

VERSTAS-PROJEKTI

Projektisuunnitelma

Antti Hakala

Tomi Laamanen

Paavo Nieminen

Jukka Toivanen

7.1.2003

Jyväskylän yliopisto

Tietotekniikan laitos

- Tekijät: Antti Hakala, Tomi Laamanen, Paavo Nieminen, Jukka Toivanen
- Yhteystiedot: verstas@korppi.jyu.fi, anthakal@cc.jyu.fi, laamanen@cc.jyu.fi, nieminen@cc.jyu.fi, jitoivan@cc.jyu.fi
- Projektitila: Agora C223.4
- Puhelin: 014-2604966
- Kotisivu: <http://kotka.it.jyu.fi/verstas/>
- Työn nimi: Verstas-sovellusprojektin projektisuunnitelma
- Työ: Tietotekniikan sovellusprojekti

Tiivistelmä:

Tämä dokumentti on Jyväskylän yliopistossa syksyllä 2002 toteutettavan Verstas-sovellusprojektin projektisuunnitelma. Suunnitelmassa esitellään projektin taustoja ja tavoitteita, kuvataan sovelluksen rakennetta ja arvioidaan sen toteuttamista ja testausta.

Avainsanat: sovellusprojekti, projekti, käyttöliittymä, mallinnusympäristö

Versiohistoria

Versio	Pvm	Tekijä	Kuvaus
0.1	07.10.2002	TL	Ensimmäinen raakaversio
0.2	12.10.2002	TL, JT	Virheitä korjattu, aikataulua parannettu, lisätty vastuualueita.
0.3	23.10.2002	TL, JT	Aikataulua ja riskit-lukua parannettu, vastuualueita jaettu lisää, lisätty viitteet, ulkoasua muotoiltu.
0.4	28.10.2002	TL, PN	Lisätty sisällysluettelo, termejä lisätty, lauserakenteita korjattu.
0.5	04.11.2002	TL	Tehty Kari Kärkkäisen ehdottamat korjaukset.

Tekijöiden lyhenteet:

- AH Antti Hakala
- TL Tomi Laamanen
- PN Paavo Nieminen
- JT Jukka Toivanen

Hyväksytty

Pvm Allekirjoitus

Pvm Allekirjoitus

Sisältö

1	Johdanto	1
1.1	Termit ja käsitteet	1
1.1.1	Sovellukset, kirjastot ja tiedostot	1
1.1.2	Ohjelmointikielet ja -ympäristöt	1
1.1.3	Mallinnusprosessi ja laskenta	2
1.1.4	Verstas	3
2	Taustat ja tavoitteet	4
2.1	Tilaaajan tarpeet	4
3	Resurssit ja organisaation toimenkuvat	5
4	Toteutussuunnitelma	6
4.1	Aikataulut vaiheittain	6
4.2	Tehtävät ja vastualueet	7
4.3	Riskit ja riskienhallinta	8
5	Ohjaussuunnitelma ja dokumentointi	10
5.1	Kokouskäytäntö	10
5.2	Projektin päättäminen	10
5.3	Dokumentointi	10
5.4	Ohjelmakoodin kommentoiminen ja muuttujien nimeäminen	11
6	Yhteenveto	12

1 Johdanto

Verstas-projekti suunnittelee ja toteuttaa Numerola Oy:lle yhtenäisen ja kokoavan käyttöliittymän mallinnus- ja simulointiohjelmistoille. Projekti toteutetaan Jyväskylän yliopiston tietotekniikan sovellusprojektina. Tämä suunnitelma kuvaa Verstas-projektin taustaa, aikataulua sekä projektin toteutukseen liittyviä asioita.

Luvussa 2 kerrotaan projektin taustoista ja tavoitteista sekä tilaajan, että projektin jäsen-ten kannalta. Luvussa 3 selvitetään projektille varatut resurssit ja organisaation toimenkuvat ts. mikä on kenenkin rooli projektissa ja mitkä ovat heidän taustansa. Luku 4 käsittelee projektin aikataulua, projektin jäsenten tehtävien ja vastualueiden jakautumista, riskejä ja niiden hallintaa sekä teknisiä ratkaisuja. Luvussa 5 kuvataan projektin kokouskäytäntö, projektin päättäminen, dokumentointi sekä ohjelmakoodin kommentoiminen ja muuttujien nimeäminen.

1.1 Termit ja käsitteet

1.1.1 Sovellukset, kirjastot ja tiedostot

Alisovellus: Numerolan toteuttama sovellusohjelma, jolla toteutetaan yksi mallinnusprosessin vaiheista.

Dmg: Alisovellus, joka muuttaa 2D-geometrian laskentaverkoksi.

Emacs: Laajennettava ja muokattava tekstieditori.

Exodus: Alisovellus, työkalu 3-ulotteisen laskentageometrian luomiseen.

Genesis: Alisovellus, työkalu 2-ulotteisen laskentageometrian luomiseen.

Makefile: Make-sovelluksen käyttämä tiedosto ohjelman lähdekoodin kääntämisen hallintaan.

Mbgen: Alisovellus, joka muuttaa 3D-geometrian laskentaverkoksi.

Numerin-ydin: Numerolan toteuttama FEM-aliohjelmakirjasto.

Plot: Alisovellus visualisointiin.

TeeKL, AutoUI: Alisovellus, työkalu graafisen käyttöliittymän automaattiseen tuottamiseen ja hallintaan.

Verkkogeneraattori: Alisovellus, joka tuottaa geometriasta laskentaverkon. Verstaan alisovellusten käytössä on useita eri tilanteisiin sopivia verkkogeneraattoreita, kuten Verkotin, Dmg, Mbgen.

1.1.2 Ohjelmointikielet ja -ympäristöt

Elisp: Emacs lisp, Emacsin laajentamiseen käytettävä ohjelmointikieli.

Fortran: Erityisesti laskennallisten ongelmien ratkaisuun käytetty ohjelmointikieli.

Kylix: C++/Object Pascal -kehitysympäristö Linux-käyttöjärjestelmälle.

Object Pascal: Olio-ohjelmointilaajennos Pascal-ohjelmointikieleen.

1.1.3 Mallinnusprosessi ja laskenta

FEM: Finite Element Method, äärellisten elementtien menetelmä, eräs numeerinen menetelmä osittaisdifferentiaaliyhtälöiden numeeriseksi ratkaisemiseksi.

Geometrian luonti: Mallinnusprosessin vaihe, jossa tutkittavan fysikaalisen järjestelmän ominaisuudet pyritään kuvaamaan tietokoneohjelman avulla. Tuloksena on ns. laskentageometria.

Laskenta: Mallinnusprosessin vaihe, jossa käännetty laskentaohjelma ajetaan syötteenään laskentaverkko sekä parametrit, tulosteena saadaan laskentatulokset.

(Laskenta)geometria: Tietokoneella luotu tutkittavan fysikaalisen kappaleen muodon ja rakenteen kuvaus, esitettynä joukosta yksinkertaisia käyriä tai pintoja muodostuvana approksimaationa.

(Laskenta)geometrian luonti: Mallinnusprosessin vaihe, jossa kuvataan tutkittavan fysikaalisen järjestelmän geometriset ominaisuudet. Tuloksena on laskentageometria.

(Laskenta)malli: Matemaattinen malli, jonka ilmenemismuoto on fortran-lähdekoodi.

Laskentaohjelma: Fortran-kääntäjällä laskentamallista käännetty ajettavissa oleva ohjelma.

Laskentatulos: Laskentaohjelman tuloste, matemaattisen mallin ratkaisu annetuilla lähtötiedoilla (geometria, parametrit), käytännössä ASCII-tekstinä esitettyä numeerista dataa.

Laskentaverkko: Verkkogeneraattorin tuloste, geometrian jako laskentaa varten osiin esim. kolmioihin tai tetraedreihin.

Laskentaverkon luonti: Mallinnusprosessin vaihe, jossa laskentageometriasta tuotetaan verkko-generaattorilla laskentaohjelman tarvitsemaan laskentaverkko.

Mallin toteutuksen etsintä: Valittua matemaattista mallia vastaavan ohjelman etsiminen tai luominen.

Mallin valinta: Mallinnusprosessin vaihe, jossa päätetään (yleensä kirjallisuuteen tai tutkimukseen nojaten) myöhemmissä vaiheissa käytettävä matemaattinen malli.

Mallinnusprosessi: Nykyaikaisen tuotekehityksen ja tutkimuksen osa-alue, jossa prototyypin rakentamisen ja kokeellisen mittaamisen sijasta tehdään kokeiluja matemaattisella mallilla, joka kuvaa järjestelmän toimintaa annetuissa olosuhteissa.

Optimointi: Simuloinnin (automaattinen) toistaminen jollakin kriteerillä geometriaa ja/tai parametreja muuttamalla, tavoitteena parempi lopputulos.

Parametrit: Laskentaohjelman syöte, esim. tutkittavan ilmiön fysikaaliset ominaisuudet.

Visualisointi: Laskentatulosten esitys ihmisen ymmärtämässä muodossa (käytännössä graafisesti erilaisin tavoin), eräs mallinnusprosessin vaihe.

1.1.4 Versta

Mallinnusympäristö: Kokonaisuus, jossa kaikki mallinnuksen vaiheet voidaan tehdä yhtenäisen käyttöliittymän avulla hallittavilla työkaluilla.

Numerrin-Versta: Versta, Versta-ympäristö. Mallinnusympäristö, jossa kaikki mallinnusprosessin vaiheet voidaan toteuttaa käyttäen Numerolan työkaluja ja Numerrin-ydintä.

Projekti: Kokonaisuus, joka sisältää kaikki tiettyyn simulointitehtävään liittyvät tiedot (s.o. mallit, geometriat, parametrit, tulokset, raportit jne).

Raportti: Tutkimus-, päivitys- tai muu raportti, esim. pdf-muodossa.

Tietokanta: Tässä projektissa hakemistorakenne sisältäen uudelleen hyödynnettäviä mallinnukseen käytettäviä tiedostoja.

2 Taustat ja tavoitteet

Tässä luvussa perehdytään projektin taustoihin ja tavoitteisiin.

2.1 Tilaajan tarpeet

Numerola Oy on vuonna 1998 perustettu yritys, joka tekee asiakassuuntautunutta tutkimusta ja ohjelmistokehitystä laskennallisen mallinnuksen ja optimoinnin alalla. Meneillään on myös oma Numerrin-tuotekehitysprojekti, jonka tarkoituksena on kehittää mallinnukseen ja mallipohjaiseen optimointiin tarkoitettujen simulointiohjelmistojen kehitysympäristö.

Verstas-projektin tarkoituksena on toteuttaa Numerola Oy:lle käyttöliittymä Numerrin-sovelluksen alisovelluksille.

2.2 Tavoitteet

Sovelluksen avulla tulee pystyä hallitsemaan mallinnusprosessissa käytettäviä Numerrin-sovelluksen alisovelluksia yhtenäisen käyttöliittymän avulla siten, että mallinnusprosessin vaiheita automatisoidaan niin paljon kuin mahdollista. Käyttäjältä ei tulisi myöskään vaatia suurta käyttöjärjestelmä- tai ohjelmointituntemusta, vaan sellaiset toiminnot pyritään piilottamaan graafisen käyttöliittymän taakse. Siirrettävyys Windows-ympäristöön on varmistettava. Sovelluksen tulee olla kaupallisesti houkutteleva, ja käyttöliittymän tulee mukautua käyttäjän tarpeisiin.

Projektin opetuksellisenä tavoitteena on antaa ryhmän jäsenille tuntumaa ja kokemusta projekti- ja ryhmätyöskentelystä. Ryhmän jäsenet oppivat arvioimaan eri työvaiheiden työmäärää ja aikatauluja. Tärkeänä osana on kommunikointi projektin osapuolten välillä, oppia tilaajan ja työntekijän roolit sekä yhteistyön merkitys. Projektin jäsenet saavat kokemusta erilaisten dokumenttien ja kaavioiden laatimisesta. Projektissa opitaan kokouskäytäntöä ja vastuuta toimimalla vuorotellen puheenjohtajana ja sihteerinä. Tehtävät pyritään jakamaan tasaisesti koko ryhmän kesken siten, että jokainen pääsee tutustumaan projektin eri vaiheisiin ja tehtäviin.

3 Resurssit ja organisaation toimenkuvat

Projektin käytössä on lukittava työhuone Agora C223.4, jossa jokaisella on oma työasema. Projekttilaisilla on käytössään tulostin, toimistotarvikkeita, kopiokone sekä vanhempia, testikäyttöön tarkoitettuja, tietokoneita. Projektin käytössä on lisäksi kokoustila Agora C223.1.

Sovelluksen ohjelmointikielenä on Object Pascal ja ohjelmointiympäristönä Kylix 3.0 Enterprise, Emacs-moduuli toteutetaan Elisp:llä. Sovellus toteutetaan Linux-ympäristössä, testaus- ja suoritetaan myös Windows-ympäristössä Delphi 6.0 -ohjelmistolla. Dokumentointi toteutetaan $\text{T}_{\text{E}}\text{X}$ -formaattiin käyttäen $\text{L}_{\text{Y}}\text{X}$ -ohjelmistoa. Tarkemmin teknisistä ratkaisuista on kerrottu Verstas-sovelluksen vaatimusmäärittelyssä.

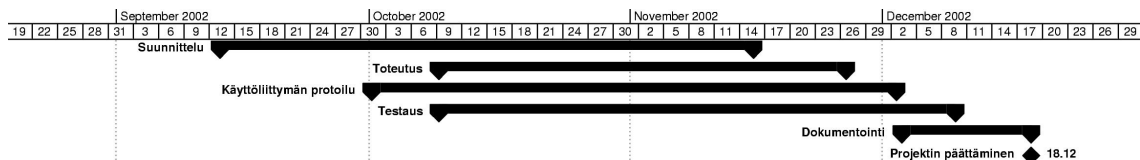
Sovellusprojektin toteuttava ryhmä koostuu neljästä opiskelijasta. Antti Hakala on neljännen vuoden tietotekniikan opiskelija, suuntautunut ohjelmistotekniikkaan. Tomi Laamanen on neljännen vuoden tietotekniikan opiskelija tietoliikenteen linjalla. Paavo Nieminen opiskelee tietotekniikkaa kolmatta vuotta, linjanaan tieteellinen laskenta. Jukka Toivanen on kolmannen vuoden tietotekniikan opiskelija, suuntautunut tieteelliseen laskentaan.

Tietotekniikan laitoksen puolelta projektia valvovat ja ohjaavat yliassistentti Kari Kärkkäinen, joka toimii projektin vastaavana ohjaajana, ja teknisenä ohjaajana fil. yo Ville Tirronen.

Numerola Oy:n puolelta tilaajan edustajina toimivat sovellusasiatuntija FT Raino Mäkinen, johtaja FT Pasi Tarvainen ja sovellusasiatuntija FM Eeva-Kaisa Rouhiainen.

4 Toteutussuunnitelma

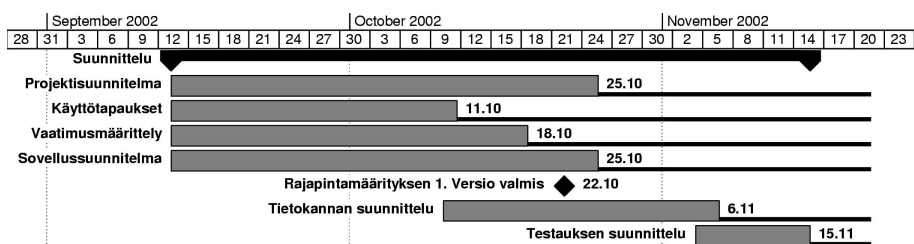
Verstas-projekti toteutetaan 13.09.2002-20.12.2002 välisenä aikana. Projekti päättyy 17.12.2002 pidettävään loppuesittelyyn. Projekti jaetaan viiteen päävaiheeseen ja ne toteutetaan kuvan 1 mukaisesti. Katselmointi järjestetään aina vaiheen päättymistä seuraavassa projektipalaverissa, jos niin on sovittu. Katselmoinnissa tarkistetaan kunkin vaiheen tuotos, jonka tuloksien tulee olla saatavilla kolmea päivää ennen katselmointia.



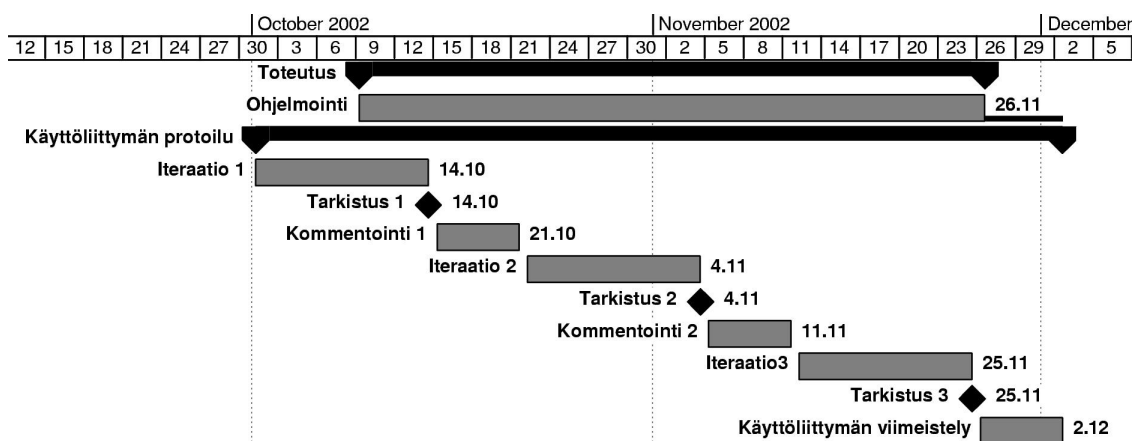
Kuva 1: Projektin aikataulu

4.1 Aikataulut vaiheittain

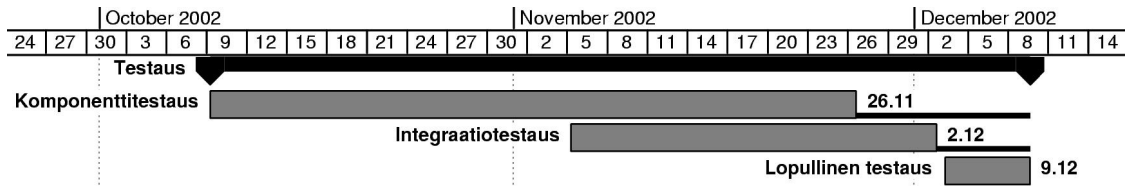
Kuvissa 2-5 on kuvattu erikseen Gantt-kaavioiden edellä jaettujen vaiheiden aikataulut.



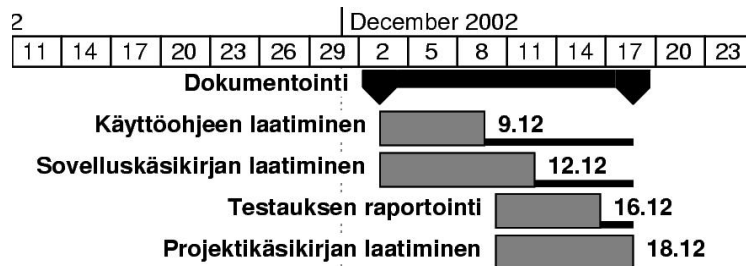
Kuva 2: Suunnittelun aikataulu



Kuva 3: Toteutuksen ja käyttöliittymän prototyylin aikataulu



Kuva 4: Testauksen aikataulu



Kuva 5: Dokumentoinnin aikataulu

4.2 Tehtävät ja vastualueet

Projektiryhmässä jokainen ryhmän jäsen osallistuu jokaiseen projektivaiheeseen: dokumentointiin, suunnitteluun, toteutukseen ja testaukseen. Vastuualueita jaettaessa on otettu huomioon ryhmän jäsenten toivomukset.

Projektipäällikkö	Aika
PN	19.09.-13.10.2002
JT	14.10.-03.11.2002
AH	04.11.-25.11.2002
TL	26.11.-

Taulukko 1: Projektipäällikkyydet

Vastuualue	Vastuuhenkilö
Käyttötapaukset	AH, TL, PN, JT
Projektsuunnitelma	TL
Sovellussuunnitelma	AH, TL, PN, JT
Vaatusmäärittely	AH
Projekti sopimus	PN
Vaitiolosopimus	JT

Taulukko 2: Suunnittelun vastualueiden jakautuminen

Vastuualue	Vastuuhenkilö
Elisp	AH
Fortran-kääntäminen, laskenta	TL
Käyttöliittymä	JT
ApplicationManager	AH, JT
ProjectManager	TL, PN
DBManager	AH

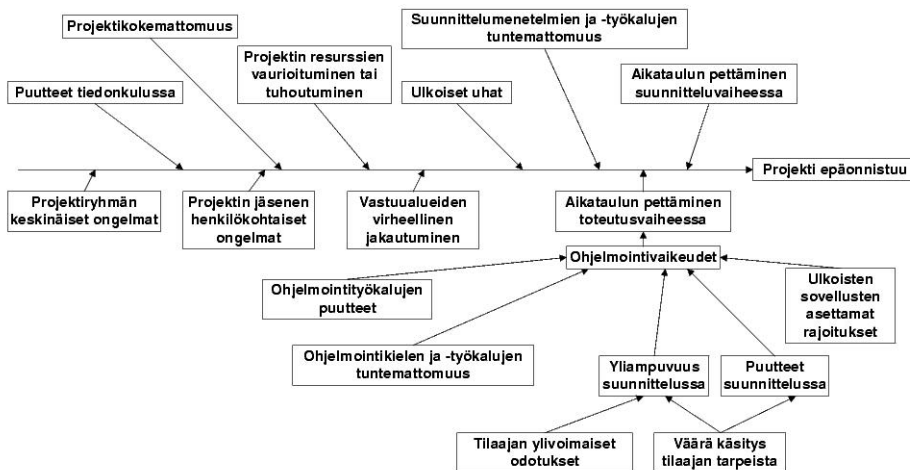
Taulukko 3: Toteutuksen vastuualueiden pääpiirteittäinen jakautuminen

Vastuualue	Vastuuhenkilö
Testaus	AH, TL, PN, JT
Käyttöohje	AH, TL
Testaussuunnitelma	AH, TL, PN, JT
Projektikäsikirja	PN
Sovelluskäsikirja	JT
Testausraportti	AH, TL, PN, JT

Taulukko 4: Testauksen ja dokumentoinnin vastuualueiden jakautuminen

4.3 Riskit ja riskienhallinta

Kuvassa 6 on kartoitettu projektin toteutukseen ja aikatauluun mahdollisesti liittyviä riskitekijöitä. Riskitekijät on arvioitu suurimmasta pienimpään asteikolla 1-3.



Kuva 6: Projektin liittyviä riskitekijöitä

Projektikokemattomuus (2) tarkoittaa sitä, ettei projektiryhmän jäsenillä ole juurikaan kokemusta projektityöskentelystä ja sen käytännöistä. Tämä voi aiheuttaa mm. sen, etteivät pro-

jektiryhmän jäsenet ole perillä kunkin projektin vaiheen merkityksestä ja tarkoituksista. Riskiä voi ennaltaehkäistä riittävällä ohjauksella ja esimerkkien saatavuudella.

Aikataulun pettäminen (1) tarkoittaa käytännössä jonkun projektin vaiheen viivästymistä, joka mahdollisesti heijastuu toiseen vaiheeseen.

Ohjelmointityökalujen ja ohjelmointikielen tuntemattomuus (2) johtaa vaikeuksiin ohjelmointivaiheessa ja mahdollisesti edelleen aikataulun pettämiseen. Tämä tulee huomioida toteutuksen aikataulun laadinnassa.

Ohjelmointivaikeuksilla (2) tarkoitetaan esim. takertumista pitkäksi aikaa jonkin osaongelman ratkaisuun, mistä seuraa mahdollisesti viivästymistä. Riskiä voidaan ennaltaehkäistä riittävällä ennakkotutustumisella aihealueeseen ja hakemalla ongelmaan ajoissa apua esim. tekniseltä ohjaajalta.

Ohjelmointityökalujen (Kylis) puutteet (3) tarkoittavat sitä, että ohjelmointiin käytettyjen sovellusten ominaisuudet eivät ole riittävät tai että sovellusten lisenssit asettavat rajoituksia, mikä aiheuttaa hankaluuksia toteutusvaiheessa. Ohjelmointityökalujen soveltuvuus projektiin tulisi selvittää mahdollisimman tarkasti ennen toteutusvaihetta.

Ulkoisten sovellusten (alisovellukset, editori) asettamat rajoitukset (1) voivat aiheuttaa sen, ettei sovelluksesta tule riittävän yhtenäinen ja asetettujen vaatimusten mukainen. Tämä tulee ottaa huomioon erityisesti vaatimusmäärittelyssä.

Ulkoisilla uhilla (3) tarkoitetaan projektista riippumatonta, esim. projektien resursseja vahingoittavaa tapahtumaa, kuten tulipaloa.

Puutteet tiedonkulussa (3) tarkoittaa tiedotukseen liittyviä ongelmia projektiin liittyvien osapuolien välillä.

Projektin resurssien vaurioitumisella tai tuhoutumisella (3) tarkoitetaan esim. projektin kannalta välttämättömän laitteen vaurioitumista ja tästä projektille aiheutuvia ongelmia, joiden laajuutta voidaan merkittävästi pienentää ennakoivalla toiminnalla, kuten varmuuskopioinnilla.

Suunnittelutyökalujen ja -menetelmien tuntemattomuus (2) aiheuttaa ongelmia ja viivästymisiä suunnitteluvaiheessa ja voi johtaa suunnitelmien yliampuvuuteen tai puutteisiin.

Yliampuvuus ja puutteet suunnittelussa (2) ovat seurauksena vääristä käsityksistä tilaajan tarpeista, mikä aiheuttaa ongelmia toteutusvaiheessa. Riskiä voidaan ennaltaehkäistä riittäväillä tarkennuksilla suunnitteluvaiheessa.

Tilaajan ylivoimaisilla odotuksilla (2) tarkoitetaan suunnitteluvaiheessa tiedostettuja sovelluksen ominaisuusvaatimuksia, jotka osoittautuvat vasta ohjelmointivaiheessa hankalaksi tai mahdottomaksi toteuttaa. Tämä tulee ottaa huomioon sovellukselle asetettavissa vaatimuksissa siten, että ominaisuuksista voidaan tarvittaessa tinkiä.

Projektiryhmän keskinäiset ja projektin jäsenen henkilökohtaiset ongelmat (2) vaikeuttavat projektityöskentelyä kokonaisuudessaan.

Vastuualueiden virheellisellä jakautumisella (3) tarkoitetaan tehtävien epätasaista tai muutoin virheellistä jakaantumista projektiryhmän kesken, mikä voi johtaa viivästymisiin ja projektiryhmän sisäisiin ongelmiin. Tehtävät tulisikin pyrkiä jakamaan projektiryhmän kesken siten, että kukin saa kykyjään ja aikatauluun vastaavia tehtäviä.

5 Ohjaussuunnitelma ja dokumentointi

Tässä luvussa on määritelty Verstas-projektin ohjaus- ja dokumentointikäytännöt.

5.1 Kokouskäytäntö

Verstas-projektin aloituspalaveri pidettiin torstaina 19.09.2002 Numerola Oy:n kokoustilassa. Projektin kuluessa järjestetään joka tiistai klo 14.15 viikkopalaveri Agora C223.1 kokoustilassa tai Numerola Oy:n kokoustilassa, jossa ovat paikalla projektiohjaajat, projektiryhmä ja tilaajan edustajat. Viikkopalaverissa kukin projektiryhmäläinen esittää kuluneella viikolla tekemänsä työn ja työmäärän projektin ohjaajille. Viikkopalaverissa käsitellään projektista heränneitä kysymyksiä. Ennen viikkopalaveria muodostetaan lista palaverissa käsiteltävistä asioista ja lähetetään se projektin postituslistalle. Viikkopalaverin kokoonkutsujana toimii projektiryhmä. Kokouksen puheenjohtajan ja sihteerin sekä projektipäällikön vuoro kiertää projektiryhmäläisten kesken.

5.2 Projektin päättäminen

Viimeinen projektipalaveri järjestetään dokumenttien ja sovelluksen hyväksymisen jälkeen. Palaverissa luovutetaan dokumentointi ja sovellus tilaajalle ja tietotekniikan laitokselle. Numerola Oy:n salaiseksi katsoma materiaali jätetään pois tietotekniikan laitokselle jäävästä dokumentaatiosta. Lisäksi suoritetaan projektin yhteenveto, sovitaan mahdollisesta jatkokehittämisestä sekä sovitaan yleisistä projektin päättymiseen liittyvistä asioista.

Projekti katsotaan päättyneeksi ja hyväksytyksi, kun ohjaajat ja tilaajan edustajat ovat hyväksyneet sekä toteutetun sovelluksen että dokumentoinnin. Tekijänoikeudet sekä muut aiheettomat oikeudet siirtyvät tilaajalle projektimaksun suorituksen yhteydessä. Projektiryhmän jäsenille jää rinnakkaisia tekijänoikeuksia, jotka on mainittu projektisopimuksessa.

5.3 Dokumentointi

Dokumenttipohjina käytetään projektiryhmän luomia dokumenttipohjia joihin on sovellettu tietotekniikan laitoksen gradu-pohjaa [1]. Verstas-projektissa toteutetaan seuraavat dokumentit:

- Projektisuunnitelma
- Käyttötapaukset
- Vaatimusmäärittely
- Sovellussuunnitelma
- Testaussuunnitelma
- Testausraportti
- Käyttö- ja asennusohje
- Sovelluskäsikirja

- Projektikäsikirja
- Liitteet / täydennykset em. dokumentteihin

Projektsuunnitelmassa kuvataan Verstas-projektin taustaa, aikataulua sekä projektin toteutukseen liittyviä asioita.

Vaatimusmäärittelyssä määritellään tuotettavan sovelluksen ominaisuudet ja toiminnallisuudet sekä sen ympäristölle asettamat vaatimukset mahdollisimman tarkasti, käyttötapaukset-dokumentti liitetään vaatimusmäärittelyyn.

Sovellussuunnitelmassa kuvataan toteutettavan sovelluksen arkkitehtuuri ja käyttöliittymä, *sovelluskäsikirjassa* sovellussuunnitelman toteutuminen.

Testaussuunnitelmassa esitetään toteutettavan sovelluksen testauksen toteutusstrategiat, testitapaukset ja testausympäristö, *testausraportissa* testauksen tulokset.

Projektikäsikirjassa raportoidaan projektin toteutus: kuvataan projektilta vaadittujen asioiden toteutumista sekä tekijöiden mielipiteitä ja kokemuksia.

Käyttö- ja asennusohje sisältää toteutettavan sovelluksen asennusohjeet ja käyttöliittymän käyttöohjeet.

5.4 Ohjelmakoodin kommentoiminen ja muuttujien nimeäminen

Ohjelmakoodi kommentoidaan englanninkielisenä. Koodia kirjoitettaessa pyritään kirjoittamaan aliohjelman toiminta ensin selventävänä kommenttina, jonka jälkeen kirjoitetaan kyseinen koodi. Jokainen projektiryhmän jäsen vastaa ohjelmakoodin selkeydestä ja tulkattavuudesta. Lähdekooditiedostossa tulee olla sen tarkoitus, projekti, haltija, ohjelmaversio ja päiväys. Lisäksi kunkin aliohjelman toiminta, parametrit ja paluuarvot on selitettävä lyhyesti. Muuttujien nimeämisessä käytetään taulukon 5 mukaista käytäntöä. Metodien nimistä tulee ilmetä sen toiminta. Parametrien nimet pyritään pitämään mahdollisimman yksinkertaisina siten, että yhdistettynä metodin nimeen ne voidaan tulkita yksikäsitteisesti.

Tyyppi	Nimeämiskäytäntö	Esimerkki
Luokat	Ensimmäinen kirjain T, sanat isolla alkukirjaimella	TMokkula, TProjectManager
Rajapintamäärittelyt	Sanat isolla alkukirjaimella	Application, ProjectManager
Metodit	Sanat isolla alkukirjaimella, ei alaviivaa	GetGeometry, SetWindowPosition
Parametrit	Pieni alkukirjain, ei alaviivaa, sanat isolla alkukirjaimella	height, fileName

Taulukko 5: Nimeämiskäytäntö

6 Yhteenveto

Tämä dokumentti määrittelee Verstas-projektin aikataulun ja toteutusmenetelmät. Dokumentti on suunnattu pääasiassa Verstas-projektiryhmälle projektin hallinnan ja seurannan tueksi. Projektisuunnitelma otetaan käyttöön välittömästi sen hyväksymisen jälkeen. Projektiryhmä vastaa Verstas-projektin toteutumisesta tämän projektisuunnitelman mukaisesti.

Viitteet

- [1] Santanen Jukka-Pekka, Gradupohja *Microsoft Wordin template-tiedosto ja lyhyt ohje tyylien käyttöön*, saatavilla WWW-muodossa
<URL: <http://www.mit.jyu.fi/progradut/tyylijohjat/gradupohja.dot>>, viitattu 21.10.2002.
- [2] Santanen Jukka-Pekka, *Opinnäytteiden kirjoittaminen, lyhyt oppimäärä*, saatavilla WWW-muodossa
<URL: <http://www.mit.jyu.fi/santanen/info/kirjoittamisesta.html>>, viitattu 21.10.2002.