

Projektisuunnitelma

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	5.6	Dokumentti lisätty SVN-repoon

Sisältö

1	Johdanto	1
2	Sanasto	2
3	Projektiorganisaatio	4
4	Riskianalyysi	4
5	Laitteisto- ja ohjelmistoympäristövaatimukset	6
6	Koko- ja kustannusarvio	6
7	Työn ositus	7
8	Aikataulu	7
9	Seuranta- ja raportointimenetelmät	7

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.

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 Projektioorganisaatio

Projektipäällikö Riku Louhimo

Vastuualueeseen kuuluvat asiakasyhteydenpito kokousten ulkopuolella, johtaa puhetta kokouksissa, vastaa projektisuunnitelmasta

Dokumenttivastaava Juho Vuori

Vastaa dokumenttien yhtenäisestä ulkoasusta, sekä opastaa muuta ryhmää LaTeXin käytössä

Koodivastaava Esa Elovaara

Vastaa koodin yhtenäisestä ulkoasusta ja rajapintojen yhtenäisyydestä

Testausvastaava Samuli Sairanen

Vastaa testauksen kattavuudesta

Vaatusmäärittelyvastaava Suvi Hiltunen

Vastaa yhtenäisestä vaatimusdokumentin rakenteesta

Suunnitteluvastaava Tomi Jylhä-Ollila

Vastaa yhtenäisistä suunnittelutason rajapinnoista ja suunnitteludokumentin yhdenmukaisesta rakenteesta, ylläpitää svn-ympäristöä, wikiä ja projektin sivustoa.

4 Riskianalyysi

Jokaisessa työprojektissa on vaihtelevat riskinsä eikä ohjelmistotuotantoprojekti ole poikkeus, vaikkakin siinä tietyt riskit ovat minimissään. Tähän listaamme keskeisemmät ja todennäköisimmät riskit ja miten projekti niihin varautuu.

Riskit on luokiteltu niiden todennäköisyyden ja vakavuuden mukaan. Riskin todennäköisyys on merkitty numeroina kunkin riskin kohdalle muodossa (*todennäköisyys/vakavuus*). Todennäköisyysarvot ovat

- 1 epätodennäköinen riski,
- 2 melko todennäköinen riski ja
- 3 todennäköinen riski.

Riskin vakavuusarvot ovat

- 1 pieni riski,
- 2 melko vakava riski ja
- 3 vakava riski.

Projektin jäsen lopettaa kesken.**(1/3)**

Ennaltaehkäisy: Kaikkien työ pitää olla raportoitua ja dokumentoitua, jolloin yksittäinen lopettaminen ei tee projektin siihenastista työtä turhaksi.

Toimenpiteet riskin toteutuessa: Karsitaan matalan prioriteetin vaatimuksista projektin loppuvaiheessa.

Projektin jäsen sairastuu**(3/1)**

Ennaltaehkäisy: Tätä ei voi ennaltaehkäistä projektin puitteissa.

Toimenpiteet riskin toteutuessa: Jokainen jäsen on toistensa vastuista ja tehtävistä niin hyvin selvillä, että pystyy lyhyen perehtymisen jälkeen tuuraamaan kipeää jäsentä.

JFreeChart osoittautuu myöhäisessä vaiheessa epäsovivaksi projektin tarpeisiin (2/2)

Ennaltaehkäisy: Huolellinen tutustuminen ja perehtyminen varmistaa, että tämä mahdollisuus suljetaan pois jo vaatimusmäärittelyvaiheessa.

Toimenpiteet riskin toteutuessa: Jos kuitenkin riski toteutuu vasta toteutusvaiheessa, on ratkaisuna vain kaiken kriittisimmän tilauksen osien (rajapinta) toteuttaminen vaatimusmäärittelyn priorisointien mukaisesti.

Suunnittelu epäonnistuu**(2/2)**

Ennaltaehkäisy: Hyvin tehty vaatimusmäärittely ehkäisee jo sinällään suunnittelun epäonnistumista. Lisäksi koko työryhmän panos on ratkaisevaa, jolloin suunnittelu ei jää yhden henkilön varaan.

Toimenpiteet riskin toteutuessa: Pyritään kirjoittamaan ohjelmisto valmiiksi ilman varsinaista suunnitteludokumenttia. Muutokset dokumentoidaan suunnitteludokumenttia vastaavaan muutosdokumenttiin.

Vaatimusmäärittely epäonnistuu**(2/2)**

Ennaltaehkäisy: Tarpeeksi kattavat kysymykset ja riittävän tiivis yhteydenpito asiakkaaseen ennaltaehkäisee.

Toimenpiteet riskin toteutuessa: Jos riski toteutuu vasta jälkikäteen, otetaan asian tiimoilta uudestaan yhteyttä asiakkaaseen ja koetetaan saada vaatimukset ajan tasalle.

Testaus epäonnistuu**(2/3)**

Ennaltaehkäisy: Suunnitellaan ja toteutetaan huolellisesti sekä varataan riittävästi aikaa

testaukselle.

Toimenpiteet riskin toteutuessa: Jos testaus joudutaan tekemään nopeasti, ryhmän täytyy sitoutua joihinkin mittoihin (esim. line coverage), jonka perusteella testausta tehdään.

Toteutus epäonnistuu

(2/3)

Ennaltaehkäisy: Suunnittelu, vaatimusmäärittely ja tutustuminen tulee hoitaa projektin loppuunviennin vaatimalla syvyydellä eli hyvin.

Toimenpiteet riskin toteutuessa: Palautetaan vain toimivat osat, ja dokumentoidaan virheet ja puutteellisuudet.

Työn määrän aliarvioiminen

(2/1)

Ennaltaehkäisy: Vaatimusmäärittelyn ja suunnittelun tekeminen hyvin.

Toimenpiteet riskin toteutuessa: Tiputetaan toteutusvaiheessa pois alhaisen prioriteetin vaatimuksia niin, että kuitenkin saadaan jokin järkevä kokonaisuus aikaan.

Ryhmäläinen ei tee sovittuja tehtäviään

(3/2)

Ennaltaehkäisy: Annetaan tehtäviä halukkuuden mukaan.

Toimenpiteet riskin toteutuessa: Ensin tuodaan asia esille ryhmän kesken. Jos tämä ei vaikuta, voidaan siirtyä ylemmälle tasolle esim. ohjelmistotuotantoprojektista vastaavalle Juha Tainalle päätettäväksi, voiko jäsen jatkaa projektia.

5 Laitteisto- ja ohjelmistoympäristövaatimukset

Projekti toteutetaan Tietojenkäsittelytieteen laitoksen laiteympäristössä, Java-kielellä (v1.3) ja Swing-käyttöliittymäkehystä käyttäen.

6 Koko- ja kustannusarvio

Kokoarvio lasketaan käyttäen funktionaalisten pisteiden laskentaa. Nämä voidaan sitten muuttaa suoraan koodiriveiksi, kun tiedetään toteutuksessa käytettävä ohjelmointikieli, joka tässä tapauksessa on Java v.1.3.

Pisteet (function points = FP) saadaan seuraavalla kaavalla:

$$FP = count - total * [0.65 + 0.01 * sum(Fj)]$$

Tässä kaavassa vakiot ovat kaavan kehittäjän tutkimia painotusarvoja. Muuttuja-attribuutit lasketaan vastaamalla listaan kysymyksiä eri osa-alueiden koos-

ta ja vaikeudesta ohjelmoitaessa. Nämä kysymykset löytyvät osoitteesta <http://www.cs.helsinki.fi/group/ohtu/resurssit/fp.html>. Viski-projektin osalta vastaukset eli lukuarvot näyttävät seuraavilta.

CT:

1. $3x4 = 12$ 2. $10x4 = 40$ 3. $17x4 = 68$ 4. $3x7 = 21$ 5. $4x10 = 40$

sum(Fj):

1. 0 2. 0 3. 2 4. 3 5. 5 6. 2 7. 1 8. 0 9. 3 10. 3 11. 5 12. 5 13. 3 14. 5

$summa = 2 + 3 + 5 + 2 + 1 + 3 + 3 + 5 + 5 + 3 + 5 = 37$

Lopuksi nämä lukuarvot syötetään kaavaan ja saadaan

$FP = count - total * [0.65 + 0.01 * sum(Fj)]$

$184,62 = 181 * [0.65 + 0.01 * 37]$

$FOC = 184,62 * 31 = 5723,22$

Tämän perusteella arvio viski-projektin koodimäärästä, jos kaikki vaatimukset toteutetaan on noin 6000 riviä koodia.

7 Työn ositus

Suunnitteluvaiheessa ryhmä jaetaan kahteen aliryhmään, jotka hoitavat alusta loppuun niille jaettujen vastualueiden suunnittelun ja, mikäli pystyvät, myös toteutuksen.

Jako on seuraava:

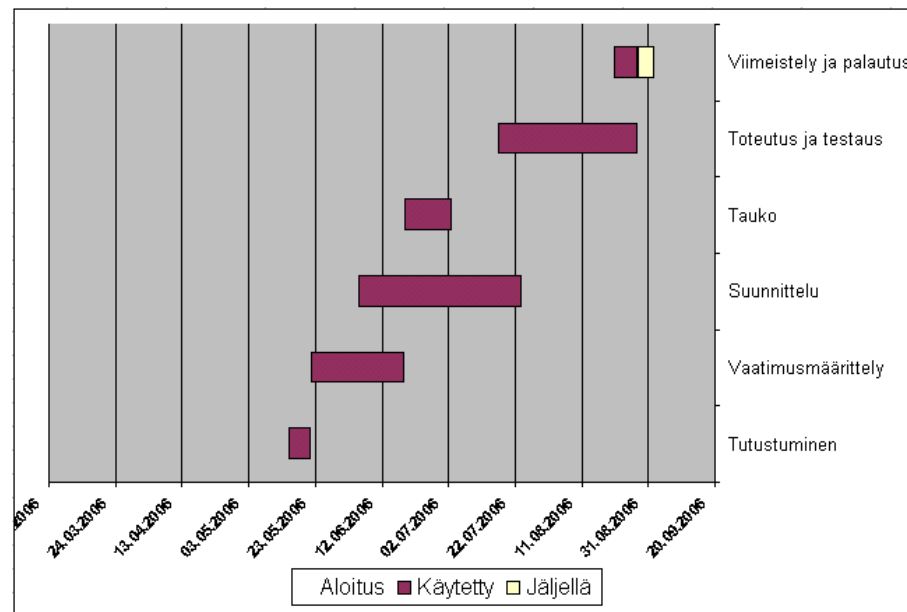
ryhmä 1: hierarkkinen puu ja lämpökartta ryhmä 2: loput

8 Aikataulu

pvm	viikot	toimet
15.5-21.5	20, 21	tutustuminen, projektisuunnitelman laadinta
22.5-18.6	21 – 24	vaatimusmäärittely (juhannus vklla 25!)
05.6-23.7	23 – 29	suunnittelu
19.6-02.7	25 – 26	tauko
17.7-27.8	29 – 34	toteutus ja testaus
	34	asiakasdemo (jossain kohtaa ko. viikkoa)
21.8-01.9	34 – 35	viimeistely ja palautus pe 1.9

9 Seuranta- ja raportointimenetelmät

Projekti kokoontuu kaksi kertaa viikossa seurantakokouksiin tk1:n tiloissa Exactumissa. Lisäksi yhteyttä pidetään sähköpostilistan, wikin ja irc-kanavan välityksellä. Tarkistus-



Kuva 1: Visualisointikirjaston arkkitehtuurin yleiskuva.

kokouksia tullaan projektin aikana järjestämään ainakin kolme, vaatimusmäärittelyn ja suunnittelun lopulla sekä koodauksen aikana. Lisäksi kerran viikossa palautetaan mieliin aikataulu ja projektin edistyminen.

sähköpostilista ohtuv06-viski-list@cs.helsinki.fi

irc #ohtu_viski @ IrcNet