

HIBBO

Tietotekniikan sovellusprojekti

Matti Eskelinen
Olli Karppinen
Harri Kosunen
Riikka Rikkola

Projektiraportti
Versio: 1.0
28.5.2003

Jyväskylän Yliopisto
Tietotekniikan laitos

Tekijät:

- Matti Eskelinen (me@amjayee.net)
- Olli Karppinen (ollkarp@cc.jyu.fi)
- Harri Kosunen (hmkosune@cc.jyu.fi)
- Riikka Rikkola (rerikkol@cc.jyu.fi)

Työ: Projektiraportti tietotekniikan sovellusprojektiin

Työtila: Agora, huone AgC223.3

Kotisivu: <http://kotka.it.jyu.fi/hibbo/>

Tiivistelmä

Tämä dokumentti on Jyväskylän yliopistossa keväällä 2003 toteutetun Hibbo-projektin projektiraportti. Hibbo-projektissa toteutettiin Jyväskylän yliopiston fysiikan laitokselle graafinen käyttöliittymä hila-Boltzmann-simulaattoriin, sekä simulointitulosten visualisointitoiminnot. Dokumentissa kuvataan projektin toteutusta ja saavutettuja tavoitteita verrattuna suunniteltuun.

Avainsanat

Tietotekniikan Sovellusprojekti, fysiikan laitos, hila-Boltzmann, simulointi, graafinen käyttöliittymä, visualisointi, Kylix, OpenGL, Delphi

Dokumentin versiohistoria

Versio	Päivämäärä	Tekijät	Kuvaus
1.0 -1	10.5.2003	OK	Alustava versio
1.0 -2	13.5.2003	OK	Lisätty sisältöä
1.0 -3	20.5.2003	OK	Korjailtu virheitä
1.0	28.5.2003	OK	Lopullinen versio

Tekijöiden lyhenteet

ME Matti Eskelinen

OK Olli Karppinen

HK Harri Kosunen

RR Riikka Rikkola

Sisältö

1	Johdanto	1
2	Termistö	2
2.1	Yleiset termit	2
2.2	Sovellukseen liittyvät termit	2
3	Projektin tausta ja tavoitteet	4
3.1	Sovellukselle asetetut tavoitteet	4
3.2	Oppimistavoitteet	4
3.3	Tavoitteiden saavuttaminen	5
4	Projektin toteutus	6
4.1	Projektipalaverit	6
4.2	Koulutus	6
4.3	Opponoinnit	6
4.4	Loppuesittely	7
4.5	Dokumentointi	7
4.5.1	Projektin dokumentit	7
4.5.2	Sovelluksen dokumentit	8
4.6	Työnjako	9
4.7	Aikataulu	9
4.8	Projektin kulku	11
4.8.1	Projektin aloitus ja suunnitteluvaihe	11
4.8.2	Toteutusvaihe	12
4.8.3	Testausvaihe	12
4.8.4	Raportointivaihe	12
4.9	Riskien arviointi, syitä viivästymisiin	13
4.10	Projektissa kohdattuja ongelmia	14
5	Resurssit ja organisaatio	15
5.1	Henkilöt ja yhteystiedot	15
5.2	Työtilat ja laitteet	16
5.3	Työaika	16
6	Henkilökohtaiset kokemukset	18
6.1	Matti Eskelinen	18
6.1.1	Ongelmat ja yllätykset	18
6.1.2	Mitä opin	19
6.1.3	Mitä tekisin toisin	19

6.2	Olli Karppinen	20
6.2.1	Ongelmat ja yllätykset	20
6.2.2	Mitä opin	20
6.2.3	Mitä tekisin toisin	21
6.3	Harri Kosunen	21
6.3.1	Ongelmat ja yllätykset	21
6.3.2	Mitä opin	22
6.3.3	Mitä tekisin toisin	22
6.4	Riikka Rikkola	22
6.4.1	Ongelmat ja yllätykset	23
6.4.2	Mitä opin	23
6.4.3	Mitä tekisin toisin	23
7	Yhteenveto	24
7.1	Yleiskuva	24
8	Lähteet	25

1 Johdanto

Hibbo-projekti oli Jyväskylän yliopiston tietotekniikan laitoksen kevään 2003 Sovellusprojekti. Projekti suunnitteli ja toteutti Jyväskylän yliopiston fysiikan laitokselle hila-Boltzmann-simulaattoriin graafisen käyttöliittymän, jonka yksi olennainen osa on laskentatulosten visualisointi.

Hila-Boltzmann-simulointi on tehokas menetelmä erilaisten nestevirtausongelmien mallintamiseen. Sillä voidaan mallintaa nesteen virtauksia erilaisissa materiaaleissa, esimerkiksi kuitumateriaalissa. Menetelmässä mallinnettava alue jaetaan säännölliseksi kuutiohilaksi ja nestettä käsitellään joukkona partikkeleja, joita edustaa nopeusvektori jokaisessa hilapisteessä.

Simulointi toteutetaan käytännössä tietokoneohjelmistoilla, joilla tarvittaessa luodaan näytekappaleet ja suoritetaan syötettyjen parametrien avulla varsinainen simulointi. Aikaisemmin fysiikan laitoksella tehty hila-Boltzmann-simulointi on tapahtunut komentorivipohjaisesti ja tulosten visualisointi on puuttunut kokonaan. Hibbo-projektin puitteissa toteutettiin siis graafinen käyttöliittymä olemassaolevaa simulointiohjelmistoa hyödyntäen siten, että tulosten visualisointi on mahdollista. Ohjelmaa tullaan käyttämään sekä opetus- että tutkimustyössä, joten helppokäyttöisyys ja selkeys olivat tärkeitä kriteereitä sovelluksellemme. Sovelluksen toimintaympäristöinä tulevat olemaan sekä Windows, että Linux.

Tässä dokumentissa kuvataan Hibbo-projektin projektityöskentelyä ja aikataulua sekä tarkastellaan toteutunutta projektia verrattuna projekti-suunnitelmaan [2]. Luvussa 2 selostetaan projektiin ja sovellukseen liittyvät termit, luvussa 3 käydään pääpiirteissään läpi projektille alunperin asetettuja tavoitteita ja niiden toteutumista sekä projektin taustaa, luvussa 4 kuvaillaan projektin toteuttamista, aikataulullisten tavoitteiden saavuttamista ja riskienhallintaa, luvussa 5 käydään läpi Hibbo-projektin toteuttanut projektiorganisaatio ja käytettävissä olleet resurssit, luku 6 sisältää jokaisen projektilaisen henkilökohtaisia kokemuksia liittyen projektiin ja luku 7 yhteenvedon dokumentista ja projektista.

2 Termistö

Tässä luvussa listataan yleisiä projektissa vastaantulleita termejä ja sovellukseen liittyviä käsitteitä.

2.1 Yleiset termit

Alfa-vaihe on sovelluksen kehittämisen vaihe, jonka aikana valmistuu sovelluksen beeta-versio.

Beeta-vaihe on sovelluksen kehittämisen vaihe, jonka lopussa sovelluksen lopullinen versio valmistuu.

Delphi on Borlandin kehittämä Windows-käyttöjärjestelmissä toimiva IDE (Integrated Development Environment) eli ohjelmankehitysympäristö, jossa ohjelmointikielenä käytetään Object Pascal -kieltä.

Kylix on Delphin vastine Linux-ympäristöön.

OpenGL on Silicon Graphics Inc:in kehittämä vapaa grafiikkakirjasto, lyhenne sanoista Open Graphics Library. Tässä projektissa OpenGL:ää käytetään laskentatulosten graafiseen esittämiseen.

Prototyyppi on ohjelma, joka toteuttaa osan suunnitellun sovelluksen toiminnallisuudesta.

2.2 Sovellukseen liittyvät termit

Beeta-versio on sovelluksen versio, jossa on kaikki osat ovat paikallaan, mutta ominaisuuksia puuttuu.

Hila-Boltzmann-simulointi on tehokas, diskreetti menetelmä nestevirtausongelmien mallintamiseen.

Käyttötapaus on käyttäjän tai sovelluksen toiminnan kuvaus tietyn tehtävän suorittamiseksi.

Laskentageometria on hila-Boltzmann-simuloinnissa kolmiulotteinen kuutiolahila, joka sisältää tiedon siitä, mitkä hilan pisteet kuuluvat kiinteään aineeseen ja mitkä ovat nestettä.

Lokitiedostot ovat simulaatio-ohjelman kirjoittamia tiedostoja, jotka sisältävät laskentaprosessin tilasta kertovia tunnuslukuja.

Virtauslaskenta on nesteiden tai kaasujen virtauksien matemaattista mallintamista.

Virtaviivat ovat visualisointikuviin piirrettäviä viivoja, jotka kuvaavat kuvitteellisten nestepartikkelien liikerataa.

Visualisointi tarkoittaa datasta luotavia graafisia, havainnollistavia esityksiä. Jos laskentageometria esitetään ennen simulointia kolmiulotteisena kuvana, käyttäjä voi varmistua näytteen järkevyydestä. Simuloinnin jälkeen visualisointi tapahtuu piirtämällä erilaisia nesteen liikettä ja tilaa kuvaavia havainnollistavia esityksiä.

3 Projektin tausta ja tavoitteet

Tässä luvussa kerrataan projektin taustaa ja sille asetettuja tavoitteita. Tavoitteenasettelu on jaoteltu sovellukselle asetettuihin tavoitteisiin ja yleisemmin projektiin liittyviin opetuksellisiin tavoitteisiin.

3.1 Sovellukselle asetetut tavoitteet

Hibbo-projektin taustan ja pohjan sen tavoitteenasettelulle muodosti Jyväskylän yliopiston fysiikan laitoksen hila-Boltzmann-simulaattori, jota on tähän asti voinut käyttää vain komentorivipohjaisesti, eikä tulosten visualisointimahdollisuutta ole ollut. Hibbo-projektin tavoitteeksi asetettiin graafisen käyttöliittymän ja visualisointitoimintojen toteuttaminen fysiikan laitoksen hila-Boltzmann-simulaattoriin. Yksityiskohtaisesti Hibbo-projektille asetetut vaatimukset on kuvattu vaatimusmäärittelyssä [1].

Sovellukselle asetetut vaatimukset tiivistetysti:

- Erillisen näytteenluontiohjelman käynnistys annetuilla parametreilla
- Erillisen hila-Boltzmann-simulaattorin käynnistys annetuilla parametreilla
- Simuloimattoman näytteen visualisointi luodun näytteen järkevyyden varmistamiseksi
- Simuloinnin tulosten tarkastelu eri visualisointitoimintojen avulla
- Visualisointitoimintoina 2d-leikkeet, virtaviivat ja animoidut merkkipartikkelit
- Toimivuus sekä Windows-, että Linux-käyttöjärjestelmissä

3.2 Oppimistavoitteet

Sovellukselle asetettujen tavoitteiden rinnalla Sovellusprojekteilla oli myös opetuksellisia tavoitteita. Projektisuunnitelmassa [2] Hibbo-projektin opetuksellisiksi tavoitteiksi asetettiin konkreettisen kokonaiskuvan saaminen laajahkosta ohjelmistoprojektista, ryhmätyötaitojen kehittäminen, OpenGL-grafiikkakirjastoon tutustuminen sekä kokonaiskuvan saaminen graafisen käyttöliittymän ja visualisoinnin tuottamisesta, aina suunnittelusta toteutukseen ja työn tulosten dokumentointiin saakka.

Dokumentoinnissa käytettävän \LaTeX :in ansiosta osalle ryhmäläisistä tarjoiu lisäksi mahdollisuus oppia uusi tehokas tapa laadukkaiden dokumenttien laadintaan.

3.3 Tavoitteiden saavuttaminen

Projektissa ei kohdattu ylitsepääsemättömiä ongelmia, sen voidaan katsoa onnistuneen hyvin ja täyttävän sille etukäteen asetetut tavoitteet. Tämä pätee sekä sovellukselle asetettuihin tavoitteisiin, että pääosin myös opetuksellisiin tavoitteisiin. Aikataululliset tavoitteet eivät kaikilta osin toteutuneet, viivästymistä tapahtui niin eri dokumenttien palautusten kuin projektin päätöspäivämäärän osalta.

4 Projektin toteutus

Tässä luvussa selostetaan projektin toteuttamiseen liittyneitä käytäntöjä, kuten projektipalaveri- ja dokumentointikäytännöt. Lisäksi käydään läpi aikataulutusta vertaillen suunniteltua ja toteutunutta aikataulua sekä yleisesti projektin toteutumista verrattuna suunnitelmaan.

4.1 Projektipalaverit

Projektipalavereja pidettiin yhteensä 16 kertaa. Palavereja pidettiin viikoittain, poikkeuksena ainoastaan pääsiäisviikko, jolloin projektipalaveria ei pidetty. Palavereissa tarkkailtiin projektin yleistä etenemistä ja ne tarjosivat mahdollisuuden työn tulosten kommentointiin ja kysymyksien esittämiseen kaikille projektiorganisaation jäsenille. Lähes kaikissa palaverissa pidettiin erillinen viikkokatsaus, jossa projektin jäsenet kertovat mitä ovat viikon aikana tehneet. Projektipalavereista kirjoitettiin myös aina pöytäkirja, joka lähetettiin projektiorganisaation jäsenille ennen seuraavaa palaveria, ja tarkastettiin seuraavan palaverin alussa. Ryhmän jäsenistä yksi toimi kahdesti peräkkäin palaverin puheenjohtajana, sihteerinä toimivat yleensä Riikka Rikkola tai Olli Karppinen. Palaverit olivat erityisen tarpeellisia projektinhallinnon välineitä, ja alussa esiintyneen ehkä liiallisenkin dokumenttien muotoseikkoihin takertumisen jälkeen ne olivat tärkeä osa projektin etenemistä. Samalla ne tarjosivat jatkoa ajatellen mainion tilaisuuden oppia perustaidot kokoustekniikoista.

Varsinaisten projektipalaverien ohessa ryhmämme kokoontui keskenään melko säännöllisesti pohtimaan projektin etenemistä ja työnjakoa. Myös yhteisiä palaverinomaisia suunnittelutuokioita pidettiin, ja ryhmämme koki ne hyödyllisiksi.

4.2 Koulutus

Projektiryhmille järjestettiin pakollisena projektiin kuuluvana osana koulutusta liittyen CVS:n käyttöön, tekijänoikeuksiin, käyttöliittymään ja käytettävyyteen sekä yleiseen projektin läpiviemiseen. Hibbo-ryhmä osallistui kokonaisuudessaan kaikkiin projekteissa järjestettyihin koulutustilaisuuksiin.

4.3 Opponoinnit

Projektin toteutukseen liittyi kaksi opponointitilaisuutta, joissa projektiryhmät arvioivat toisten ryhmien töitä sekä esittelivät omaa aihettaan ja

sovellustaan. Ensimmäinen opponointi pidettiin 24.3. ja toinen 28.4. Muiden ryhmien antaman palautteen sekä ryhmän oman kokemuksen mukaan opponoinnit sujuivat kokonaisuudessaan varsin hyvin.

4.4 Loppuesittely

Projektin läpiviennin ”huipentumana” ja saavutettujen tulosten esittelytilaisuutena järjestettiin 15.5. loppuesittely. Siellä projektiryhmät esittelivät aiheitaan ja työstämäänsä sovellusta. Hibbo-ryhmä toteutti esittelynsä siten, että Riikka Rikkola esitteli projektia yleensä ja kertoi sen taustasta, Olli Karppinen kertoi sovelluksen toiminnallisista yhteyksistä ja tiedostoista, Matti Eskelinen selvitti toteutuskäytäntöjä ja yhdessä Harri Kosusen kanssa näytti sovelluksen toimintaa käytännössä. Ryhmän esitys eteni suunnitelmien mukaan ja onnistui kokonaisuutena mainiosti. Esityksen jälkeen Markus Inkeroinen antoi palautetta koskien koko projektia. Positiivisena hän piti ryhmän innokkuutta työntekoon ja sovelluksen onnistumista, huomautettavaa oli sen sijaan työnjaon vinoutumisesta ja suunnitteluvaiheen sekä koko projektin viivästyisestä.

4.5 Dokumentointi

Dokumenttien hyväksymiskäytäntönä oli, että viimeistään palaveria edeltävänä perjantaina ryhmä lähetti käsiteltävän dokumentin projektiorganisaation tarkasteltavaksi, ja palaverissa voitiin vain käsitellä kyseisen dokumentin puutteet ja päättää hyväksymisestä. Ryhmän tuottamat dokumentit voidaan jakaa kahteen pääryhmään sen mukaan, kuuluvatko ne projektin vai sovelluksen dokumentaatioon. Seuraavissa aliluvuissa on lueteloitu ryhmän tuottamat dokumentit tarkoituksineen, tavoitteeksi asetetut valmistumispäivämäärät ja toteutununeet päivämäärät käyvät ilmi luvussa 4.7.

4.5.1 Projektin dokumentit

Palaverien pöytäkirjoissa selostettiin palavereissa käsitellyt asiat ja tehdyt päätökset.

Projektisuunnitelmassa kuvattiin ryhmän suunnitelma projektin yleisestä läpiviemisestä. Siinä kuvattiin projektin taustaa, tavoitteita, resursseja, aikataulutusta ja työnjakoa sekä mahdollisia riskejä ja niiden hallintaa [2].

Opponointiraporteissa selostettiin opponoinnin jälkeen ryhmän omat tuntemukset esityksen sujumisesta sekä eriteltiin muiden ryhmien antamaa palautetta [9], [10].

Projektiraportti on tämä dokumentti, jossa kuvaillaan projektin toteuttamista ja saavutettuja tavoitteita verrattuna projektisuunnitelmaan. Projektiraportin yhteyteen on myös liitetty jokaisen ryhmäläisen itsearvioinnit saavutetuista tavoitteista ja kokemuksista [11].

4.5.2 Sovelluksen dokumentit

Vaatimusmäärittelyssä määriteltiin tarkoin sovellukselle asetetut ominaisuudet, vaatimukset ja määrittelyt. Vaatimusmäärittely toimi ohjeena sovelluksen toteuttajille suunnitteluvaiheessa sekä sopimuksena tilaajan kanssa [1].

Sovellussuunnitelmassa kuvattiin suunnitelma Hibbo-sovelluksen toteuttamiseksi vaatimusmäärittelyn pohjalta. Dokumentissa annetaan kuva sovelluksen rakenteesta ja toiminnasta sekä yhteydestä muihin sovelluksiin ja tiedostoihin [3].

Sovellusraportissa kuvattiin sovelluksen toimintaa ja toteuttamista sovellussuunnitelman pohjalta. Sovellusraportissa myös selostettiin tarkoin sovellussuunnitelmassa avoimiksi jääneiden ominaisuuksien toteuttaminen [5].

Testaussuunnitelmassa kuvattiin suunnitelma Hibbo-sovelluksen testaamiseksi. Testaussuunnitelmassa käytiin läpi testauksen vaiheet ja tarkoitus sekä laitteet ja ohjelmistot, joilla testaus suoritetaan [4].

Testausraportissa selostettiin testauksen tulokset, joiden pohjalta kehitystyötä on edelleen jatkettu tai sovellus todettu toimivaksi [6].

Käyttöohjeet sisältävät yksikäsitteiset ja selkeät ohjeet sovelluksen tarkoituksenmukaista käyttöä varten. Ne ovat käytettävissä itse sovelluksessa HTML-muodossa ja lisäksi tulostettuna versiona [7].

Asennusohje sisältää yksikäsitteiset ja selkeät ohjeet Hibbo-sovelluksen asentamiseksi [8].

4.6 Työnjako

Projektin työnjakoa suunniteltaessa pyrittiin siihen, että jokainen saisi kiinnostustaan ja taitojaan vastaavia tehtäviä kuitenkin siten, että kaikki osallistuvat kaikkiin työvaiheisiin. Vastuualueiksi sovittiin projektin alussa seuraavat: Matti Eskelinen, tekniikka, Olli Karppinen, tiedotus ja dokumentit, Harri Kosunen, projektipäällikkö ja Riikka Rikkola, dokumentit. Projektipäällikön tehtävään valittiin Harri Kosunen hänen itse halutessaan tehtävään, jossa hän myös toimi läpi koko projektin.

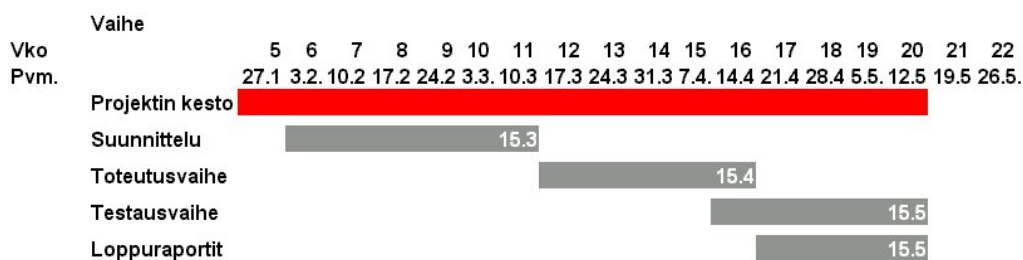
Toteutunut työnjako noudatteli kohtalaisen tarkkaan suunniteltuja päävastuualueita. Matti Eskelinen ryhmän selvästi kokeneimpana ohjelmoijana kantoi raskaimman vastuun sovelluksen toteuttamisesta, varsinkin visualisointiominaisuuksien osalta. Olli Karppinen vastasi tiedottajan ominaisuudessa ryhmän yhteydenpidosta lähinnä muuhun projektiorganisaatioon, Harri Kosunen kantoi projektipäällikön vastuun ja osallistui myös vahvalla panoksella toteutukseen, samoin kuin Riikka Rikkola päävastuualueensa dokumentoinnin ohessa.

Ryhmän etukäteissuunnitelma työnjaon osalta toteutui siltä osin hyvin, että jokainen todellakin pääsi osallistumaan kaikkiin projektin työvaiheisiin, suunnitteluun, toteutukseen ja dokumentointiin. Työnjako kuitenkin ehkä jossain määrin vinoutui liikaa sikäli, että Matti Eskelinen vastasi lähes yksinään visualisointitoiminnoista. Tämä puolestaan oli tietyllä tapaa "pakon sanelemaa", koska hänellä oli eniten kokemusta ohjelmoinnista ja tiukan projektiaikataulun vuoksi osa ryhmästä ei ennätänyt paneutua läheskään riittävästi OpenGL:n ominaisuuksiin.

4.7 Aikataulu

Projekti toteutettiin 27.1.2003-28.05.2003 välisenä aikana. Projektisuunnitelmassa [2] määriteltiin projektin yleiselle aikataulutukselle karkea kuukausittainen kolmijako. Projektin alusta maaliskuun puoliväliin oli varattu 1,5 kuukautta suunnitteluun ja tutustumiseen, maaliskuun puolivälistä huhtikuun puoliväliin kuukausi toteuttamiseen ja viimeinen kuukausi toukokuun puoliväliin saakka testaukselle ja loppudokumentoinnille. Lisäksi projektisuunnitelmaan asetettiin erillisiä tarkistuspistepäivämääriä välietapeiksi. Toteutusvaihe oli erikseen jaettu alfa- ja beta-vaiheisiin.

Etenkin dokumenttien valmistuspäivämäärien suunniteltiin noudattelevan tätä kolmijakoa. Kuvassa 1 projektin suunniteltu aikataulu kaaviona.



Kuva 1: Projektin suunniteltu aikataulu.

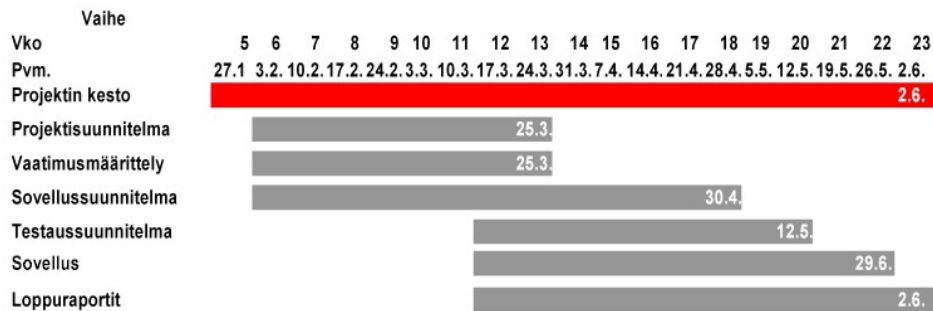
Aikataulutus toteutui tietyiltä osin moitteettomasti, mutta useissa projektinvaiheissa tapahtui myös selvää viivästymistä suunniteltuun aikatauluun nähden. Syitä jotka tähän johtivat pohditaan luvuissa 4.8 ja 4.9 tarkemmin.

Suunnitellut ja toteutuneet tarkistuspäivämäärät

Työ	Tavoite	Valmis
Projektisuunnitelma	12.3.	26.3.
Vaativuusmäärittely	12.3.	26.3.
Sovellussuunnitelma	19.3.	30.4.
Beeta-versio	26.3.	26.3.
Lopullinen versio	16.4.	28.5.
Asennusohjeet	15.5.	28.5.
Käyttöohjeet	15.5.	28.5.
Projektiraportti	15.5.	2.6.
Sovellusraportti	15.5.	2.6.

Pahiten laadittuihin tarkistuspäivämääriin nähden jouduttiin taipumaan itse sovellusta ja varsinkin visualisointitoimintoja suunniteltaessa. Ryhmän ensimmäinen versio sovellussuunnitelmasta [3] ei kuvannut tarpeeksi eksaktisti suunnitelmaa virtausten visualisointien toteuttamisesta ja oli muutenkin puutteellinen. Sovellussuunnitelma saatiinkin hyväksyttävään kuntoon vasta yli kuukausi suunniteltua myöhemmin, eli huhtikuun lopussa.

Sovellukselle suunniteltu jako alfa- ja beta-vaiheisiin toteutui alfa-vaiheen osalta kohtuullisesti, mutta beta-vaiheen myötä valmistuvaksi suunniteltu lopullinen versio myöhästyi lopulta yli kuukauden. Kuvassa 2 projektin toteutunut aikataulu kaaviona.



Kuva 2: Projektin toteutunut aikataulu.

4.8 Projektin kulku

Tässä aliluvussa selostetaan projektin etenemistä ja kulkua lähinnä suunniteltujen työvaiheiden toteuttamisen kannalta. Lisäksi kommentoidaan projektityön eri vaiheita.

4.8.1 Projektin aloitus ja suunnitteluvaihe

Projektin aloitettiin tehtävänannon taustalla olevaan aiheeseen, eli hila-Boltzmann-simulointiin tutustumisella. Tähän sisältyi muunmuassa tutustumiskäynti ja pienimuotoinen aiheeseen ja tehtävään perehdyttäminen fyysikan laitoksella. Tutustumiskäynti selkiytti huomattavasti työstettävää tehtävää ja sen ideaa, samoin kuin alkuvaiheen palaverit. Suunnitteluvaihe käsitti itse projektikonaisuuden, sovelluksen kuin myös testauksen suunnittelun. Eri vaiheiden dokumentit on listattu luvussa 4.5. Virtausten visualisoinnissa käytettävien algoritmien suunnittelussa ja taustalla olevan matematiikan kuvaamisessa olivat tarpeen myös vastaavien ohjaajien Kari Kärkkäisen ja Markus Inkeröisen aiheesta pitämät perehdytykset. Rainer Koreasalo piti myös ryhmälle projektin alussa tarpeellisen perehdytyksen OpenGL-kirjaston käyttöön.

Sovellusta hahmoteltiin jo suunnitteluvaiheessa prototyyppejä ohjelmoimalla. Prototyyppien työstämistä ja suunnittelua päällekkäin jatkettiin kuitenkin ehkä liian pitkään, kun olisi jo pitänyt saada paperille selkeä suunnitelma sovelluksen toteuttamisesta. Tämän vuoksi huhtikuussa jouduttiin jo alkanut toteuttaminen keskeyttämään yhteisellä päätöksellä, selkeän ja kattavan sovellussuunnitelman aikaansaamisen varmistamiseksi. Ongelmia aiheutti lähinnä se, että muunmuassa visualisointitoimintojen toteuttamisen suunnitelma oli puutteellinen ja että kaikille ryhmäläisille ohjelmointi kyseisen sovellussuunnitelman pohjalta ei olisi ollut mahdollista. Ryhmän keskittäessä koko työpanoksensa sovellussuunnitelmaan se

saatiinkin riittävän laadukkaaksi, tosin yli kuukauden suunnitellusta aikataulustaan jäljessä.

4.8.2 Toteutusvaihe

Toteutusvaihe käsitti itse sovelluksen ohjelmoinnin. Toteuttamisen voidaan katsoa jakaantuvan tehtävänannon mukaisesti erillisiin graafiseen käyttöliittymään ja visualisointitoimintoihin. Toteutustyökaluiksi määriteltiin jo projektin alussa Delphi/Kylix, joten ne eivät tulleet uutena yhdellekään ryhmän jäsenistä. Sen sijaan visualisoinnissa käytettävä OpenGL-grafiikkakirjasto oli lähes täysin tuntematon kaikille muille ryhmän jäsenille paitsi Matti Eskeliselle, joten oli luonnollista, että hän otti päävastuun visualisointitoimintojen toteuttamisesta. Toteutuksessa olennaiseen rooliin nousivat Matin projektin käyttöön toteuttamat omat komponentit, joiden avulla vaativat visualisointitoiminnot saatiin hienosti onnistumaan. Toteutuksessa olivat ryhmäläisten itsensä toteuttamat komponentit muutenkin merkittävässä asemassa, niiden avulla voitiin toteuttaa sovelluksen vaatimia spesifisiä tarpeita, esimerkiksi Harri Kosusen toteuttamalla Graafi-komponentilla piirretään kuvaajaa permeabiliteetin kehitymisestä simuloinnin aikana.

Toteutuksessa vastuuta ja töitä pyrittiin jakamaan kunkin ryhmäläisen taitojen mukaan. Lukuunottamatta visualisointitoimintoja, jotka olivat Matin vastuulla, voidaan sanoa, että ryhmäläisistä jokainen osallistui sovelluksen toteutukseen.

4.8.3 Testausvaihe

Testausvaihe käsitti sovelluksen järjestelmällisen testauksen, laadittua testaussuunnitelmaa noudattaen [4]. Testausvaiheeseen liittyen kirjoitettiin testausraportti [6].

4.8.4 Raportointivaihe

Loppuraportoinnin tarkoituksena on koota yhteen projektin kokemukset ja arvoioda toisaalta itse sovelluksen toimivuutta ja ominaisuuksia ja toisaalta yleistä projektin läpivientiä. Tähän vaiheeseen liittyvät raportit ja niiden tarkoitus on listattu luvussa 4.5.

4.9 Riskien arviointi, syitä viivästymisiin

Tässä luvussa käydään läpi projektisuunnitelmassa [2] arvioitujen riskien toteutumista ja niiden hallintaa.

Projektisuunnitelman arvioita riskeistä

Otsikko	Todennäköisyys	Vaikutus
Kokemattomuus	Suuri	Kohtalainen
Ohjelmointi	Kohtalainen	Erittäin suuri
Tilaaaja	Pieni	Suuri
Laitteisto	Pieni	Erittäin suuri
Ohjaajat	Pieni	Kohtalainen
Sairastumiset	Pieni	Kohtalainen
Ulkoisen viestintä	Pieni	Kohtalainen
Sisäinen viestintä	Pieni	Kohtalainen
Odottamattomat ongelmat	Kohtalainen	Ongelman luonteesta riippuen suuri tai pieni

Projektisuunnitelmassa tehdyssä riskianalyysissä listattiin suurimmiksi projektin onnistumista uhkaaviksi riskeiksi projektikokemattomuus ja eri syistä johtuvat ongelmat ohjelmoinnissa. Projektikokemattomuudesta voidaan katsoa juontaneen erään ryhmämme kohtaaman ongelman, aikataulun viivästyminen sovellusta suunniteltaessa. Mattia lukuunottamatta muut ryhmäläiset olivat ensimmäistä kertaa mukana vastaavanlaisessa projektityössä, joten kokemattomuus oli ymmärrettävää. Kokemattomuus ilmeni käytännön ongelmina etenkin siinä, ettei kaikilta osin ymmärretty kaikkien suunnitteludokumenttien merkitystä ja niiden vaatimuksia. Ohjaajien kommentit ja ohjeet näiden dokumenttien kehittämisessä olivatkin etenkin sovellussuunnitelman tapauksessa arvokkaita. Ohjelmointiongelmat oli projektisuunnitelmassa arvioitu kohtalaisen todennäköiseksi riskiksi. Ylitsepääsemättömiä ohjelmointiongelmia ei ryhmämme kohdannut, Matti onnistui ammattimaisella otteellaan ratkaisemaan myös haastavat ongelmat visualisointien suhteen.

Muita riskianalyysissä arvioituja riskejä ei ryhmämme kohdannut, vaan yhteistyö ja viestintä niin tilaajien kuin ohjaajien kanssa toimi hyvin, samoin kuin käytössämme ollut laitteisto. Kukaan ryhmäläisistä ei onneksi myöskään sairastunut vakavasti projektin aikana.

4.10 Projektissa kohdattuja ongelmia

Projektissa koettiin ongelmana lähinnä työnjaon vinoutuminen toteutus-toimintoihin liittyen. Kireästä aikataulusta ja Matti Eskelisen muita huomattavasti suuremmasta kokemuksesta johtuen hänen vastuulleen tuli virtausten visualisointiin liittyvät toiminnot, eivätkä muut niihin juuri kajonneet. Ryhmän sisällä työnjako kuitenkin käsitettiin ”projektin realiteetiksi”, ja kaikki hyväksyivät tilanteen.

Yleisesti projekteihin liittyvänä ongelmana ryhmämme havaitsi esimerkiksi ”kroonisen” ajanpuutteen, tätä voisi ehkä helpottaa siirtämällä projektin aloitusajankohdan niin lähelle lukukauden alkua kuin mahdollista, näin käytössä olisi arvokkaita lisäviikkoja. Toisaalta projektien aiheet ryhmittäin vaihtelevat vaativuudeltaan ja laajuudeltaan kohtalaisen paljon, tämäkin yhdessä kaikille yhteisen, tiukan aikataulun kanssa saatetaan kokea ongelmana tai epäoikeudenmukaisena.

Muilta osin projektin koettiin etenevän ilman suurempia ongelmia ja projekti koettiin mielekkäänä. Projektin kulkua ryhmäläisten itsensä näkökulmasta selostettu luvussa 6.

5 Resurssit ja organisaatio

Tässä luvussa käydään läpi projektiin osallistuneet henkilöt sekä heidän yhteystietonsa. Myös projektin käytössä olleet resurssit esitellään.

5.1 Henkilöt ja yhteystiedot

Hibbo-projektiryhmän jäseniä olivat

Matti Eskelinen (me@amjayee.net),
Olli Karppinen (ollkarp@cc.jyu.fi),
Harri Kosunen (hmkosune@cc.jyu.fi) sekä
Riikka Rikkola (rerikkol@cc.jyu.fi).

He suunnittelivat, toteuttivat ja dokumentoivat tehtävänannon mukaisen sovelluksen. Projektin vastaavana ohjaajana toimivat

Kari Kärkkäinen (ktkar@mit.jyu.fi) sekä
Markus Inkeroinen (marink@mit.jyu.fi),

joka oli sovellusprojektissa harjoittelemassa vastaavan ohjaajan tehtäviä. Teknisenä ohjaajana toimi

Rainer Koreasalo (rakorea@cc.jyu.fi)

ja projektin toteutusta ja kulkua seurasi

Jonne Itkonen (ji@mit.jyu.fi).

Projektin tilaajaa edustivat Jyväskylän yliopiston fysiikan laitokselta

Markku Kataja (markku.kataja@phys.jyu.fi),
Antti Koponen (antti.koponen@phys.jyu.fi),
Ari Jäsberg (ari.jasberg@phys.jyu.fi) sekä
Pasi Raiskinmäki (pasi.raiskinmaki@phys.jyu.fi).

Aktiivisimmin projektiin osallistui tilaajan edustajista Ari Jäsberg, joka oli aktiivisesti mukana palaverissa sekä sovelluksen ohjaamisessa ja testauksessa.

5.2 Työtilat ja laitteet

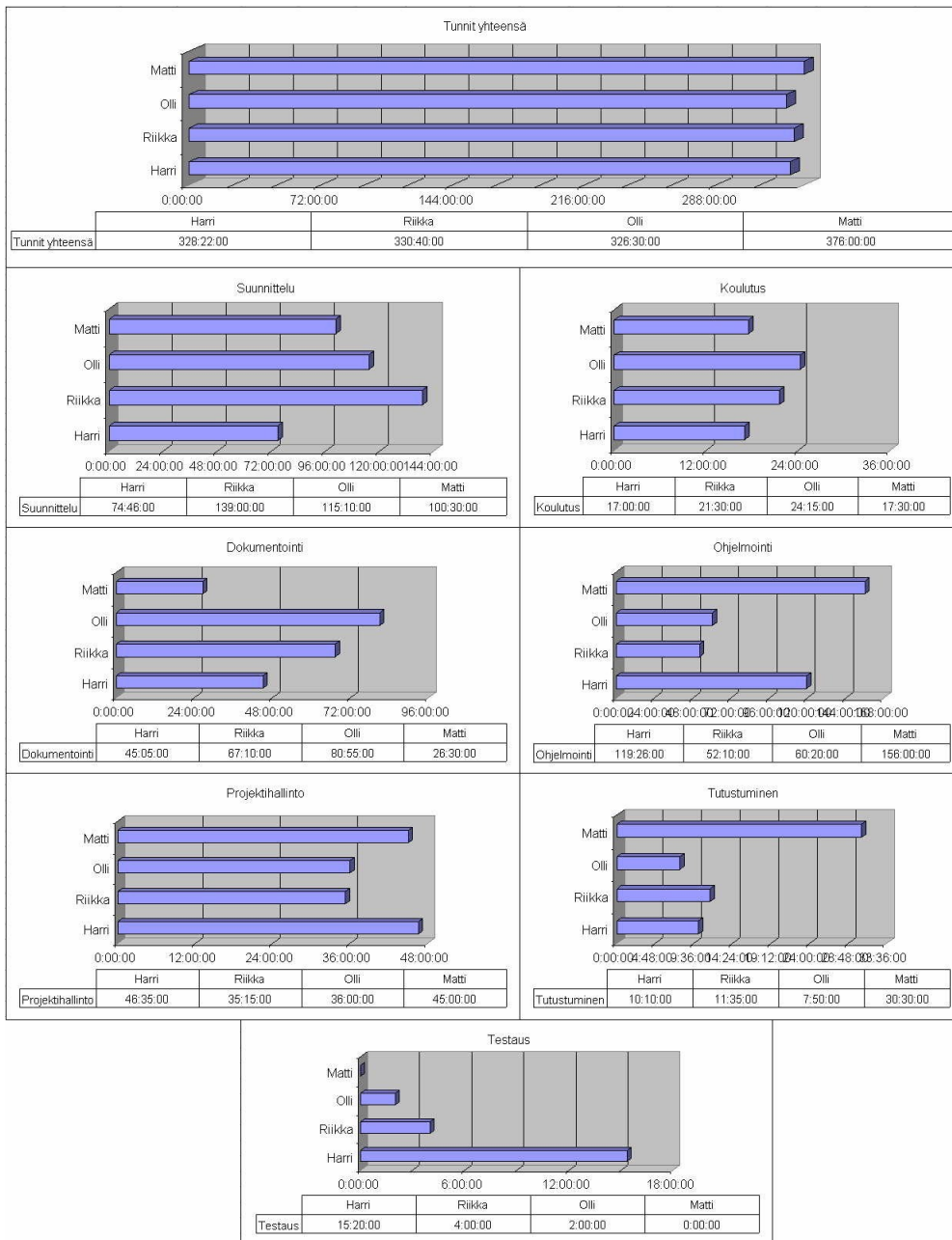
Jyväskylän yliopiston tietotekniikan laitos antoi projektiryhmän käyttöön Agoran tiloissa sijaitsevan projektihuoneen, AgC223.3.

Projektihuoneessa oli neljä tietokonetta. Kolmessa koneessa käyttöjärjestelmänä oli Windows 2000 ja yhdessä RedHat 7.3. Ohjelmankehitysympäristöinä olivat Delphi ja Kylix. Projektin alkuvaiheessa ryhmä sai myös tilaajalta käyttöönsä hila-Boltzmann-simulaattorin sekä näytteenluontisovelluksen, joita tarvittiin sovellusta kehitettäessä ja testattaessa. Ryhmällä oli myös käytettävissä tietotekniikan laitoksen kopiokone, projektitilassa Ag C223.1 oleva tulostin sekä projektihuoneessa oleva puhelin.

Projektin käytössä ollut tekniikka toimi koko projektin ajan pääpiirteissään moitteettomasti.

5.3 Työaika

Kuvassa 3 kaavioituna projektilaisten ajankäyttö.



Kuva 3: Toteutuneet työtunnit.

6 Henkilökohtaiset kokemukset

Seuraavissa aliluvuissa Hibbo-projektin jäsenet kertovat omista henkilökohtaisista kokemuksistaan projektiin liittyen.

6.1 Matti Eskelinen

Ennen projektia olin ollut töissä ohjelmointihommissa, tosin kyseisessä yrityksessä projektikäsitys ei ollut kovin selkeä. Samoin muutamat kaverit olivat tehneet projektin aiemmin ja heiltä kuulin joitakin kokemuksia. Projektin aikana ei siis tullut suuria yllätyksiä. Ainoastaan aiheemme osoittautui työläämmäksi kuin odotin.

Minulla on kokemusta kaikenlaisesta ohjelmoinnista pitkältä ajalta, joten tiesin selviäväni kaikista toteutukseen liittyvistä asioista. Ryhmätyöskentely oli ainoa epäilyksenalainen asia; aiemmasta kokemuksesta tiesin, että en pysty työskentelemään aivan kaikenlaisten ihmisten kanssa. Tässä suhteessa ei tullut mitään ongelmia. Muut ryhmän jäsenet olivat mukavia ja ahkeria, kukaan ei löysäillyt tehtävissään. Ainoastaan muiden kokemattomuus ohjelmistonsuunnittelutehtävissä aiheutti vaikeuksia. Toisilla ei ollut selkeitä näkemyksiä toteutukseen liittyvissä asioissa, ja monesti päätökset vain jäivät minun harteilleni. Tiedostin ryhmätyöskentelyn ja työn jakamisen tarpeen, mutta töiden jakaminen ei onnistunut niin kuin toivoin. Ryhmä olisi selkeästi kaivannut toisen kokeneen ohjelmoijan, jonka kanssa näkemyksiä olisi voinut vaihtaa.

Olen tyytyväinen ryhmän toimintaan. Kaikki ottivat työnsä tosissaan. Olen myös hyvilläni siitä, että kun minulla toteutusasiat vaativat kaiken huomion, muut kantoivat vastuuta muista asioista. Jos minun olisi pitänyt murehtia myös kaikkia muita projektin osa-alueita, ohjelma ei olisi valmistunut ollenkaan. Tietysti työnjaon vinoutuminen oli ikävää, mutta tässä ryhmässä, tällä työn määrällä ja ajan vähyys huomioon ottaen se oli ehkä välttämätöntä sovelluksen valmistumisen kannalta.

6.1.1 Ongelmat ja yllätykset

Suunnitteluvaiheen venyminen yllätti ja aiheutti hieman ongelmia. Osittain ongelma johtui siitä, että projektin aihe oli laaja ja vaikea, ja aikaa suunnitteluun oli liian vähän; muista projekteista saatujen kokemusten perusteella tiedän, että ainakin minulla tarkan suunnittelun tekeminen vaatii aikansa. Nyt suunnittelu meni hyvin, ja olin kohtalaisen tyytyväinen suunnitelmaan, mutta aikaa meni juuri sen verran kuin tarvittiin. Ikävää, että sitä tarvittiin suunniteltua enemmän, ja toteutukseen jäi aikaa lii-

an vähän. Kuukausi enemmän aikaa, ja töitäkin olisi pystynyt jakamaan enemmän. Olin toivonut, että olisin voinut antaa joitakin visualisointihommia muiden tehtäväksi sen jälkeen, kun olin saanut pohjatyön tehtyä, mutta nyt aika ei riittänyt. Aikatauluongelmien takia en myöskään pystynyt osallistumaan dokumenttien tekemiseen niin paljon kuin olisin halunnut.

6.1.2 Mitä opin

Ohjelmointipuolella uutta oppii aina. OpenGL oli hiukan tuttu jo ennestään, mutta en ollut itse tehnyt sillä mitään. Nyt sain varsin hyvän kuvan aiheesta ja sain myös kehitettyä hyödyllisiä, yleiskäyttöisiä komponentteja, joiden avulla omien jatkoprojektien toteuttaminen on helppoa. Oli myös hyödyllistä nähdä toisten ohjelmointitapoja; toisten ajatusmalleista, hyvistä ideoista ja virheistäkin oppii. Ja toisten koodin lukeminen on hyödyllistä, vaikka siihen onkin jo kohtalaisen tottunut.

L^AT_EX oli tuttu jo ennestään, sen osalta uutta ei juuri tullut. Dokumenttien kirjoittamisessa ymmärsi tiettyjen muotoseikkojen tärkeyden. Tiivistelmät, johdannot ja yhteenvedot ovat tärkeitä. IEEE:n standardi vaatimusmäärittelyille tuli myös tutuksi.

Palaveritaitoja opin jonkin verran, samoin ryhmätyötaidot kehittivät. Loppuvaiheessa töiden jakaminen alkoi onnistua paremmin; tosin syynä oli osittain se, että suunnitelman tekemisen jälkeen oma kuva ohjelmasta oli paljon selkeämpi. Myös projektinhallinnan merkitys selkeni. Tulevissa projekteissa ymmärrän paremmin, kuinka asioita kannattaa viedä eteenpäin, ja tiedostan paremmin suunnitteludokumenttien merkityksen.

6.1.3 Mitä tekisin toisin

Suunnittelun venyminen oli ongelma, ja pyrkisin ehkä puristamaan suunnittelun loppuun aiemmin. Ongelma on kuitenkin se, että tiedän kyllä kokemuksesta, että suunnittelu vain vaatii aikansa. Sitä olisi voinut nopeuttaa, jos ryhmässä olisi ollut toinen jäsen, jolla olisi ollut selkeitä näkemyksiä toteutuksesta.

Tiesin, että projekti pitää aloittaa terävästi, ja omasta mielestäni aloitus sujuikin hyvin. Ehkä alussa olisi pitänyt käyttää vieläkin enemmän aikaa suunnitteluun, vaatimusmäärittelyyn ja protoiluun. Käytettyyn protoilu/suunnittelumenetelmään olen tyytyväinen; se on toiminut ennenkin, ja se on ainoa oikea tapa ainakin minulle, jos en aiemmin ole tehnyt vastaavaa ohjelmaa. Aikaa vain oli liian vähän.

6.2 Olli Karppinen

En ollut koskaan aikaisemmin ollut mukana missään varsinaisessa tiiviillä, kiinteällä aikataululla toteutetussa projektityössä. Sovellusprojekteista olin kuullut edellisenä keväänä projektin suorittaneilta kavereiltani. Pyrin siis lähtemään ennakkoluulottomalla asenteella mukaan.

Tiesin etukäteen omaavani vain välttävät tai kohtalaiset perusohjelmointitaidot, jotka tulisivat joutumaan todella kovalle koetukselle. Onneksi kuitenkin projektissamme pääsi käyttämään Delphiä, joka oli ennakkoon tuttu. Tietenkin uuden kielen opettelussa olisi ollut omat puolensa, mutta tämä sopi minulle hyvin. OpenGL oli minulle ennakkoon täysin vieras, ja valitettavan vähän - huomattavasti vähemmän kuin olin kaa vaillut - kykenin siihen paneutumaan. Tämä johtui hyvin kireästä projekti aikataulusta ja pitkälle rationalisoidusta työnjaosta ryhmässämme, Matti kokeneimpana kantoi kovimman vastuun visualisointitoiminnoista.

Projektin toteutuksessa kannoin vastuuta suunnittelusta, dokumentoinnista, tiedottamisesta ja rahkeitteni mukaan myös toteutuksesta. Kokonaisuutena projekti oli erittäin antoisa ja opettavainen kokemus, tulini mainioisti toimeen kaikkien projektiorganisaatioon kuuluvien kanssa ja työtä oli hienoa tehdä yhteisen tavoitteen eteen.

6.2.1 Ongelmat ja yllätykset

Ongelmaksi koen projektikokonaisuudessa omalta osaltani sen, että visualisointitoiminnot siirtyivät niin yksin Matin harteille. Tähän vaikutti Matin huomattavasti suurempi kokemus ja projektin kireä aikataulu. Tietenkin on luonnollista, että työnjako projektissa voi olla jyrkkä, mutta kuitenkin oman panoksen olisi pitänyt olla suurempi. Tästä siis opittavaa itselle!

Projektin kokonaistyömäärä ei varsinaisesti yllättänyt, se oli tietona jo projektiin ilmoittautuessa. Kevään aikana joutuinkin joustamaan muissa opinnoissa jonkin verran. Dokumenttien vaatimukset eivät myöskään varsinaisesti yllättäneet, sen sijaan hieman yllätyin alussa siitä, miten täysin vetovastuu siirtyy ryhmälle heti projektin alussa. Tähänkin onneksi tottui pian ja vastuun kantaminen valmentaa varmasti parhaiten jatkoa ajatellen.

6.2.2 Mitä opin

Ohjelmointitaitoni kehittyivät paljon, vaikka näin nopeassa ajassa ei valtavia harppauksia otetakaan. Kuitenkin huomasi sen, että tekemällä, kokeilemalla ja virheistä oppimalla kehittyy. Jonnen kommentti siitä, että tois-

ten tuottamista koodeista oppii eniten, pitää varmasti paikkaansa ainakin minun kohdalla. Matin ammattimainen ote oli erittäin opettavainen, hän myös neuvoi eri vaiheissa paljon.

Projektityöskentelystä yleensä kaikki oli minulle uutta, joten oppia kertyi valtavasti. Oppia tuli myös lisää Linux-käyttöjärjestelmästä ja ennen kaikkea L^AT_EX-dokumenttien laadinnasta. L^AT_EXon todella mainio tapa luoda laadukkaita dokumentteja. OpenGL tuli tutuksi, kun aikaisemmin se oli aivan vieras, jatkossa kynnys sen käyttämiseen on huomattavasti alempana.

6.2.3 Mitä tekisin toisin

Projektin alussa pyrkisin huomattavasti enemmän “omalla ajalla” paneutumaan OpenGL:n ominaisuuksiin ja pyrkisin kokeilemaan erilaisia juttuja sen avulla. Täten olisimme voineet jakaa paremmin vastuuta visualisointitoiminnoista. Periaatteessa kaiken voi aina tehdä laadukkaammin ja paremmin, mutta koen silti että kykenin antamaan omat panokseni tämän projektin eteen.

6.3 Harri Kosunen

Projekti oli minulle uusi ja raskas, mutta kuitenkin hieno kokemus, josta tulee varmasti olemaan myös hyötyä jatkossa. Projektin alkuvaiheessa tuntui sille, etten oikein muistanut mitään ohjelmoinnista ja olinkin epäluuloinen siitä, että näinköhän taitoni riittäisivät tämän projektin suorittamiseen. Kuitenkin projektin edetessä huomasin oppivani paljon uutta ohjelmoinnista ja etenkin Matti oli suurena apuna oppimisessa ja työn tekemisessä. Ohjelmointikieli ja työkalut olivat jo ennestään tuttuja OpenGL:ää lukuunottamatta. Projektin alkuvaiheessa yritinkin löytää netistä tietoa mahdollisimman paljon liittyen juuri OpenGL-ohjelmointiin. Rainerin antama tutoriaali ohjelmointivaiheen alussa selvensi myös hyvin käsityksiä kyseisen kirjaston käytöstä.

6.3.1 Ongelmat ja yllätykset

Omana miinuspuolena projektissa voisin kuitenkin pitää sitä, että alkuinnostuksen laannuttua ja huomattuani, että Matti tekee saman asian parissa tunnissa kuin mihin minulla menee koko päivä, visualisoinnin teko siirtyi kokonaan Matin harteille. Nyt projektin lopussa en osakaan välttämättä käyttää OpenGL:ää niin hyvin kun mitä alkuvuodesta ajattelin.

Tosin sovellussuunnitelman selkiydyttyä aloin ymmärtämään Matin tekemää koodia jo paljon paremmin. Täytyy siis vain yrittää nyt itsenäisesti opiskella asia oikealle tasolle.

Projektin aikana tutustuin myös ensimmäistä kertaa Linux-käyttöjärjestelmään ja kirjoitin ensimmäiset dokumentit käyttäen \LaTeX ia. Nämä molemmat seikat aiheuttivat hieman lisätyötä verrattuna siihen, että jos projekti olisi toteutettu kokonaan Windows-käyttöjärjestelmällä ja Wordilla.

Aikataulun suunnittelu ja dokumenttien sisältö oli minulle myös ennen projektia täysin vierasta ja nämä seikat aiheuttivatkin myös koko projektille ongelmia suunnitteluvaiheessa, sillä sovellussuunnitelmaa ei saatu ohjaajia miellyttävälle tasolle ajoissa.

6.3.2 Mitä opin

Opin todellakin paljon uutta. Perusasioina Linuxin käyttöä ja \LaTeX -dokumenttien kirjoittamista. Projektityöskentelyn taitoja kuten palaverikäytännöt, suunnittelut ja dokumentoinnit. Opin myös luottamaan omaan ohjelmointitaitooni ja siihen, että mitä en vielä osaa, niin voin oppia. Opin myös perusteet 3D-ohjelmoinnista OpenGL:llä ja tuota taitoa aionkin varmasti kehittää.

6.3.3 Mitä tekisin toisin

Aloittaisin projektin työstämisen tehokkaammin ja kuormittaisin ohjaajia enemmän ”tyhmillä” kysymyksillä. Opettelisin kaikki projektiin liittyvät asiat kunnolla riippumatta siitä tarvitseeko minun toteuttaa kyseistä osaluuetta, esimerkiksi 3D-ohjelmointi olisi pitänyt opetella paremmin, vaikka vastuu siitä olikin Matilla.

6.4 Riikka Rikkola

Sovellusprojekti oli kokemuksena uusi ja sen mukana tullut työmäärä ylätti. Projekti kieltämättä aiheutti välillä hampaiden kiristystä ja harmaita hiuksia, mutta toisaalta olo oli aina helpottunut ja tyytyväinen, kun jotain saatiin valmiiksi. Projekti oli taatusti kokemuksena vertaansa vailla ja sen hyöty on kiistämätön. Jokaisen ryhmän jäsenen kanssa oli mukava työskennellä ja erityisesti Matille haluan nostaa hattua. Matti kantoi kokemuksensa ja taitojensa takia suurimman vastuun toteutuksesta ja jaksoi lisäksi kärsivällisesti neuvoa ja vastaila heränneisiin kysymyksiin.

6.4.1 Ongelmat ja yllätykset

Laadittavien dokumenttien määrä ja niihin uppoavien työtuntien määrä yllätti. Varsinkin sovellusuunnitelman teko osoittautui hankalaksi. Suuremman kokonaisuuden näkeminen oli vaikeaa ja aikaisempaa kokemusta tarkkojen suunnitelmien laatimisesta ei ollut. Lisäksi joskus pilkkujen viilaaminen, erityisesti alkupalavereissa, tuntui varsin turhauttavalta. Kysymysten ja ongelmien ratkaisussa Matti oli korvaamaton apu.

6.4.2 Mitä opin

Object pascal muistui takaisin mieleen ja opin sillä saralla myös paljon uutta. Dokumenttien laatimisessa käytetty \LaTeX oli minulle entuudestaan täysin vieras ja sen opettelu oli todella mieluisaa. Myös perusteet 3D-ohjelmoinnista OpenGL-kirjastoa hyödyntäen tulivat tutuiksi, mutta perehtyminen olisi voinut olla paljon syvällisempääkin. Halua perusteellisempaan oppimiseen olisi kyllä ollut, mutta aikaa sen sijaan ei.

Opponoinneista oli mielestäni todellista hyötyä. Palaute omasta esiintymisestä oli todella tervetullutta ja opponointien avulla pystyi totuttelemaan esiintymiseen. Lisäksi projektityöskentely oli minulle kokemuksena aivan uusi. Suunnittelu ja raportointi saivat aivan uuden merkityksen ja palaverikäytännötkin tuli opittua.

6.4.3 Mitä tekisin toisin

Ehkä jälkeinpäin ajateltuna ryhmän työnjakoa ei olisi tarvinnut tehdä heti alussa niin "jyrkäksi". Suunnitteludokumentit olisi voinut yhdessä hoitaa ensin alta pois ja sen jälkeen alkaa tarkemmin jakamaan vastuualueita. Lisäksi joitakin vastuualueita, esimerkiksi projektipäällikön pestiä, tiedottamista ja sihteerinä olemista, olisi voinut kierrättää enemmän. Tällöin jokainen olisi päässyt kokeilemaan taitojaan eri osa-alueilla. Tehtävien kierrätys olisi tuonut mukanaan vaihtelevuutta ja tätä kautta pitänyt motivaation korkealla.

Lisäksi itse aloittaisin projektin tehokkaammin. Koska projekti oli kokemuksena uusi, tuntui siltä, että alku meni hämmennyksen parissa. Tiesi hyvin, että oli kiire ja paljon tehtävää, mutta ei oikein aina tiennyt mistä tulisi ensimmäisenä aloittaa. Myös OpenGL:n käyttöön haluaisin perehtyä perusteellisemmin.

7 Yhteenveto

Tämä dokumentti käsitteli Jyväskylän yliopiston tietotekniikan laitoksella vuonna 2003 toteutettua Hibbo-sovellusprojektia. Etenkin pyrittiin vertaamaan laadittuja suunnitelmia ja saavutettuja lopputuloksia toisiinsa. Seuraavassa aliluvussa käsitellään vielä tiivistetysti Hibbo-projektin onnistumista ja annetaan projektiryhmäläisten kokemuksiin perustuvia ohjeita tuleviin sovellusprojekteihin.

7.1 Yleiskuva

Neljästä opiskelijasta koostuva Hibbo-projektiryhmä suunnitteli, toteutti ja dokumentoi Jyväskylän yliopiston fysiikan laitokselle graafisen käyttöliittymän ja visualisointitoiminnot hila-Boltzmann-simulaattoriin. Projekti toteutettiin tietotekniikan laitoksen ohjaamana ja valvomana.

Projekti saatettiin päätökseen likipitään suunniteltuna ajankohtana; alkuperäisestä suunnitelmasta saada projekti päätökseen loppuesittelypäivänä jouduttiin taipumaan vajaat kaksi viikkoa projektin päättyessä 28.5.2003.

8 Lähteet

- [1] Eskelinen Matti, Karppinen Olli, Kosunen Harri ja Rikkola Riikka, "Hibbo-projektin vaatimusmäärittely", Jyväskylän yliopisto, tietotekniikan laitos, saatavilla WWW-muodossa
<URL:<http://kotka.it.jyu.fi/hibbo/vaatimusmaarittely/vaatimusmaarittely.pdf>>, viitattu 12.5.2003.
- [2] Eskelinen Matti, Karppinen Olli, Kosunen Harri ja Rikkola Riikka, "Hibbo-projektin projektisuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, saatavilla WWW-muodossa
<URL:<http://kotka.it.jyu.fi/hibbo/projektisuunnitelma/projektisuunnitelma.pdf>>, viitattu 12.5.2003.
- [3] Eskelinen Matti, Karppinen Olli, Kosunen Harri ja Rikkola Riikka, "Hibbo-projektin sovellussuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, saatavilla WWW-muodossa
<URL:<http://kotka.it.jyu.fi/hibbo/sovellussuunnitelma/sovellussuunnitelma.pdf>>, viitattu 12.5.2003.
- [4] Eskelinen Matti, Karppinen Olli, Kosunen Harri ja Rikkola Riikka, "Hibbo-projektin testausuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, saatavilla WWW-muodossa
<URL:<http://kotka.it.jyu.fi/hibbo/testausuunnitelma/testausuunnitelma.pdf>>, viitattu 12.5.2003.
- [5] Eskelinen Matti, Karppinen Olli, Kosunen Harri ja Rikkola Riikka, "Hibbo-projektin sovellusraportti", Jyväskylän yliopisto, tietotekniikan laitos, saatavilla WWW-muodossa
<URL:<http://kotka.it.jyu.fi/hibbo/sovellusraportti/sovellusraportti.pdf>>, viitattu 12.5.2003.
- [6] Eskelinen Matti, Karppinen Olli, Kosunen Harri ja Rikkola Riikka, "Hibbo-projektin testausraportti", Jyväskylän yliopisto, tietotekniikan laitos, saatavilla WWW-muodossa
<URL:<http://kotka.it.jyu.fi/hibbo/testausraportti/testausraportti.pdf>>, viitattu 12.5.2003.
- [7] Eskelinen Matti, Karppinen Olli, Kosunen Harri ja Rikkola Riikka, "Hibbo-sovelluksen käyttöohjeet", Jyväskylän yliopisto, tietotekniikan laitos, saatavilla WWW-muodossa

- <URL:<http://kotka.it.jyu.fi/hibbo/ohjeet/kayttoohje.pdf>>, viitattu 12.5.2003.
- [8] Eskelinen Matti, Karppinen Olli, Kosunen Harri ja Rikkola Riikka, "Hibbo-sovelluksen asennusohjeet", Jyväskylän yliopisto, tietotekniikan laitos, saatavilla WWW-muodossa
<URL:<http://kotka.it.jyu.fi/hibbo/asennusohjeet/asennusohjeet.pdf>>, viitattu 12.5.2003.
 - [9] Eskelinen Matti, Karppinen Olli, Kosunen Harri ja Rikkola Riikka, "Hibbo-projektin 1. opponointiraportti", Jyväskylän yliopisto, tietotekniikan laitos, saatavilla WWW-muodossa
<URL:<http://kotka.it.jyu.fi/hibbo/Opponoinnit/Oppis.pdf>>, viitattu 12.5.2003.
 - [10] Eskelinen Matti, Karppinen Olli, Kosunen Harri ja Rikkola Riikka, "Hibbo-projektin 2. opponointiraportti", Jyväskylän yliopisto, tietotekniikan laitos, saatavilla WWW-muodossa
<URL:<http://kotka.it.jyu.fi/hibbo/Opponoinnit/Raportti.pdf>>, viitattu 12.5.2003.
 - [11] Eskelinen Matti, Karppinen Olli, Kosunen Harri ja Rikkola Riikka, "Hibbo-projektin projektiraportti", Jyväskylän yliopisto, tietotekniikan laitos, saatavilla WWW-muodossa
<URL:<http://kotka.it.jyu.fi/hibbo/projektiraportti/projektiraportti.pdf>>, viitattu 12.5.2003.
 - [12] OpenGL-tutorials <URL:<http://nehe.gamedev.net>>, viitattu 12.5.2003.
 - [13] Jyväskylän yliopiston fysiikan laitoksen WWW-sivut, osoitteessa
<URL: <http://www.phys.jyu.fi/research/dismat/index.html>>, viitattu 13.3.2003.
 - [14] Shemitz Jon, "Kylix: The Professional Developer's Guide and Reference", Apress 2002.
 - [15] Swan Tom, "Delphi 4", Teknolit Oy 1999.
 - [16] IEEE Recommended Practice for Software Requirements Specifications, saatavilla WWW-muodossa
<URL:http://www.cs.rit.edu/~wmc/srs_pdf.pdf>, viitattu 13.5.2003.