessa useita. Nämä kaikki on dokumentoitu muutosdokumentissa.

Visualisointikokoelma toteutui myös hyvin suhteessa vaatimusdokumenttiin, mutta tähän oli enemmänkin syynä epämääräinen vaatimusmäärittely kokoelman osalta. Vaatimus, että visualisoinnit voisi yhdistää JPanel-komponenteiksi, ei toteutunut, mutta tämä johtui JFreeChart-kirjastosta, jota ei ole suunniteltu ottamaan huomioon useampaa kuin yhtä visualisointia kerrallaan. Tätä emme tiedäneet vaatimusmäärittelyvaiheessa. Suunnittelua vastaavaksi visualisointikokoelma kuitenkin tuli, koska jouduimme perehtymään juurta jaksuen JFreeChart-kirjastoon. Ratkaisu jäi hieman keinotekoisen oloiseksi, mutta tämänhetkinen JFreeChart-kirjasto ei yksinkertaisesti pysty parempaan ilman merkittävää uudelleenkirjoitusta.

Kaiken kaikkiaan vaatimusmäärittely onnistui sekä meidän että asiakkaan mielestä erittäin hyvin, kun taas suunnittelu aiheutti meille jonkin verran hankaluuksia lähinnä koke-

mattomuutemme vuoksi.

## 4 Dokumenttianalyysi

Viski-ryhmä tuotti projektin aikana kuusi dokumenttia tämän dokumentin lisäksi. Jokaiseen dokumenttiin nimettiin vastuuhenkilö, mutta ketään ei jätetty yksin minkään dokumentin kanssa, vaan dokumentit kirjoitettiin joko koko ryhmän voimin tai sitten puoliryhmissä. Tapana oli lisäksi oikaisulukea erityisesti ulkoiset dokumentit kokouksissa ennen jäädytystä.

- **Projektisuunnitelma** laadittiin heti projektin ensimmäisellä viikolla, eikä siihen juurikaan puututtu enää sen jälkeen. Suunnitelman suurinta antia oli aikataulu, mutta muuten se on nopeasti tehty, mitä todistaa omalta osaltaan se, että kaikista samaan aikaan olleista projekteista viski käytti tähän vaiheeseen vähiten työtunteja.
- **Vaatusdokumenttia** ja vaatimusmäärittelyä tehtäessä lähdettiin aluksi hieman liian tarkalla tasolla ajattelemaan vaatimuksia ja myös kirjaamaan niitä. Tämä johti määrittelyn loppupuolella ylimääraisiin työtunteihin, kun suuri osa dokumentista jouduttiin kirjoittamaan uudelleen yleisemällä tasolla. Valmiista vaatimusdokumentista tuli kuitenkin (ehkä tästä syystä) niin hyvä, että myös asiakas ilmaisi tyytyväisyytensä siihen.
- **Suunnitteludokumenttia** varten oli tehty jo suuri työ vaatimusmäärittelyssä, mutta siitä huolimatta kohdattiin joitain ongelmia. Lähinnä tutustuminen JFreeChart-kirjastoon ennen koodin kirjoittamista oli hankalaa, eikä ko. ohjelmiston omituisuuksia tullut tässä vaiheessa esille. Tämä johti suunnittelun osittaiseen pitkittymiseen ja tiettyihin epämääräisyyksiin. Suunnitteludokumenttia täydentää samaan tapaan kirjoitettu muutosdokumentti, johon on lisätty muutokset alkuperäisestä suunnittelusta.
- **Testaussuunnitelmaa** kirjoitettiin koko toteutusvaiheen ajan, ja lopulta siitä tuli lyhin dokumenttimme. Testaus oli kaikille alunperin melko epämääräinen asia, eikä myöhään järjestetty testauskoulutus helpottanut asiaa. Ohjaajan avulla kuitenkin määrittelimme konkreettiset kriteerit testausta varten ja sovelsimme niitä käytäntöön.
- **Käyttöohje** esittelee interaktiokäyttäjän käyttömahdollisuudet, joita ei ole kovin paljon projektin luonteen vuoksi.
- **Ylläpitodokumenttiin** on kirjoitettu arkkitehtuurikäyttäjän käyttöohje eli koodin käytön kannalta keskeisimmät esimerkit ja käytötavat. Näiden lisäksi olemme analysoineet koodin virheitä ja listanneet parannusehdotuksia.

Kaiken kaikkiaan projektin työstämiseen kuului paljon dokumenttien kirjoittamisen harjoittelua, ja dokumentteja hiottiin pitkään varsinaisen toteutuksen jäädessä taka-alalle —



myös työtunneissa mitattuna. Dokumentit syntyivät kuitenkin ilman suuria ongelmia ryhmästä löytyneen LaTeX-osaajan ansiosta.

## 5 Projektin päättöanalyysi

Tässä luvussa käydään lyhyesti läpi projektin eteneminen.

### 5.1 Työvaiheet

Työvaiheiden oli projektin aikataulun mukaan tarkoitus limittyä hieman keskenään. Jokainen työvaihe kuitenkin venyi aina niin, ettemme käytännössä ehtineet aloittaa seuraavaa työvaihetta ennen kuin olimme jäädyttäneet ja joissain tapauksissa hyväksyttäneet edellisen työvaiheen tuotoksen. Tällä ei ollut käytännön kannalta merkitystä, sillä jokainen työvaihe saatiin aikataulun mukaisena päivänä jäädytetyksi.

Vaatusmäärittelyyn käytimme selvästi liikaa aikaa, mikä johtui osaltaan alun harha-askeleista. Pysyimme vaatimusmäärittelyssä aikataulussa, joten vika oli enemmänkin aikataulutuksessa. Toisaalta testaukseen oli varattu aivan liian vähän aikaa, eikä testauskoulutuksen järjestäminen testausvaiheen alkamisen jälkeen auttanut asiaa.

Jokaisessa projektissa työvaiheet elävät suhteessa aikatauluun, eikä projektiryhmä saisi-kaan pitää liian tiukasti määräajoista kiinni. Tärkeintä on, että tarpeellisiin päivämääriin ehditään. Niitä oli tässä projektissa tasan kaksi: vaatimusmäärittelyn jäädytyksen takaraja ja palautuksen takaraja, ja molempiin ehdimme.

### 5.2 Toteutuneet riskit

Riskeistä toteutui projektin kannalta vain kaksi: suunnittelun epäonnistuminen ja jossain määrin työn määrän aliarvioiminen. Suunnittelun epäonnistuminen johtui ryhmän kokemattomuudesta. Se ei kuitenkaan epäonnistunut täysin, vaan ohjelman runko ja keskeiset osat saatiin suunniteltua hyvin. Lisäykset ja muutokset dokumentoitiin riskianalyysin toimenpide-kohdan mukaisesti muutosdokumenttiin.

Työn määrää aliarvointiin jonkin verran, mutta tämä huomattiin jo melko varhaisessa vaiheessa — käytännössä suunnittelun alettua. Tällöin ryhdyttiin heti käyttämään riskianalyysissa esitettyä toimenpidettä, ja suunniteltiin ja toteutettiin ensimmäisen prioriteetin vaatimukset. Näin ollen yksi vaatimusmäärittelyn visualisoinneista jäi projektilta toteuttamatta.

### 5.3 Ryhmäläisten kommentit

Tässä kappaleessa jokainen ryhmäläinen on saanut purkaa omia tuntojaan liittyen projektiin. Projektilaiset ovat aakkosjärjestyksessä.

### **Esa Elovaara**

Projekti oli positiivinen yllätys. Ystäviltä kuulemieni ohtu-projekti tarinoiden perusteella olin aika pessimistinen enkä odottanut paljoa. Ryhmä toimi hyvin ja kaikki ryhmän jäsenet olivat motivoituneita ja tilanteen tasalla. Koin projektin hyvin dokumentaatio intensiiviseksi, mutta se kai oli tarkoituskin.

### **Suvi Hiltunen**

Mielestäni ryhmä toimi hyvin. Yhteistyö oli vaivatonta, ja tuotti myös tuloksia. Ongelmien ratkaisemisessa sai hyvin tukea muilta ryhmäläisiltä. Useampaan kertaan jäimme kuitenkin kiinni turhankin tarkkaan yksityiskohtien viilaamiseen, mikä oli ehkä kurssin vaatimusten kannalta oleellista, mutta projektin edistymisen kannalta hieman haitallista. Aikaa olisi ollut mukavaa käyttää enemmän varsinaisen työn suunnitteluun, toteuttamiseen ja testaukseen, kun nyt se kului paljolti dokumenttien suunnitteluun ja kirjoittamiseen. Kokonaisuutena olen projektin antiin tyytyväinen. Mielestäni kurssin pointti olikin opetella ohjelmistotuotannon menetelmiä. Seuraavassa projektissa niitä osaa jo käyttääkin.

### **Tomi Jylhä-Ollila**

Omaksuin ”hienosääntäjän” roolin ryhmässä, jonka jäsenistä tuntui muodostuvan hyvin yhteensopiva kokonaisuus. Tässä ryhmässä työskentely ei ollut koskaan pakonomaista puurtamista. Useimmista prosessin vaiheista suoriuduttiin onnistuneesti, mutta olisin toivonut saavani paremman käsityksen toimivasta suunnitteluvaiheesta. Yleisesti ottaen tehty työ ja saavutettu kokemus jättivät kuitenkin hyvän maun.

Kirjojen lukeminen ja pienet irralliset harjoitukset eivät ole antaneet minulle vakuuttavaa kuvaa ohjelmistotuotannon eri menetelmien hyödyllisyydestä. Tämä projekti antoi.

### **Riku Louhimo**

Ryhmän henki oli hyvä, ja koin, että jokainen hoiti kohdalleen sattuneet tehtävät hyvin ja sovitun aikataulun mukaisesti. Ryhmässä vallitsi myös avoimen keskustelun ilmapiiri, ja kaikki olivat valmiita kompromisseihin, mikäli toisten näkemys niin vaati. Tämän lisäksi laitoksen projekteja yleensä vaivaava hirveä alustojen yms. säätäminen sujui leikiten, sillä jokaiselle työkalulle (mm. ant, svn, latex) löytyi kivuttomasti asian hallinnut ryhmäläinen, joka myös teki hyvää työtä. Aikataulu oli kireä ja saatoimme haukata ehkä hieman liian ison palasen toteutettavaksi näin lyhyessä ajassa. Selvisimme kuitenkin rajallisesta ajasta mielestäni kohtuullisesti.

## **Samuli Sairanen**

Mukavata oli. Projekti myös selvensi ohjelmistotuotannon todellisuutta. Ryhmätyöskentely toimi varsin mukavasti kun ryhmäläiset olivat sopivasti erilaisiin asioihin vihkiytyneitä. Ryhmädynamiikka muutenkin on jänskää. Prosessimalli ei tosin (vieläkään) vakuuttanut.

## **Juho Vuori**

Projekti meni pääosin mukavasti. Suurinpana haasteena oli ehkäpä JFreeChartiin tutustuminen, joka ei juuri onnistunut. Tämän vuoksi myös suunnitteluvaihe epäonnistui, ja lopulta koko tuotantoprosessi muuttui vesiputousmallista hyvin sekavaksi. Harmillista sinänsä, vesiputousmallia olisi ollut asian oppimisen kannalta erinomaisen kiinnostavaa seurata mahdollisimman sääntillisesti.

# **6 Yhteenveto**

Koodirivejä projektiin tuli lopulta noin 7200, mikä on noin 1400 enemmän kuin projektisuunnitelmassa arvioimme. Tässä luvussa ovat kuitenkin mukana Javadoc-kommentit. Kaiken kaikkiaan käytimme runsaasti työtunteja kokonaisuudessaan moniin muihin projekteihin verrattuna. Erityisesti suunnitteluun ja testaukseen käytimme kaksinkertaisesti aikaa verrattuna keskimääräiseen tämän kesän projektiin, mikä ehkä kertoo siitä, että muutkin olivat aliarvoineet testaukseen tarvittavan ajan. Istuimme myös eniten kokouksissa ja teimme vaatimusmäärittelyä hieman muita kauemmin.