

HIBBO

Tietotekniikan sovellusprojekti

Matti Eskelinen
Olli Karppinen
Harri Kosunen
Riikka Rikkola

Vaatimusmäärittely
Versio: 1.0
25.3.2003

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: Vaatimusmäärittely tietotekniikan sovellusprojektiin

Työtila: Agora, huone AgC223.3, puhelinnumero 014-260 4965

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

Tiivistelmä

Tämä dokumentti on Jyväskylän yliopistossa keväällä 2003 toteutettavan Hibbo-projektin vaatimusmäärittely. Dokumentissa kuvataan projektin tuotettaman sovelluksen käyttäjät, käyttötapaukset, yhteydet muihin sovelluksiin ja tiedostoihin sekä järjestelmän vaatimukset.

Avainsanat

Tietotekniikan sovellusprojekti, fysiikan laitos, hila-Boltzmann, simulointi, graafinen käyttöliittymä, visualisointi, OpenGL

Dokumentin versiohistoria

Versio	Päivämäärä	Tekijät	Kuvaus
1.0 -1	5.3.2003	ME	Alustava versio.
1.0 -2	7.3.2003	ME, HK	Korjailtu, lisätty käyttötapaukset ja esimerkitiedostot.
1.0 -3	13.3.2003	ME	Tehty korjauksia ja lisäyksiä käyttötapauksiin ja vaatimuksiin.
1.0	25.3.2003	ME	Tehty vaaditut korjaukset.

Tekijöiden lyhenteet

ME Matti Eskelinen

OK Olli Karppinen

HK Harri Kosunen

RR Riikka Rikkola

Sisältö

1	Johdanto	1
1.1	Päämäärä	1
1.2	Kattavuus	1
1.3	Tausta	1
1.4	Sisältökatsaus	2
2	Termistö	3
3	Sovelluksen yleiskuvaus	4
3.1	Sovellusympäristö	4
3.2	Käyttäjät	4
3.3	Käyttötapaukset	4
3.3.1	Asetusten muuttaminen	6
3.3.2	Näytteen luominen	6
3.3.3	Näytteen lataaminen	7
3.3.4	Näytteen tutkiminen	7
3.3.5	Simuloinnin käynnistäminen	7
3.3.6	Simulointituloksen lataaminen	8
3.3.7	Simuloinnin keskeyttäminen	8
3.3.8	Simulointituloksen tutkiminen	9
3.3.9	Kuvien tallentaminen	9
3.3.10	Käyttöliittymän sulkeminen	10
4	Sovelluksen vaatimukset	11
4.1	Rajapinnat	11
4.1.1	Käyttäjäraja- pinta	11
4.1.2	Näytteenluontiraja- pinta	11
4.1.3	Simulointiraja- pinta	12
4.1.4	Tulostiedosto	13
4.1.5	Aikakehitystiedosto	15
4.1.6	Kenttätiedosto	16
4.1.7	Näytetiedosto	16
4.1.8	Projektitiedosto	17
4.2	Järjestelmävaatimukset	17
4.2.1	Ohjelman käynnistäminen ja sulkeminen	17
4.2.2	Asetusten muuttaminen	17
4.2.3	Näytteen luominen	18
4.2.4	Näytteen lataaminen	18
4.2.5	Näytteen tutkiminen	18

4.2.6	Simulointi	18
4.2.7	Simuloinnin keskeyttäminen	19
4.2.8	Simulointituloksen lataaminen	19
4.2.9	Simuloinnin jatkaminen	19
4.2.10	Simulointituloksen tutkiminen	20
4.2.11	Kuvien tallentaminen	21
4.2.12	Lokitietojen tarkastelu	22
4.3	Tehokkuusvaatimukset	22
4.4	Suunnittelurajoitteet	22
5	Yhteenveto	24

1 Johdanto

Hibbo-projekti on Jyväskylän yliopiston tietotekniikan laitoksen kevään 2003 sovellusprojekti. Projekti toteuttaa Jyväskylän yliopiston fysiikan laitokselle hila-Boltzmann-simulaattorin graafisen käyttöliittymän, jonka yksi olennainen osa on laskentatulosten visualisointi.

Varsinainen simulointiohjelma on kehitetty jo aiemmin fysiikan laitoksella. Se toimii komentorivipohjaisesti ja tulosten visualisointityökälyt puuttuvat. Hibbo-projektiin sisältyy ainoastaan olemassaolevaa simulointiohjelmaa hyödyntävä graafinen käyttöliittymä visualisointitoimintoinen.

1.1 Päämäärä

Tämän vaatimusmäärittelydokumentin tarkoituksena on kuvata Hibbo-projektin toteuttaman sovelluksen vaatimukset ja määrittelyt. Se toimii sovelluksen toteuttajien ohjeena heidän suunnitellessaan sovellusta sekä sopimuksena sovelluksen tilaajien kanssa. Tämä dokumentti on kohdennettu Hibbo-projektiryhmän jäsenille sekä sovelluksen tilaajille.

1.2 Kattavuus

Tässä dokumentissa kuvataan vaatimukset sovellukselle nimeltä Hibbo. Sovellus on hila-Boltzmann-menetelmällä tuotettujen virtaussimulaatiotulosten visualisointityökalu ja sen avulla voidaan käynnistää simulaation laskennan suorittava ohjelma tai laskentamallin luova ohjelma. Sovellusta itseään ei voi käyttää minkäänlaiseen laskentaan tai tulosten analysointiin tai raportointiin, ainoastaan tulosten visualisointiin.

Hibbo-sovellusta on tarkoitus käyttää apuna opetuksessa. Sovelluksen avulla opiskelijat voivat tehdä pienimuotoista tutkimusta itsenäisesti työskennellen. Ohjelman tulee siis olla helppokäyttöinen ja visualisointien havainnollisia. Sovelluksen täytyy kuitenkin olla niin luotettava ja monipuolinen, että sillä voi tehdä laajempaakin tutkimusta.

1.3 Tausta

Hila-Boltzmann on tehokas, diskreetti menetelmä monenlaisten monimutkaisten nestevirtausongelmien mallintamiseen. Menetelmässä mallinnettava alue jaetaan säännölliseksi kuutiopilaksi, mistä nimikin tulee; nestettä käsitellään joukkona partikkeleja, joita edustaa nopeusvektori jokaisessa hilapisteessä.

Laskenta-alue on siis jaettu säännöllisiksi kuutioiksi. Jokainen kuutio on ns. laskentakoppi ja voi olla joko nestettä tai kiinteää ainetta. Jokaisessa nestekuutiossa ratkaistaan nesteen virtaus hila-Boltzmann-algoritmillä ja tuloksena saadaan nesteen nopeusvektori ja paine jokaisessa laskentakopissa.

1.4 Sisältökatsaus

Jäljempänä tässä dokumentissa kuvataan tarkemmin Hibbo-sovellusta. Dokumentti pyrkii selostamaan yksiselitteisesti, mitä Hibbo-sovellus on ja mitä se ei ole. Luvussa 3 kerrotaan yleisiä asioita sovelluksesta, sen toiminnosta ja käyttäjistä. Luvussa 4 kuvataan sovelluksen rajapinnat sekä määritellään tarkat vaatimukset. Lopuksi luvussa 5 esitetään yhteenveto sovelluksen vaatimuksista.

2 Termistö

Seuraavassa määritellään joitakin tässä dokumentissa käytettäviä olennaisia termejä.

Hibbo on sovellus, jota tämä dokumentti käsittelee ja jonka Hibbo-projekti toteuttaa. Hibbosta käytetään myös nimityksiä Hibbo-sovellus tai pelkkä sovellus.

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

Käyttötapaus on käyttäjän tai sovelluksen toiminnan kuvaus tietyn tehtävän suorittamiseksi.

Näyte on geometrinen malli kohteesta, jossa tapahtuvia virtauksia simuloidaan. Käytännössä näyte on kolmiulotteinen taulukko hilapisteitä, joihin liittyy tieto siitä, ovatko ne nestettä vai kiinteää ainetta.

Näytetiedosto on tietyn muotoinen tiedosto, jossa on tallennettuna näytteen sisältämä tieto. Tiedoston sisältö on kuvattu luvussa 4.1.7.

Näytteenluontiohjelma on fysiikan laitoksen toteuttama ohjelma, joka luo halutun kokoisen, tietyt vaatimukset täyttävän näytteen, ja tallentaa sen tiedostoon.

Projekti on kokonaisuus, joka sisältää kaikki tarvittavat tiedot tietystä simuloinnista, kuten siihen liittyvät tiedostot ja asetukset.

Simulointitulokset on hila-Boltzmann-algoritmilla tuotettu tieto nesteen virtauksista näytteessä. Käytännössä tulos on kolmiulotteinen taulukko hilapisteitä, joihin liittyy tieto nesteen nopeusvektorista ja paineesta kyseisessä pisteessä.

Simulointiohjelma on fysiikan laitoksen toteuttama ohjelma, joka laskee simulointituloksen annetussa näytteessä ja tallentaa simulointituloksen sekä joitakin muita simuloinnin kulkua kuvaavia tietoja tiedostoihin.

3 Sovelluksen yleiskuvaus

Tässä luvussa kerrotaan yleisiä asioita Hibbo-sovelluksesta. Luvussa 3.1 kerrotaan ympäristöstä, jossa sovellus toimii, sekä sovelluksen yhteyksistä muihin ohjelmiin. Luvussa 3.2 kuvataan tyypilliset käyttäjät, joille sovellus on suunniteltu, ja luvussa 3.3 listataan sovelluksen tärkeimmät toiminnot ja selostetaan niihin liittyvät käyttötapaukset.

3.1 Sovellusympäristö

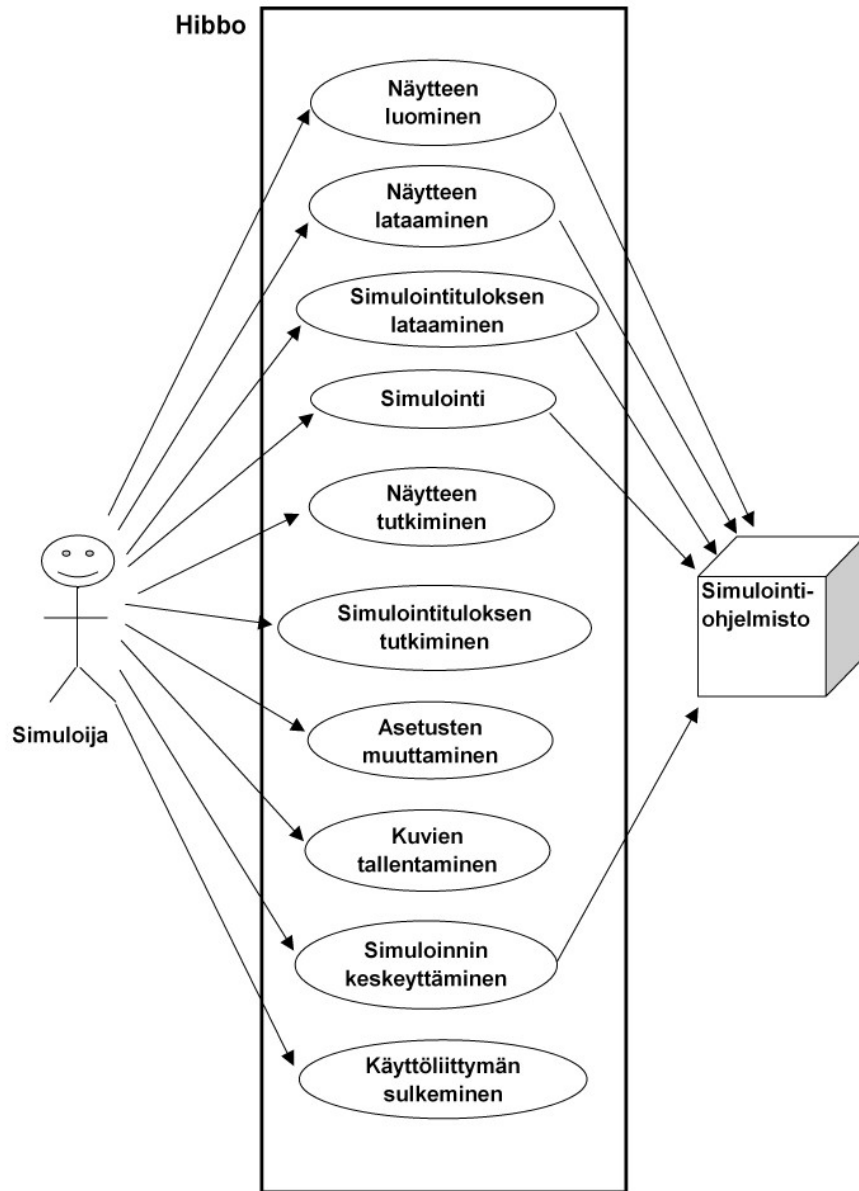
Hibbo-sovellus on itsenäinen ohjelma, mutta se käyttää kahta erillistä fyysikan laitoksen ohjelmaa. Toinen on simulointiohjelma, joka suorittaa hila-Boltzmann-simulaation laskennan, ja toinen näytteenluontiohjelma, joka luo näytteen geometrian. Näiden ohjelmien rajapinnat on kuvattu luvuissa 4.1.3 ja 4.1.2.

3.2 Käyttäjät

Hibbo-sovelluksella on vain yksi käyttäjäryhmä, johon kuuluvia käyttäjiä kutsutaan nimellä simuloija. Simuloija on opiskelija tai tutkija, joka tuntee käytettävän laskentamenetelmän periaatteet sekä aiheeseen liittyvät fysikaaliset perusteet. Välttämättä simuloijalla ei ole syvällistä tietokoneosaamista, joten ohjelman käyttöliittymä suunnitellaan yleisten käytäntöjen mukaisesti ja siitä pyritään tekemään havainnollinen.

3.3 Käyttötapaukset

Sovelluksella on muutamia päätoimintoja. Seuraavassa on selostettu nämä toiminnot sekä niihin liittyvät käyttötapaukset. Toimintoihin liittyvät tarkat vaatimukset on selostettu luvussa 4.2. Käyttötapaukset on esitetty kaaviona kuvassa 1.



Kuva 1: Käyttötapauskaavio.

3.3.1 Asetusten muuttaminen

Suorittaja: Simuloija.

Edellytykset: -

Kuvaus: Simuloija muuttaa projektin asetuksia haluamallaan tavalla tai palauttaa käyttöön oletusasetukset, ja tallentaa tekemänsä muutokset. Halutessaan simuloija voi peruuttaa tekemänsä muutokset, jolloin vanhat asetukset jäävät voimaan. Simuloija voi myös tallentaa nykyiset asetukset oletusasetuksiksi, jotka otetaan käyttöön uutta projektia luotaessa, tai kun oletusasetukset palautetaan olemassa olevaan projektiin.

Lopputilanne: Muutetut asetukset on tallennettu, tai toiminto on peruutettu.

Poikkeukset: -

3.3.2 Näytteen luominen

Suorittaja: Simuloija.

Edellytykset: Näytteenluontiohjelma on käytettävissä.

Kuvaus: Jos nykyisessä projektissa on jo käytössä jokin näyte, simuloijalta kysytään, haluaako hän korvata sen. Jos näytettä ei haluta korvata, toiminto keskeytyy. Mutta jos nykyisessä projektissa on jo simulointitulokset, näytettä ei saa vaihtaa. Tässä tapauksessa simuloijalle ilmoitetaan asiasta ja kehoitetaan luomaan uusi projekti. Simuloija syöttää luvussa 4.1.2 esiteltyt parametrit. Jos parametrien arvot ovat virheellisiä, simuloijaa huomautetaan tästä ja pyydetään korjaamaan arvot. Syötettyään parametrit simuloija pyytää sovellusta käynnistämään mallinluontiohjelman. On mahdollista, että prosessin käynnistys ei onnistu. Tällöin simuloijalle ilmoitetaan asiasta, jolloin hän voi peruuttaa toiminnon tai yrittää uudestaan. Simuloija voi milloin tahansa peruuttaa toiminnon, jolloin palataan edelliseen tilanteeseen käynnistämättä mallinluontia.

Lopputilanne: Näytteenluontiohjelma on käynnistetty erillisenä prosessina, tai toiminto on peruutettu.

Poikkeukset: Käytössä on jo näyte. Käytössä on jo simulointitulokset. Simuloija syöttää virheellisiä parametrien arvoja. Prosessin käynnistys epäonnistuu.

3.3.3 Näytteen lataaminen

Suorittaja: Simuloija.

Edellytykset: Levyllä on olemassa aiemmin luotu näytetiedosto.

Kuvaus: Simuloija etsii levyiltä haluamansa näytetiedoston ja pyytää sovellusta lataamaan näytteen käyttöön. Jos käytössä on jo näyte, simuloijalta kysytään, korvataanko se. Jos simuloija ei halua korvata näytettä, toiminto keskeytyy. Jos käytössä on jo simulointitulos, näytettä ei saa vaihtaa. Tässä tapauksessa simuloijalle ilmoitetaan asiasta ja kehoitetaan luomaan uusi projekti. Muuten sovellus lukee näytetiedoston ja lataa sen kuvaaman näytteen käyttöön nykyiseen projektiin. Simuloija voi myös milloin tahansa keskeyttää toiminnon, jolloin palataan edelliseen tilanteeseen lataamatta näytettä.

Lopputilanne: Näytetiedosto on ladattu käyttöön ja se on simuloijan visualisoitavissa, tai toiminto on peruutettu.

Poikkeukset: Käytössä on jo näyte. Käytössä on jo simulointitulos. Tiedostonluku epäonnistuu.

3.3.4 Näytteen tutkiminen

Suorittaja: Simuloija.

Edellytykset: Jokin näyte on ladattu ja käytössä.

Kuvaus: Simuloija tarkastelee näytettä visualisointinäkyvän avulla, siirrellen ja pyöritellen sitä.

Lopputilanne: -

Poikkeukset: -

3.3.5 Simuloinnin käynnistäminen

Suorittaja: Simuloija.

Edellytykset: Simulointiohjelma on käytettävissä.

Kuvaus: Jos nykyisessä projektissa on jo suoritettu simulointi, simuloijalta kysytään, haluaako hän jatkaa edellistä simulointia. Jos edellinen simulointi keskeytyi tai keskeytettiin, simuloijalle kerrotaan tästä. Jos simulointia ei haluta jatkaa, toiminto keskeytyy. Simuloija syöttää

luvussa 4.1.3 esitellyt parametrit. Jos jatketaan vanhaa simulointia, vanhat parametrien arvot ovat parametrien oletusarvoina. Jos parametrien arvot ovat virheellisiä, simuloijaa huomautetaan tästä ja pyydetään korjaamaan arvot. Syötettyään parametrit simuloija pyytää sovellusta käynnistämään simulointiohjelman. On mahdollista, että prosessin käynnistys ei onnistu. Tällöin simuloijalle ilmoitetaan asiasta, jolloin hän voi peruuttaa toiminnon tai yrittää uudestaan. Simuloija voi milloin tahansa peruuttaa toiminnon, jolloin palataan edelliseen tilanteeseen käynnistämättä simulointia.

Lopputilanne: Simulointiohjelma on käynnistetty erillisenä prosessina, tai toiminto on peruutettu.

Poikkeukset: Simulointi on jo suoritettu. Simuloija syöttää virheellisiä parametrien arvoja. Prosesin käynnistys epäonnistuu.

3.3.6 Simulointituloksen lataaminen

Suorittaja: Simuloija.

Edellytykset: Levyllä on olemassa aiemman simulaation tulostiedostot.

Kuvaus: Simuloija etsii levyiltä haluamansa tulostiedostot ja pyytää sovellusta lataamaan simulointitulokset käyttöön. Jos nykyisessä projektissa on jo simulointi tai näyte käytössä, simuloijalta kysytään, haluaako hän luoda uuden projektin näitä simulointituloksia varten. Jos simuloija ei halua luoda uutta projektia, toiminto keskeytyy. Muuten sovellus luo tarvittaessa uuden projektin ja lukee simulointitulokset käyttöön. Simuloija voi milloin tahansa keskeyttää toiminnon, jolloin palataan edelliseen tilanteeseen lataamatta tuloksia.

Lopputilanne: Aikaisemman simuloinnin tulokset on ladattu ja ne ovat simuloijan visualisoitavissa, tai toiminto on peruutettu.

Poikkeukset: Käytössä on jo simulointitulos. Tiedoston luku epäonnistuu.

3.3.7 Simuloinnin keskeyttäminen

Suorittaja: Simuloija.

Edellytykset: Simulointiohjelma on käynnissä.

Kuvaus: Simuloija pyytää sovellusta keskeyttämään simuloinnin. Simuloijalta varmistetaan, haluaako hän todella tehdä näin. Jos vastaus on myönteinen, pyydetään simulointiohjelmaa keskeyttämään simulointi ja tallentamaan tänänsäiset tulokset. Muuten toiminto keskeytetään lähettämättä keskeytyspyyntöä. On mahdollista, että pyynnön lähettäminen ei onnistu. Tällöin simuloijalle ilmoitetaan asiasta, jolloin hän voi peruuttaa toiminnon tai yrittää uudestaan.

Lopputilanne: Simulointiohjelmalle on lähetetty pyyntö simuloinnin keskeyttämisestä, tai toiminto on peruutettu.

Poikkeukset: Pyyntön lähettäminen ei onnistu.

3.3.8 Simulointituloksen tutkiminen

Suorittaja: Simuloija.

Edellytykset: Simulointitulos on ladattu ja käytössä.

Kuvaus: Simuloija voi tarkastella näytettä visualisointinäkymän avulla. Hän voi myös pyytää sovellusta luomaan 2d-leikkeen haluamaltaan kohdalta haluamaansa suuntaan ja asettaa visualisoitavan suureen. Sovellus luo leikekuvan visualisointinäkymään, jonka avulla simuloija voi siirrellä ja pyöritellä leikekuvaa. Simuloija voi myös pyytää sovellusta esittämään virtaviivakuvan määräämällä tason, jolta virtaviivat lähtevät, ja viivojen määrän sekä värikoodauksen. Sovellus luo virtaviivakuvan, jota simuloija voi tarkastella kuten edellä. Lopuksi simuloija voi vielä pyytää sovellusta luomaan partikkelianimaation määräämällä tason, jolta partikkelit lähtevät liikkeelle, ja partikkelien määrän. Sovellus luo animaationäkymän, jota simuloija voi tarkastella kuten edellä.

Lopputilanne: -

Poikkeukset: -

3.3.9 Kuvien tallentaminen

Suorittaja: Simuloija.

Edellytykset: Jokin näkymä näytteeseen tai simulointituloksiin on visualisoituna.

Kuvaus: Simuloija pyytää sovellusta tallentamaan nykyisen näkymän kuvaksi. Sovellus tallentaa kuvan käyttäen simuloijan asetuksissa määräämää perusnimeä ja lisäämällä juoksevan numeron. Kuvatiedoston kirjoittaminen voi epäonnistua. Tällöin simuloijalle ilmoitetaan asiasta, jolloin hän voi yrittää kuvan tallentamista uudestaan.

Lopputilanne: Levylle on tallennettu kuva tai kuvia.

Poikkeukset: Tiedostoon ei voi kirjoittaa.

3.3.10 Käyttöliittymän sulkeminen

Suorittaja: Simuloija.

Edellytykset: -

Kuvaus: Simuloija pyytää sovellusta sulkeutumaan. Simuloijalta kysytään varmistuksena, haluaako hän todella lopettaa ohjelman käytön ja haluaako hän tallentaa projektin tiedot. Jos simuloija ei halua lopettaa ohjelmaa, toiminto keskeytetään. Muuten, jos simuloija haluaa tallentaa tiedot, ne tallennetaan. Tietojen tallennus voi epäonnistua, jolloin simuloijalle ilmoitetaan asiasta ja hän voi yrittää uudestaan, peruuttaa ohjelman lopetuksen tai lopettaa tallentamatta. Jos näytteenluonti- tai simulointiprosessi on käynnissä, simuloijalta kysytään, haluaako hän keskeyttää prosessit. Jos vastaus on myöntävä, prosessit suljetaan. Muuten prosessit jäävät pyörimään ja käyttöliittymä suljetaan.

Lopputilanne: Käyttöliittymä on suljettu, tai toiminto on peruutettu.

Poikkeukset: -

4 Sovelluksen vaatimukset

Seuraavassa tarkastellaan sovellukselle asetettuja vaatimuksia. Luvussa 4.1 kuvataan sovelluksen rajapinnat muihin ohjelmistoihin ja resursseihin. Luvussa 4.2 kuvataan sovelluksen tarkat vaatimukset.

4.1 Rajapinnat

Tässä luvussa selostetaan sovelluksen käyttämät rajapinnat muihin ohjelmiin ja resursseihin.

4.1.1 Käyttäjäraja-pinta

Sovelluksen käyttäjäraja-pintana toimii käyttöliittymä, joka on samanlainen kaikille käyttäjille. Käyttöliittymä tarjoaa käyttäjälle mahdollisuuden käynnistää näytteenluonti- tai simulointiohjelman ja syöttää ohjelmille niiden tarvitsemat parametrit. Käyttöliittymä myös näyttää tietoja käynnissä olevasta simuloinnista. Tärkein käyttöliittymän ominaisuus on mahdollisuus tarkastella näytettä ja simulointituloksia visualisoituina.

4.1.2 Näytteenluontiraja-pinta

Sovellus käyttää erillistä ohjelmaa näytteen geometrian luontiin. Tässä luvussa kuvataan kyseisen ohjelman raja-pinta eli ohjelman ottamat parametrit ja niiden sallitut arvot. Ohjelma ottaa seuraavat parametrit:

Näytetiedosto

Tiedosto, johon luotu näyte tallennetaan.

Sallitut arvot: tiedostonimi ilman tarkenninta

Näytteen koko

Määrää luotavan kuutiohilan koon kolmessa ulottuvuudessa. Sisältää koon x-, y- ja z-suunnissa.

Sallitut arvot: kolme kokonaislukua väliltä 10-1000

Pallon säde

Ohjelma arpoo hilaan tietyn kokoisia palloja. Tämä parametri kertoo pallojen säteen.

Sallitut arvot: kokonaisluku väliltä 1-100

Huokoisuus

Kertoo, millainen huokosien ja kiinteän aineen tilavuuden suhteen täytyy olla valmiissa mallissa *Sallitut arvot*: reaalityluku väliltä 0-1

4.1.3 Simulointirajapinta

Sovellus käyttää erillistä ohjelmaa simulaatiotulosten laskemiseen. Tässä luvussa kuvataan kyseisen ohjelman rajapinta eli ohjelman ottamat parametrit ja niiden sallitut arvot. Ohjelma ottaa seuraavat parametrit:

Näytetiedosto

Sisältää näytteen laskentageometrian.

Sallitut arvot: tiedostonimi, tarkennin .sample

Tulostiedosto

Perusnimi josta johdetaan tulostiedostojen nimet eri tarkentimin.

Sallitut arvot: tiedostonimi, ilman tarkenninta

Relaksaatioparametri, τ

Määrää simulointinesteen viskositeetin.

Sallitut arvot: reaalityluku, $0.5 < \tau \leq 1$

Aika-askeleet, t

Määrää, kuinka monta kierrosta algoritmia suoritetaan.

Sallitut arvot: kokonaisluku väliltä 1-100 000

Alkunopeus, v_0

Kertoo nesteen alkunopeuden z-suunnassa.

Sallitut arvot: reaalityluku väliltä 0-0.1

Vapaan nestekerroksen paksuus, h_v

Koska algoritmissa näytettä käsitellään periodisesti, z-suunnassa näytteen ylä- ja alapuolella pitää olla vapaa nestekerros jotta neste virtaisi kunnolla. Tämä parametri kertoo kuinka monta laskentayksikköä nestettä on sekä näytteen ylä- että alapuolella.

Sallitut arvot: kokonaisluku väliltä 0 - 50

Simulointimoodi

Tämä kertoo, mitä kolmesta mahdollisesta moodista käytetään. Moodit ovat Navier-Stokes, lineaarinen muunneltavan τ -parametrin kera sekä lineaarinen moodi jossa τ on aina 1.0.

Sallitut arvot: 0, 1 tai 2.

Vaikutukset muihin parametreihin: Valittaessa moodi 2 τ on aina 1.0.

Pituuskaala, D

Tällä parametrilla kerrotaan mallin koko suhteessa todelliseen maailmaan. Useimmiten parametri tarkoittaa mallissa olevien huokosten tai kiinteän aineen lankojen keskimääräistä halkaisijaa laskentayksiköissä.

Sallitut arvot: kokonaisluku väliltä 1-100

Lopetusehto

Simulaatio voidaan lopettaa jo ennen asetetun aika-askelmäärän täyttymistä jos nesteen permeabiliteetti vaihtelee tämän parametrin määrittelymän vaihteluvälin puitteissa. Vaihteluväli määräytyy kahden peräkkäisen mittaustuloksen suhteena. Parametri on siis suhdeluku eli välillä 0-1, mutta käytännössä hyvin pieni.

Sallitut arvot: 0-0.1

4.1.4 Tulostiedosto

Tulostiedosto sisältää simulaatio-ohjelmalle annetut parametrit sekä joitakin simulaation aikana laskettuja suureita. Tiedostonimi on muotoa *datafile.dat*, missä *datafile* on simulaatio-ohjelmalle annettu tulostiedoston nimi ilman tarkenninta. Tiedosto sisältää rivejä, joissa on selkokielisenä rivin sisältämä tieto sekä lukuja. Seuraavassa on kerrottu jokaisen rivin sisältö. Luvut on merkitty sulkeisiin kertoen, mitä luvut ovat sekä niiden tyytit.

Hilan koko ($L_x L_y L_z + 2h_v$: kolme kokonaislukua), z-koossa on mukana vapaa nestekerros

Näytteen korkeus (L_z : kokonaisluku), ilman vapaata nestekerrosta

Huokoisuus (reaaliluku)

Alkunopeus (v_0 : reaaliluku)

Tilavuusvoima (reaaliluku) jolla nestettä liikutetaan

Relaksaatioparametri (τ : reaaliluku), **viskositeetti**: (reaaliluku)

Suoritettavat laskentakierrokset (t: kokonaisluku)

Suoritettavat laskentakierrokset (kokonaisluku)

Simulointimoodi: ("Navier Stokes / Linear, variable tau / Linear, tau=1.0")
Suoritus aika sekunteina (reaaliluku)
Syötetiedosto (tiedostonimi.sample)
Tulostiedoston perusnimi: (tiedostonimi, ei tarkenninta)
Nesteen momentti näytteessä, mukana vapaa nestekerros (kolme reaalilukua)
Nesteen liikemäärä näytteessä, mukana vapaa nestekerros (reaaliluku)
Nesteen liikemäärä näytteessä (kolme reaalilukua)
Nesteen massa näytteessä (reaaliluku)
Keskimääräinen nopeus näytteessä, mukana vapaa nestekerros (kolme reaalilukua)
Keskimääräinen nopeus näytteessä (kolme reaalilukua)
Keskimääräinen nopeus vapaassa nestekerroksessa (kolme reaalilukua)
Näytteen Reynoldsin luku (reaaliluku)
Näytteen ulkopuolen Reynoldsin luku (reaaliluku)
Permeabiliteetti (reaaliluku)
Painehäviökerroin (reaaliluku)
Paine-ero näytteessä (reaaliluku)

Seuraavassa esimerkkiedosto.

```

Lattice dimensions: 100 50 40
Sample height: 20
Porosity: 0.40172
Initial v_z: 0
Local body force: 0.0001
Relaxation parameter: 0.668, viscosity: 0.056
Number of requested time iterations: 60
Number of executed time iterations: 60
Running mode : Linear, tau = 1.0.
Time spent in the kernel: 10
Input file name: ./naytteen/3d0456.sample
Output file, base name: datafile
Total fluid momentum in sample plus fluid layer: -1.57308 \
  3.27486 80.4813
Total fluid mass in sample plus fluid layer: 140178
Total fluid momentum in sample: -1.16458 3.11712 60.7174
Total fluid mass in the sample: 40112.4
Average velocity in sample plus fluid layer: -1.1222e-05 \
  2.33621e-05 0.000574136
  
```

```

Average velocity in sample: -2.90329e-05 7.77096e-05 \
    0.00151368
Average velocity in fluid layer: -4.08236e-06 1.57641e-06 \
    0.000197509
Sample Reynolds number for diameter 10: 0.2703
Outside Reynolds number for diameter 10: 0.0352695
Permeability: 0.329258
Pressure loss coefficient K_0: 3616.47
Pressure difference over the screen: 0.004

```

4.1.5 Aikakehitystiedosto

Aikakehitystiedosto on ascii-muotoinen tiedosto, johon simulointiohjelma kirjoittaa aika-ajoin tiettyjä simulaation tilaa kuvaavia tunnuslukuja. Tiedoston nimi on muotoa *datafile.evol*, missä *datafile* on simulointiohjelmalle annettu tulostiedoston nimi ilman tarkenninta. Tiedostossa on keskenään samanmuotoisia rivejä, joilla on yksitoista lukua välilyönnein eroteltuina. Luvut ovat

Aika-askel, eli monesko laskentakierros on menossa (kokonaisluku)
Nesteen massa vapaa nestekerros mukaanluettuna (reaaliluku)
Nesteen momentti vapaa nestekerros mukaanluettuna (kolme reaalilukua)
Nesteen massa ilman vapaata nestekerrosta (reaaliluku)
Nesteen momentti ilman vapaata nestekerrosta (kolme reaalilukua)
Permeabiliteetti (reaaliluku)
K_0, eli painehäviökerroin (reaaliluku)

Seuraavassa esimerkkitiedosto.

```

0 144882 140172 0 0 0 40172 0 0 0 -0.0112482
1 144882 140172 0 0 14.0164 40172 0 0 4.01697 0.0112469
2 144882 140172 -0.00642383 0.00269927 27.2204 40172 \
    -0.0064238 0.00269922 7.53337 0.0309387
3 144882 140172 -0.025807 0.0127797 39.1364 40171.8 \
    -0.0259229 0.0128542 10.2409 0.0461013
4 144882 140172 -0.0535078 0.0324042 50.4993 40171.5 \
    -0.0537986 0.0325808 12.671 0.0597102

```

4.1.6 Kenttätiedosto

Kenttätiedosto on simuloinnin varsinainen tulostiedosto. Se sisältää simuloitua nopeus- ja painekenttää binäärimuotoisena. Tiedostonimi on muotoa *datafile.field*, missä *datafile* on simulointiohjelmalle annettu tulostiedoston nimi ilman tarkenninta. Tiedosto on binäärimuotoinen. Se sisältää jokaista hilapistettä kohti neljä reaalilukua, jotka ovat nopeusvektorin x-, y- ja z-komponentit sekä nesteen paine. Pisteiden tiedot ovat koordinaattien mukaisessa järjestyksessä siten, että nopeimmin juoksee z-koordinaatti, sitten y-koordinaatti ja hitaimmin x-koordinaatti.

4.1.7 Näytetiedosto

Näytetiedosto on näytteenluontiohjelman luoma tiedosto, joka sisältää tiedon näytteen geometriasta, eli tiedon siitä, mitkä pisteet kuuluvat nesteseen ja mitkä kiintoaineeseen. Tiedostonimi on muotoa *samplefile.sample*, missä *samplefile* on näytteenluontiohjelmalle annettu tiedostonimi ilman tarkenninta. Näytetiedosto sisältää ascii-muotoisia kokonaislukuja ja siinä on seuraavanlaiset rivit:

Koko x-, y- ja z-suunnassa (kolme lukua) sekä samalla rivillä

Neljä turhaa lukua jotka ovat mukana historiallisista syistä

Tiedoston rivien määrä - 3

Ensimmäisen hilapisteen tyyppi (1 = kiintoaine, 0 = neste)

Näiden kolmen rivin jälkeen tiedostossa on 2. rivin ilmoittama määrä rivejä, joilla kerrotaan aina montako samanlaista peräkkäistä pistettä mallissa on. Pisteiden tyyppi vaihtuu aina kerrotun määrän jälkeen ja ensimmäisen pisteen tyyppi kerrotaan 3. rivillä. Koordinaatit juoksevat nopeimmin z-suunnassa, sitten y-suunnassa ja hitaimmin x-suunnassa.

Seuraavassa ote esimerkkiedoston alusta.

```
100 50 20 13 13 13 13
10489
1
60
2
12
5
```

4.1.8 Projektitiedosto

Projekti sisältää tiedon tietystä simulointi- ja visualisointitapahtumasta. Tiedostoon tallennetaan tieto käytetyistä näyte- ja tulostiedostoista ja ohjelman asetuksista. Projektitiedoston muoto jätetään tässä vaiheessa vielä avoimeksi, mutta se kiinnitetään sovelluksen toteutuksen alkuvaiheessa.

4.2 Järjestelmävaatimukset

Tässä luvussa selostetaan tarkasti sovellukselle asetetut vaatimukset. Vaatimukset on ryhmitelty ohjelman ominaisuuksien mukaan ja numeroitu juoksevasti. Ominaisuuksista kerrotaan tarkemmin käyttötapauksissa.

4.2.1 Ohjelman käynnistäminen ja sulkeminen

Vaatus 1. Projektitiedoston luominen ja avaaminen Ohjelmaa käynnistettäessä pitää ladata aina projektitiedosto. Jos aiemmalla käyttökerralla jäi jokin simulointi kesken, pitää avata kyseisen simulaation projektitiedosto. Käyttäjälle pitää kertoa onko aloitettu simulointi valmis, keskeytetty tai vielä käynnissä ja näyttää sitä koskevat tiedot. Muuten pitää luoda tyhjä projekti oletusasetuksin.

Vaatus 2. Käynnistetyt prosessit ohjelman sulkeutuessa Kun ohjelma suljetaan, mahdollisesti käynnissä olevien näytteenluonti- tai simulointiprosessien täytyy jäädä käyntiin ohjelman sulkeuduttua. Käyttäjältä pitää kuitenkin kysyä, haluaako hän lopettaa prosessit vai jättää ne käyntiin.

4.2.2 Asetusten muuttaminen

Vaatus 3. Projektin asetukset Käyttäjän pitää voida muuttaa sovelluksen asetuksia. Asetukset tallennetaan nykyiseen projektitiedostoon. Hallittaessa nykyisen projektin asetukset voi tallentaa oletuksiksi, jolloin ne tulevat luotavien uusien projektien asetuksiksi. Muutettavat asetukset jätetään vielä tässä vaiheessa avoimiksi.

4.2.3 Näytteen luominen

Simulaatio tarvitsee geometrisen mallin jonka pohjalta nesteen virtausta lasketaan. Käyttäjän pitää voida käynnistää laskentamallin luova ohjelma käyttöliittymästä.

Vaatus 4. Näytteenluontiohjelman käynnistäminen Käyttöliittymästä pitää voida käynnistää erillinen ohjelma, joka luo näytteen.

Vaatus 5. Näytteenluonnin parametrien asettaminen Näytteenluontiohjelman käynnistämisen yhteydessä käyttäjän pitää voida asettaa kyseisen ohjelman tarvitsemat parametrit. Annettavat parametrit on selostettu luvussa 4.1.2.

Vaatus 6. Parametrien oikeellisuustarkistus Ohjelmalle annettavien parametrien arvot pitää tarkastaa ja varmistaa, että annetut arvot ovat järkevät. Parametrit ja niiden sallitut arvoalueet on selostettu luvussa 4.1.2.

4.2.4 Näytteen lataaminen

Vaatus 7. Valmiin näytteen lataaminen Käyttäjän pitää voida ladata näyte olemassa olevasta näytetiedostosta.

4.2.5 Näytteen tutkiminen

Käyttäjä haluaa varmistua näytteen järkevyydestä. Näytettä pitää voida tarkastella kolmiulotteisena kuvana.

Vaatus 8. Kolmiulotteisen kuvan piirtäminen Käytössä olevasta näytteestä pitää piirtää kolmiulotteinen kuva.

Vaatus 9. Kolmiulotteisen kuvan käsittely Piirrettyä kuvaa pitää voida pyöritellä ja siirrellä.

Vaatus 10. Näytteen leikkaaminen Näytettä pitää voida tarkastella kokonaisena, ja siitä pitää voida leikata pienempi osa tarkasteltavaksi.

4.2.6 Simulointi

Visualisoitavat laskentatulokset tuotetaan simulointiohjelmalla, joka pitää voida käynnistää käyttöliittymästä.

Vaatus 11. Simulointiohjelman käynnistäminen Käyttöliittymästä pitää voida käynnistää erillinen ohjelma, joka suorittaa virtaussimulaation laskennan ja tallentaa tuloksen erityisiin tiedostoihin.

Vaatus 12. Simuloinnin parametrien asettaminen Simulointiohjelman käynnistämisen yhteydessä käyttäjän pitää voida asettaa kyseisen ohjelman tarvitsemat parametrit. Annettavat parametrit on selostettu luvussa 4.1.3.

Vaatus 13. Parametrien oikeellisuustarkistus Ohjelmalle annettavien parametrien arvot pitää tarkastaa ja varmistaa, että annetut arvot ovat järkevät. Parametrit ja niiden sallitut arvoalueet on selostettu luvussa 4.1.3.

Vaatus 14. Etäajo Simulointiohjelma pitää voida käynnistää etäajona fysiikan laitoksen rinnakkaiskoneella. Tätä ominaisuutta ei toteuteta tämän projektin puitteissa, mutta kirjataan jatkokehitystä ajatellen.

4.2.7 Simuloinnin keskeyttäminen

Vaatus 15. Käynnissä olevan simuloinnin keskeyttäminen Käyttäjän pitää voida keskeyttää käynnissä oleva simulointi hallitusti ennen lopetusehtojen täyttymistä. Välitulokset pitää saada talteen.

4.2.8 Simulointituloksen lataaminen

Valmistunutta simulointia pitää voida tarkastella myöhemmin, ja keskeytetty simulaatio pitää voida ladata jatkettavaksi.

Vaatus 16. Aiempien tulosten tarkasteleminen Aiemmin luotu tai muualta tuotu simulointitulos pitää voida ladata tulosten tarkastelua varten.

Vaatus 17. Keskeytyneen simuloinnin lataaminen Aiemmillä käyttökerroilla keskeytetty simulointi pitää voida ladata siihen tilanteeseen mihin jäätin.

4.2.9 Simuloinnin jatkaminen

Simulointien laskentaa pitää voida jatkaa.

Vaatimus 18. Keskeytyneen simuloinnin jatkaminen Aiemmillä käyttökerroilla keskeytettyä ja uudelleen ladattua simulointia pitää voida jatkaa siitä kohdasta, mihin jäätiin.

Vaatimus 19. Valmistuneen simuloinnin jatkaminen Myös valmista simulointia pitää voida jatkaa tarvittaessa vaikka lopetusehdot olisivat jo täyttyneet.

Vaatimus 20. Jatkettavan simuloinnin parametrit Valmistuneen tai keskeytyneen simuloinnin parametrit on tallennettu tulostiedostoon, jonka rajapinta on kuvattu luvussa 4.1.4. Nämä parametrit luetaan uudelleenkäynnistettävän simulaation parametreiksi. Käyttäjä voi muuttaa parametrien arvoja halutessaan.

4.2.10 Simulointituloksen tutkiminen

Sovelluksen keskeisin osa on simulointitulosten tutkiminen. Käyttäjän pitää voida luoda erilaisia havainnollistavia kuvia tuloksista.

Vaatimus 21. Näkymän leikkaaminen Käyttäjän pitää voida tarkastella näkymää kokonaisuutena ja hänen pitää voida leikata siitä pienempi osa tarkasteltavaksi.

Vaatimus 22. 2d-leikkeiden muodostus Käyttäjän pitää voida muodostaa näytteestä kaksiulotteinen leike. Leikkaaminen tapahtuu koordinaattiakseleiden suuntaan, ja käyttäjän pitää voida valita suunta, johon leikataan, ja leikkauksen kohta.

Vaatimus 23. 2d-leikkeiden värikoodaus Käyttäjän pitää voida havainnollistaa laskentatuloksia muodostetulla leikepinnalla värikoodaamalla pinta halutun suureen mukaisesti. Valittavia suureita ovat

- Nesteen paine
- Nopeus x- y- tai z-suuntaan
- Vauhti, eli nopeusvektorin pituus

Vaatus 24. 2d-leikkeiden väriskaalan valinta Käyttäjän pitää voida säätää 2d-leikkeen värikoodauksen väriskaala. Käyttäjä valitsee kaksi väriä sekä niiden välisen skaalauksen joko lineaarisena tai logaritmisena. Tämän perusteella sovellus muodostaa väriskaalan.

Vaatus 25. Leikepinnalta laskettavat suureet Käyttäjän pitää voida tarkastella visualisoitavan suureen keskiarvoa ja huokoisuutta leikepinnalla.

Vaatus 26. Virtaviivojen piirtäminen Malliin pitää käyttäjän pyynnöstä piirtää virtaviivoja, jotka kuvaavat kuvitteellisten nestepartikkelien liikerataa. Käyttäjän pitää voida valita taso, jolta alkaen viivat piirretään, sekä viivojen määrä.

Vaatus 27. Virtaviivojen värikoodaus Virtaviivat pitää värikoodata käyttäjän toiveiden mukaan joko siten, että joka viivalla on oma väri tai siten, että jokainen viivasegmentti värikoodataan halutun suureen mukaan.

Vaatus 28. Virtaviivojen hukkuminen Normaalissa näytteessä korkeintaan 50% lähetetyistä virtaviivoista saa hukkaa matkalla seinämiin törmäyksien seurauksena.

Vaatus 29. Partikkelianimaatiot Käyttäjän pitää voida luoda animaatioita, joissa tietyltä kohdalta lasketaan liikkeelle nestepartikkeleita, jotka seuraavat nesteen kulkua näytteessä simulointitulosten mukaisesti. Käyttäjän pitää voida valita partikkelien määrä sekä taso, jolta partikkelit lasketaan liikkeelle.

4.2.11 Kuvien tallentaminen

Käyttäjän pitää voida tallentaa kuvia visualisoinneista.

Vaatus 30. Still-kuvien kaappaus Käyttäjän pitää voida milloin tahansa kaapata kuva nykyisestä visualisointinäkymästä. Kuville voidaan asettaa perusnimi, johon lisätään juokseva numerointi. Oletuksena käytetään projektin nimeä.

Vaatus 31. Kuvien automaattinen tallennus Partikkelianimaatioiden yhteydessä käyttäjän pitää voida pyytää ohjelmaa tallentamaan kuvia automaattisesti tietyin väliajoin. Kuvat tallennetaan juoksevilla numeroinnilla.

Vaatus 32. Videokuvan tallennus Tallennettu kuvasarja pitää voida muuttaa avi-videoksi. Tämä ominaisuus on valinnainen, ja toteutetaan, jos aikaa on.

4.2.12 Lokitietojen tarkastelu

Simulointiohjelma tallentaa väliaikatietoja lokitiedostoon. Käyttäjän pitää voida tarkastella näitä väliaikatietoja.

Vaatus 33. Lokitiedoston lukeminen Simulointiohjelma kirjoittaa tietyn väliajoin yhden rivin tiedostoon, jonka tarkennin on .evol. Sovelluksen täytyy käydä lukemassa viimeinen rivi tiedostosta aika ajoin ja tulkita sen sisältö. Tiedoston sisältö on kuvattu luvussa 4.1.5.

Vaatus 34. Lokitietojen esittäminen Käyttäjän pitää voida tarkastella ainakin permeabiliteetti-parametria, joka luetaan lokitiedostosta.

4.3 Tehokkuusvaatimukset

Sovellukselle ei aseteta kovin tiukkoja tehokkuusvaatimuksia. Käyttöliittymän vasteaikojen täytyy kuitenkin pysyä kohtuullisina.

4.4 Suunnittelurajoitteet

Sovelluksen vaatimukseen liittyy tekijöitä, jotka rajoittavat sovelluksen suunnittelijoiden mahdollisuuksia. Tässä luvussa esitellään nämä rajoitteet.

Sovelluksen täytyy toimia sekä Windows- että Linux-käyttöjärjestelmässä. Tämä on otettava huomioon sovellusta suunniteltaessa ja ohjelmoitaessa; valittujen ratkaisujen täytyy toimia kummassakin käyttöjärjestelmässä. Toimivuus täytyy varmistaa ohjelmoimalla samaan aikaan molemmissa käyttöjärjestelmissä ja testaamalla koko ajan kaikkea koodia.

Visualisoinnissa käytetään apuna OpenGL-grafiikkakirjastoa. Tämä ei juurikaan rajoita suunnittelua, sillä OpenGL saadaan kytkettyä saumattomasti muuhun käyttöliittymään. Asia vaatii kuitenkin paneutumista ja opiskelua toteuttajien taholta.

Käyttöliittymän suunnittelussa täytyy noudattaa yleisiä käytäntöjä helpokäyttöisyyden lisäämiseksi. Tämä on otettava huomioon valikoiden ja ikkunoinnin ulkoasua ja nimeämistä suunnitellessa.

5 Yhteenveto

Tässä dokumentissa on kuvattu Hibbo-sovelluksen käyttäjät ja käyttötapaukset, yhteydet muihin ohjelmiin, käytetyt tiedostot sekä järjestelmälle asetetut tarkat vaatimukset. Seuraavassa listassa esitetään vielä Hibbo-sovelluksen vaatimukset tiivistetysti. Numerointi viittaa vaatimusten numerointiin luvussa 4.2.

1. Projektitiedoston luominen ja avaaminen
2. Käynnistetyt prosessit ohjelman sulkeutuessa
3. Projektin asetukset
4. Näyteenluontiohjelman käynnistäminen
5. Näytteenluonnin parametrien asettaminen
6. Parametrien oikeellisuustarkistus
7. Valmiin näytteen lataaminen
8. Kolmiulotteisen kuvan piirtäminen
9. Kolmiulotteisen kuvan käsittely
10. Näytteen leikkaaminen
11. Simulointiohjelman käynnistäminen
12. Simuloinnin parametrien asettaminen
13. Parametrien oikeellisuustarkistus
14. Etäajo
15. Käynnissä olevan simulaation keskeyttäminen
16. Aiempien tulosten tarkasteleminen
17. Keskeytyneen simuloinnin lataaminen
18. Keskeytyneen simuloinnin jatkaminen
19. Valmistuneen simuloinnin jatkaminen
20. Jatkettavan simuloinnin parametrit

21. Näkymän leikkaaminen
22. 2d-leikkeiden muodostus
23. 2d-leikkeiden värikoodaus
24. 2d-leikkeiden väriskaalan valinta
25. Leikepinnalta laskettavat suureet
26. Virtaviivojen piirtäminen
27. Virtaviivojen värikoodaus
28. Virtaviivojen hukkuminen
29. Partikkelianimaatiot
30. Still-kuvien kaappaus
31. Kuvien automaattinen tallennus
32. Videokuvan tallennus
33. Lokitiedoston lukeminen
34. Lokitietojen esittäminen