

Dogma

Vaatimusdokumentti

ohtu06-dogma-list@cs.helsinki.fi

Helsingin yliopisto, Tietojenkäsittelytieteen laitos

Ohjelmistotuotantoprojekti

10/10/2006

Kurssi

581260 Ohjelmistotuotantoprojekti (9 op, 6 ov)

Projektiryhmä

Bjorkgren Joakim
Kuronen Ville
Ristola Toni
Tani Antti
Vihavainen Arto

Asiakas

Niinivaara Olli

Johtoryhmä

Taina Juha
Moilanen Ilari

Kotisivu

<http://cs.helsinki.fi/group/dogma/>

Versiohistoria

<i>Versio</i>	<i>Päiväys</i>	<i>Muutokset</i>
0.1	13.09.2006	Ensimmäinen versio
0.2	20.09.2006	Lisää sisältöä
0.3	26.09.2006	Prioriteetit
0.4	27.09.2006	Korjattu versio
0.41	28.09.2006	Lisää muutoksia
0.5	29.09.2006	Korjauksia, lisää vaatimuksia
0.51	01.10.2006	Korjauksia
0.52	03.10.2006	Korjauksia
0.75	07.10.2006	Korjauksia
1.0	10.10.2006	Dokumentti jäädytetty

Sisältö

1.Johdanto.....	1
2.Sanasto.....	1
3.Sidosryhmät.....	1
4.Käyttötapausesimerkki.....	2
5.Toiminnalliset vaatimukset.....	2
5.1 Verkkonäkymä.....	2
5.1.1 Verkon laajennus.....	2
5.1.2 Resurssien lisääminen verkkonäkymään.....	3
5.1.3 Resurssin poisto verkkonäkymästä.....	3
5.1.4 Verkkonäkymän tallennus kuvaksi tai tiedostoon.....	3
5.2 Resurssitaulukko.....	3
5.2.1 Resurssitaulukon kentät.....	3
5.2.2 Resurssitaulukkoon hakeminen tietokannasta.....	3
5.2.3 Resurssitaulukon sisäinen etsintä.....	3
5.2.4 Resurssitaulukon järjestäminen attribuuttien mukaisesti.....	4
5.2.5 Resurssitaulukon sarakkeiden järjestäminen.....	4
5.2.6 Viimeisimmän haun toistaminen.....	4
5.2.7 Lomakenäkymän avaaminen resurssitaulukosta.....	4
5.2.8 Resurssien valitseminen resurssitaulukosta.....	4
5.2.9 Uuden osajoukon muodostaminen resurssitaulukosta.....	4
5.2.10 Resurssin yhteyksien avaaminen uutena osajoukkona.....	5
5.2.11 Resurssitaulukon asettelun tallentaminen.....	5
5.3 Resurssin tietojen muokkaus.....	5
5.3.1 Resurssin tietojen muokkaus resurssitaulukossa.....	5
5.3.2 Monen resurssin yhtäaikainen muokkaus resurssitaulukossa.....	5
5.3.3 Resurssin tietojen muokkaus verkkonäkymässä.....	5
5.3.4 Resurssin tietojen muokkaus lomakenäkymässä.....	5
5.3.5 Resurssin lisääminen.....	6
5.3.6 Resurssien yhdistäminen.....	6
5.3.7 Resurssien puolittaminen.....	6
5.3.8 Resurssin poisto.....	6
5.3.9 Resurssin attribuuttien automaattinen päivitys.....	6
5.3.10 Resurssien välisten viitteiden eheyden tarkastus.....	6
5.4 URI – taulukko.....	6
5.4.1 URI:n lisäys.....	6
5.4.2 URI:n poisto.....	7
5.4.3 URI:n muokkaus.....	7
5.4.4 URI:n avaaminen.....	7
5.5 Vapaamuotoinen SQL-kysely.....	7
5.6 Verkkohakukoneet.....	7
5.7 Vapaasanahaku dokumenttien sisällön perusteella.....	7
5.8 Tapahtumien seuranta ja tallennus.....	7
5.9 Resurssiselaimen statuspaneeli.....	8
5.10 Resurssileikepöytä.....	8
5.10.1 Kopiointi.....	8
5.10.2 Leikkaaminen.....	8
5.10.3 Liittäminen.....	8

5.10.4	Lisääminen.....	8
5.10.5	Vaihtaminen.....	8
5.10.6	Joukko-opin mukainen leikkaus.....	9
5.10.7	Suodattaminen.....	9
5.11	Tietokannan valinta.....	9
5.12	Resurssin metadatan automaattinen luonti metadatan sisältöä jäsentämällä.....	9
5.13	Pikanäppäinten toteutus.....	9
5.14	Tekstikenttä hakusanoille.....	9
5.15	Testausviitteiden luonti.....	9
6.	Ei toiminnalliset vaatimukset.....	11
6.1	HSQLDB rajapinta.....	11
6.2	GUESS rajapinta.....	11
6.3	Hakutuloksien järjestämisen ja vierittämisen nopeus.....	11
6.4	Connections, fromX, toX avaamisen nopeus.....	11
6.5	Hakujen nopeus.....	11
6.6	Toteutuskieli englanti.....	11
7.	Toimintaympäristövaatimukset.....	12
7.1	Java.....	12
7.2	Eclipse SWT.....	12
7.3	HSQLDB.....	12
7.4	GUESS.....	12
7.5	Muu toimintaympäristö.....	12
7.6	Vaatimusten prioriteetit.....	12
7.6.1	Toteutetaan.....	12
7.6.2	Pyritään toteuttamaan.....	13
7.6.3	Toteutetaan, jos jää aikaa.....	13
7.6.4	Ei toteuteta.....	14
8.	Resurssien attribuutit ja mahdolliset arvot.....	15
8.1	URI-taulukon attribuutit.....	15
9.	Järjestelmäarkkitehtuuri.....	16
10.	Järjestelmän elinkaari.....	17
11.	Liitteet.....	18
11.1	Järjestelmään liittyvät komponentit.....	18
11.1.1	GUESS – the Graph Exploration System.....	18
11.1.2	Eclipse SWT Table.....	18
11.1.3	HSQL database engine.....	18

1. Johdanto

Dogma on Helsingin yliopiston tietojenkäsittelytieteen laitoksen ohjelmistotuotantoprojektin ryhmä. Kurssilla opiskelijoista koostuva ryhmä tekee asiakkaalle vaatimusmäärittelyn mukaisen ohjelmiston. Tässä projektissa asiakas haluaa itselleen metatadataperustaisen resurssiselaimen. Resurssiselainta tullaan käyttämään lähinnä tutkintamielessä dokumenttien metadatan hallintaan, sekä niiden yhteyksien väliseen visualisointiin.

Resurssiselain soveltuu yleiseen resurssien hallintaan. Tiedostojen ja hakemistojen sijaan kohteena ovat kuitenkin todellisen maailman olioita vastaavat resurssit. Resurssit voivat olla esimerkiksi henkilöitä ja dokumentteja, mutta sovellus ei aseta rajoitteita resurssien tyypeille.

Resurssiselaimen keskeisiin palveluihin kuuluu monipuolinen resurssien haku ja järjestäminen, resurssien osajoukkojen valinta ja poiminta sekä resurssien ja niiden yhteyksien visualisointi. Apuna verkkojen visualisoinnissa resurssiselain käyttää GUESS-ohjelmaa. Resurssiselain kerää myös tapahtumalokia käyttäjän toiminnasta, joten sovellusta voidaan jatkokehittää lisäämällä personointi- ja adaptaatio-ominaisuuksia.

2. Sanasto

Attribuutti	Ominaisuusarvo.
GDF	GUESS:n käyttämä oma verkkojen tallennusformaatti.
GraphML	Verkkojen tallennusformaatti. http://graphml.graphdrawing.org/
GUESS	Ks. kohta 11.1.1
HSQLDB	Ks. kohta 11.1.3
Lomakenäkymä	Näkymä resurssin attribuuttien muokkaamiseen, sekä resurssien poistamiseen ja lisäämiseen.
Resurssi	Metadasta tunnistettu yksilöllinen olio kuten henkilö, dokumentti tai julkaisukanava.
Resurssitaulukko	Ohjelmassa käytetty taulukko, johon on listattu käyttäjän hakuehdoilla löydetty tulosjoukko.
URI	Uniform Resource Identifier. Resurssin sijainnin yksilöllinen tunniste.
Verkkonäkymä	Resursseista muodostettu verkko GUESS:in esittämänä.

3. Sidosryhmät

Asiakas	Olli Niinivaara
Ryhmän ohjaaja	Ilari Moilanen
Kurssin vastuhenkilö	Juha Taina

4. Käyttötapausesimerkki

Teppo kirjoittaa artikkelia aiheesta "Katsaus PageRank- tyyppisiin solmujen keskeisyysmittoihin". Hän avaa sovelluksen ja tekee vapaatekstihaun "graph centrality pagerank" dokumenttityyppisille resursseille. Tulosjoukko sisältää oletusarvoisesti enintään 1000 resurssia ja on järjestetty relevanssin mukaan. Teppo huomaa tulosjoukon alkupäätä selaillessaan muutaman potentiaalisesti mielenkiintoisen artikkelin ja leikkaa ja liittää ne toiseen ikkunaan/tabiin. Sitten Teppo järjestää tulosjoukon siten, että alkuun tulevat hänelle tutuimmat artikkelit ja leikkaa myös osan näistä lähempää tarkastelua varten. Lopuksi Teppo järjestää dokumentit vielä niin, että alussa ovat artikkelit, joita hän ei ole aikaisemmin lukenut ja poimii vielä näistä osan. Sitten Teppo siirtyy lukemaan poimittuja artikkeleita järjestäen ne niiden tuottoajankohdan mukaiseen järjestykseen.

Teppo visualisoi dokumentteihin viittaavat artikkelit verkkoon ja löytää näin muutamia uusia mielenkiintoisia dokumentteja. Eräs dokumentti sisältää niin hätkähdyttäviä tuloksia, että hän hakee listan dokumentin tekijöistä ja siirtyy hetkeksi heidän kotisivuilleen katselemaan, mitä porukkaa on kyseessä.

Teppo huomaakin erään ennestään tuntemattoman tutkijan, jonka ajatukset näyttävät lupaavilta, ja nostaa tämän toimijan arvoluokitusta parilla pykälällä. Dokumentteja lukiessa vastaan tulee myös dokumentti, jonka sijainnista Tepolla ei ole tietoa. Teppo pyytää sovellusta avaamaan muutaman hakukoneen tätä dokumenttia koskevalla kyselyllä ja löytääkin näin pdf-dokumentin, jonka hän tallentaa sovelluksen pdf-lähdehakemistoon.

Istunnon lopuksi Teppo merkkää kerralla kaikkiin poimittuihin resursseihin tagin "jatka tästä" ja sulkee sovelluksen. Seuraavana päivänä hän jatkaa työtään hakemalla kaikki tagin "jatka tästä" sisältävät resurssit (ja ehkä poistaa tämän tagin). Lisäksi Teppo tekee uudestaan eilisen vapaatekstihaun tällä kertaa lisäehdolla, että imported = tänään. Näin hän saa listan uusista resursseista, jotka sovellus on yön aikana hakenut tietolähteistä ja indeksoinut hänen käyttöönsä. Artikkelin kirjoittamisprosessi jatkuu jouhevasti...

5. Toiminnalliset vaatimukset

5.1 Verkkonäkymä

Resurssiselaimesta valitaan graph, jolloin ohjelma luo käyttäjän valitsemasta osajoukosta verkkonäkymän. Verkossa näkyy aluksi osajoukon resurssit ja niiden väliset yhteydet, sekä jokaisen resurssin muun kuin osajoukossa jo olevan, yhden kaaren päässä sijaitsevat resurssit.

Prioriteetti: toteutetaan

5.1.1 Verkon laajennus

Verkkoa voi laajentaa klikkaamalla solmua, tällöin verkko laajenee yhden kaaren matkan kyseisestä solmusta.

Prioriteetti: pyritään toteuttamaan

5.1.2 Resurssien lisääminen verkkonäkymään

Käyttäjä pystyy lisäämään uuden osajoukon tai yksittäisen resurssin jo esillä olevaan verkkonäkymään.

Prioriteetti: pyritään toteuttamaan

5.1.3 Resurssin poisto verkkonäkymästä

Käyttäjä voi poistaa resursseja verkkonäkymästä. Poisto tapahtuu ainoastaan verkkonäkymässä, ei itse tietokannassa.

Prioriteetti: pyritään toteuttamaan

5.1.4 Verkkonäkymän tallennus kuvaksi tai tiedostoon

Käyttäjä voi tallentaa verkkonäkymän kuvaksi tai tiedostoon. Tiedostoon tallennettaessa käytetään GDF- ja GraphML tallennusformaatteja.

Prioriteetti: toteutetaan, jos aikaa jää

5.2 Resurssitaulukko

Käyttöliittymässä on resurssitaulukko, jonka avulla käyttäjä voi tutkia resursseja.

Prioriteetti: toteutetaan

5.2.1 Resurssitaulukon kentät

Resurssitaulukon yksi rivi kuvaa yhtä resurssia. Rivillä on sarakkeita, jotka kuvaavat resurssin attribuutteja.

Prioriteetti: toteutetaan

5.2.2 Resurssitaulukon hakeminen tietokannasta

Resurssiselaimessa on kentät tietokantakyselyä varten. Jokaiselle attribuutille on oma kenttensä hakuehtoja varten ja yksinkertaisia hakuetoja pystytään automaattisesti jäsentämään (vertailumerkit < > >= <= !). Haku tehdään niillä järjestysehdolla, jotka ovat listalla voimassa. Jos mitään ei ole voimassa, ei SQLI-kyselyynkään laiteta järjestysehtoja. Hakuetoja voi asettaa useita ja haun tuloksia esitetään korkeintaan sen verran, mitä käyttäjä on määritellyt.

Prioriteetti: toteutetaan

5.2.3 Resurssitaulukon sisäinen etsintä

Resurssiselaimessa on käytössä hakukentät jokaista metadatan attribuuttia varten. Resurssihaun voi suorittaa yhdellä tai useammalla haku ehdolla.

Prioriteetti: pyritään toteuttamaan

5.2.4 Resurssitaulukon järjestäminen attribuuttien mukaisesti

Resurssitaulukon jokaisen sarakkeen voi järjestää attribuuttien arvojen mukaan aakkosjärjestykseen tai numerojärjestykseen (laskeva/nouseva).

Attribuutteja voi myös priorisoida valitsemalla useamman kuin yhden sarakkeen ja päättämällä näiden järjestysperiaatteet.

Prioriteetti: toteutetaan

5.2.5 Resurssitaulukon sarakkeiden järjestäminen

Käyttäjällä on mahdollisuus järjestää resurssitaulukon sarakkeet haluaamansa järjestykseen.

Resurssitaulukon sarakkeita voi lukita paikoilleen, jolloin taulukkoa leveyssuuntaisesti vierittäessä lukitut sarakkeet pysyvät koko ajan näkyvissä.

Käyttäjä voi säätää sarakkeiden leveyttä, sekä valita mitkä sarakkeet ovat näkyvissä taulukossa, ja mitkä eivät.

Prioriteetti: pyritään toteuttamaan

5.2.6 Viimeisimmän haun toistaminen

Kenttäehdot ovat valittavissa viimeisimpien hakujen alavetovalikosta, jolloin ne voidaan toistaa nopeasti.

Prioriteetti: toteutetaan, jos jää aikaa

5.2.7 Lomakenäkymän avaaminen resurssitaulukosta

Käyttäjä pystyy avaamaan yksittäistä resurssia kaksoisklikkaamalla kyseisen resurssin lomakenäkymään. Avaus onnistuu myös Enter:iä painamalla rivin kohdistuksen ollessa resurssitaulukon rivillä.

Prioriteetti: toteutetaan

5.2.8 Resurssien valitseminen resurssitaulukosta

Resurssitaulukosta voi valita resursseja ctrl-nappulaa pohjassa painamalla ja klikkaamalla hiirellä. Tietyllä hetkellä valitut resurssit erottuvat vaaleansinisinä riveinä. Shift pohjassa kursorin asettaminen tietylle riville lisää kerralla kaikki edellisen kursoririvin ja valitun kursoririvin väliset resurssit mukaan valittuihin. Kuitenkin jos kaikki tällä välillä kuuluvat jo valintoihin, ne poistetaankin valituista.

Valinnat onnistuvat myös näppäimistön kautta. Mahdollisia tapoja valita ovat valitse kaikki (select all), poista valinnat (deselect) ja käänteinen valinta (invert selection).

Prioriteetti: pyritään toteuttamaan

5.2.9 Uuden osajoukon muodostaminen resurssitaulukosta

Valitut alkiot voi siirtää uuteen resurssitaulukkoon uudeksi osajoukoksi ja ne voi myös nimetä.

Prioriteetti: pyritään toteuttamaan

5.2.10 Resurssin yhteyksien avaaminen uutena osajoukkona

Käyttäjä voi muodostaa uuden osajoukon tiettyyn resurssiin viittaavista resursseista tuplaklikkaamalla resurssin yhteyksien lukumäärää resurssitaulukosta.

Prioriteetti: toteutetaan, jos jää aikaa

5.2.11 Resurssitaulukon asettelun tallentaminen

Käyttäjä pystyy tallentamaan yhden resurssitaulukon asettelun, "layoutin", jolloin ohjelmaa käytettäessä asiakas voi muokata näkymänsä juuri hänelle sopivaksi.

Käyttäjän oman layoutin lisäksi resurssiselain pitää tallessa aina oletusasetelman, johon käyttäjä voi halutessaan palata.

Prioriteetti: toteutetaan, jos jää aikaa

5.3 Resurssin tietojen muokkaus

Käyttäjä pystyy muokkaamaan lähes kaikkia resurssin attribuutteja. Osa arvoista ei ole muokattavissa. Tällöin ne ovat joko automaattisesti luotuja tai automaattisesti laskettuja arvoja. Muokkaus toteutuu joko resurssitaulukon tai lomakenäkymän kautta.

Prioriteetti: toteutetaan

5.3.1 Resurssin tietojen muokkaus resurssitaulukossa

Resurssiselaimessa voi editoida yksittäisen resurssin tietoja suoraan taulukossa valitsemalla taulukon solun.

Prioriteetti: pyritään toteuttamaan

5.3.2 Monen resurssin yhtäaikainen muokkaus resurssitaulukossa

Käyttäjä voi muokata useamman asteikkotyypin resurssin attribuuttia resurssitaulukon kautta valitsemalla ne kaikki samanaikaisesti. Tällöin käyttäjän vaihtaessa jonkin resurssin attribuutin arvoa (+ tai -), vaihtuu samalla myös muiden resurssien kyseisen attribuutin arvo.

Prioriteetti: toteutetaan, jos jää aikaa

5.3.3 Resurssin tietojen muokkaus verkkonäkymässä

Verkkonäkymässä voi hetkellisesti muokata resurssien tietoja. Suoritetut muutokset eivät kuitenkaan välity käytettävään tietokantaan.

Prioriteetti: toteutetaan, jos jää aikaa

5.3.4 Resurssin tietojen muokkaus lomakenäkymässä

Resurssiselaimen kautta käyttäjä voi avata erinäisen lomakenäkymän, jonka kautta resurssin attribuutit ovat muokattavissa. Lomakenäkymässä voi muokata myös resurssin viitteitä.

Prioriteetti: toteutetaan

5.3.5 Resurssin lisääminen

Resurssiselaimen päävalikon resurssi-valikosta voi valita toiminnon “uusi resurssi”, joka luo uuden resurssin ja avaa sen lomakenäkymässä muokkaustilassa. Samalla käyttäjä voi luoda resurssille viitteitä toisiin olemassa oleviin resursseihin ja muokata resurssin attribuuttien arvoja.

Prioriteetti: toteutetaan

5.3.6 Resurssien yhdistäminen

Resurssiselaimessa voi yhdistää kaksi solmua yhdeksi. Tällöin käyttäjän on valittava ns. parent-solmu, johon toinen solmu ja kaikki sen kaaret yhdistetään.

Prioriteetti: toteutetaan, jos aikaa jää

5.3.7 Resurssien puolittaminen

Resurssiselaimessa voi valita resurssin puolitettavaksi kahdeksi eri resurssiksi. Tätä ennen voidaan valita yksi tai useampi viite siirrettäväksi uuteen resurssiin. Valitut viitteet eivät enää osoita vanhaan resurssiin.

Prioriteetti: toteutetaan, jos aikaa jää

5.3.8 Resurssin poisto

Resurssiselaimessa voi poistaa resurssin. Tällöin kaikki resurssiin liittyvät viitteet myös poistetaan.

Resurssin voi poistaa resurssitaulukon riviltä suoraan tai lomakenäkymän kautta.

Prioriteetti: toteutetaan, jos aikaa jää

5.3.9 Resurssin attribuuttien automaattinen päivitys

Resurssien laskennalliset attribuutit päivitetään tapahtumalokista automaattisesti.

Prioriteetti: pyritään toteuttamaan

5.3.10 Resurssien välisten viitteiden eheyden tarkastus

Resurssiselain tarkastaa käyttäjän käskystä koko tietokannan viitteiden eheyden.

Prioriteetti: toteutetaan, jos jää aikaa

5.4 URI – taulukko

Resurssiselaimessa on taulukko, jossa näkyy valittuun resurssiin liittyvät URI:t ja niiden attribuutit.

Prioriteetti: toteutetaan

5.4.1 URI:n lisäys

URI-tilukoon voi lisätä uuden URI:n antamalla sille URI-merkkijonon, josta attribuutit

jäsennetään automaattisesti.

Prioriteetti: toteutetaan

5.4.2 URI:n poisto

URI-taulukosta voi poistaa URI:n.

Prioriteetti: toteutetaan

5.4.3 URI:n muokkaus

URI-taulukossa olevia attribuutteja voi muokata

Prioriteetti: pyritään toteuttamaan

5.4.4 URI:n avaaminen

URI:n voi avata URI-taulukosta tuplaklikkaamalla tai painamalla Enter:iä.

Prioriteetti: toteutetaan

5.5 Vapaamuotoinen SQL-kysely

Käyttäjä pystyy luomaan vapaamuotoisen SQL-lauseen, joka suoritetaan sillä hetkellä käytössä olevaan tietokantaan, jonka mahdolliset tulokset esitetään SQL-näkymässä. SQL-lauseen muodostaminen on *täysin* käyttäjän omalla vastuulla.

Prioriteetti: toteutetaan, jos jää aikaa

5.6 Verkkohakukoneet

Resurssiselaimen kautta käyttäjä pystyy hakemaan määrittelemillään hakukoneilla Internetistä tietoa liittyen valittuun resurssiin.

Prioriteetti: ei toteuteta

5.7 Vapaasanahaku dokumenttien sisällön perusteella

Resurssiselaimen avulla voi etsiä tietoja tekstidokumenttien sisältä. Vapaasanahaku toimisi Lucene-komponentin avulla. Käyttäjä voisi etsiä haluamaan merkkijonoa jokaisesta tietokannan sisältämästä URI:sta.

Prioriteetti: ei toteuteta

5.8 Tapahtumien seuranta ja tallennus

Osa resursseihin liittyvistä tapahtumista tallennetaan loki-tiedostoon. Tallennettavat tapahtumat ovat: resurssin esiintyminen hakutuloksessa, resurssin liittäminen osajoukkoon, resurssin avaaminen lomakenäkymään, resurssin avaaminen oletusohjelmalla, resurssin liittäminen verkkonäkymään, resurssin attribuuttien muuttaminen.

Prioriteetti: toteutetaan

5.9 Resurssiselaimen statuspaneeli

Resurssiselaimen statuspaneelissa näkyy kursoririvinumero sekä hakutuloksen, valittujen ja resurssileikepöydän koot.

Prioriteetti: pyritään toteuttamaan

5.10 Resurssileikepöytä

Käyttäjä pystyy siirtämään resursseja ohjelman sisällä copy/cut-paste toiminnolla. Resurssin leikkaaminen tulosjoukosta poistaa sen vain osajoukosta, ei itse tietokannasta.

(Osien 5.10.* tarkennuksissa R=resurssitaulukko, V=valitut resurssit, L=leikepöytä)

Prioriteetti: toteutetaan

5.10.1 Kopiointi

Käyttäjä voi kopioida resurssitaulukosta valitsemansa resurssit leikepöydälle.

Tapahtuma: $L=V$

Prioriteetti: toteutetaan

5.10.2 Leikkaaminen

Käyttäjä voi leikata myös valitsemansa resurssit pois, jolloin ne poistetaan resurssitaulukon näymästä/hakutuloksesta.

Tapahtuma: $C=V$; $R=R-V$

Prioriteetti: toteutetaan, jos aikaa jää

5.10.3 Liittäminen

Käyttäjä voi liittää leikepöydällä olevat resurssit resurssitaulukkaan.

Tapahtuma: $R=R+L$

Prioriteetti: toteutetaan

5.10.4 Lisääminen

Käyttäjä voi lisätä leikepöydälle valittuja resursseja.

Tapahtuma: $L=L+V$

Prioriteetti: toteutetaan, jos aikaa jää

5.10.5 Vaihtaminen

Käyttäjä voi vaihtaa leikepöydällä olevat resurssit resurssitaulukkaan, jolloin resurssitaulukossa olevat resurssit siirtyvät leikepöydälle.

Tapahtuma: $C=R$; $R=C$

Prioriteetti: toteutetaan, jos aikaa jää

5.10.6 Joukko-opin mukainen leikkaus

Käyttäjä voi suorittaa joukko-opin mukaisen leikkauksen valittujen resurssien ja leikepöydällä olevien resurssien kesken.

Tapahtuma: $V=L \cap V$

Prioriteetti: toteutetaan, jos aikaa jää

5.10.7 Suodattaminen

Käyttäjä voi poistaa valituista resursseista leikepöydällä olevat resurssit.

Tapahtuma: $V=V-L$

Prioriteetti: toteutetaan, jos aikaa jää

5.11 Tietokannan valinta

Ennen kuin yhtään hakua on tehty, käyttäjä pystyy valitsemaan käytettävän tietokannan.

Prioriteetti: toteutetaan

5.12 Resurssin metadatan automaattinen luonti metadatan sisältöä jäsentämällä

Resurssiselain jäsentää metadatasta automaattisesti joitakin siihen liittyviä attribuutteja. Esimerkiksi tekstitiedostosta selain voisi saada selville ainakin luontipäivämäärän. Lisäksi resurssiselaimen jäsentäjä kävisi tekstitiedoston sisältöä läpi etsien sieltä muita mahdollisia attribuutteja, kuten dokumentin kirjoittajan, nimen, jne.

Prioriteetti: ei toteuteta

5.13 Pikanäppäinten toteutus

Suurinta osaa resurssiselaimen toiminnoista pitää myös pystyä käyttämään pikanäppäinten avulla.

Kaikki paitsi vieritykseen liittyvät toiminnot löytyvät resurssiselaimen päävalikosta. Jokaiselle valikkotoiminnolle on pikanäppäin.

Hiiren oikealla nappulla auakeaa valikko sillä hetkellä relevanteimmista toiminnoista.

Prioriteetti: pyritään toteuttamaan

5.14 Tekstikenttä hakusanoille

Resursseilla on avainsanoille kenttä, johon haun voi myös kohdistaa. Avainsanakenttä on nähtävissä ja muutettavissa lomakenäkymässä.

Prioriteetti: pyritään toteuttamaan

5.15 Testausviitteiden luonti

Käyttäjällä on mahdollisuus luoda valittuun resurssikantaan mielevaltainen määrä viitteitä

kannassa olevien resurssien välille.

Prioriteetti: toteutetaan

6. Ei toiminnalliset vaatimukset

6.1 HSQLDB rajapinta

HSQLDB:tä varten tarvitaan rajapinta, joka mahdollistaa resurssien haun, lisäämisen ja poistamisen tietokannasta riippumatta alla olevan HSQLDB-tietokannan tyypistä.

Prioriteetti: pyritään toteuttamaan

6.2 GUESS rajapinta

GUESS-komponentin ja käyttöliittymän väliseen kommunikointiin tarvitaan toimiva rajapinta.

Prioriteetti: pyritään toteuttamaan

6.3 Hakutuloksien järjestämisen ja vierittämisen nopeus

Hakutuloksien järjestämisen ja vierittämisen pitää olla nopeaa. Toimintoja hidastavat osat voidaan piilottaa, jolloin ne kytkeytyvät pois päältä.

Prioriteetti: toteutetaan, jos jää aikaa

6.4 Connections, fromX, toX avaamisen nopeus

Resurssien viitteitä esittävien ikkunoiden avaaminen pitää olla nopeaa.

Prioriteetti: toteutetaan, jos jää aikaa

6.5 Hakujen nopeus

Haun suorituksen keston tulee olla suhteellinen haku- ja järjestysehtojen monimutkaisuuteen ja tulosjoukon haluttuun maksimikokoon.

Prioriteetti: toteutetaan, jos jää aikaa

6.6 Toteutuskieli englanti

Resurssiselain toteutetaan englanniksi

Prioriteetti: toteutetaan

7. Toimintaympäristövaatimukset

7.1 Java

Sunin Java versio 1.5 pitää olla asennettuna.

7.2 Eclipse SWT

Eclipse SWT pitää olla saatavilla (ohjeiden mukaan).

7.3 HSQLDB

HSQLDB pitää olla asennettuna (ohjeiden mukaan).

7.4 GUESS

GUESS pitää olla asennettuna (ohjeiden mukaan).

7.5 Muu toimintaympäristö

Resurssiselain toteutetaan Javalla ja mahdolliset käyttöjärjestelmäriippuvat piirteet dokumentoidaan. Ohjelma testataan tietojenkäsittelytieteen laitoksen Linux/KDE-ympäristössä.

7.6 Vaatimusten prioriteetit

7.6.1 Toteutetaan

- 5.1 Verkkonäkymä
- 5.2 Resurssitaulukko
 - 5.2.1 Resurssitaulukon kentät
 - 5.2.2 Resurssitaulukkon hakeminen tietokannasta
 - 5.2.4 Resurssitaulukon järjestäminen attribuuttien mukaisesti
 - 5.2.7 Lomakenäkymän avaaminen resurssitaulukosta
- 5.3 Resurssin tietojen muokkaus
 - 5.3.4 Resurssin tietojen muokkaus lomakenäkymässä
 - 5.3.5 Resurssin lisääminen
- 5.4 URI-taulukko
 - 5.4.1 URI:n lisäys
 - 5.4.2 URI:n poisto
 - 5.4.4 URI:n avaaminen

5.8 Tapahtumien seuranta ja tallennus

5.10 Resurssileikepöytä

5.10.1 Kopiointi

5.10.3 Liittäminen

5.11 Tietokannan valinta

5.15 Testausviitteiden luonti

6.6 Toteutuskieli englanti

7.6.2 Pyritään toteuttamaan

5.1.1 Verkon laajennus

5.1.2 Resurssien lisääminen verkkonäkymään

5.1.3 Resurssin poisto verkkonäkymästä

5.2.3 Resurssitaulukon sisäinen etsintä

5.2.5 Resurssitaulukon sarakkeiden järjestäminen

5.2.8 Resurssien valitseminen resurssitaulukosta

5.2.9 Uuden osajoukon muodostaminen resurssitaulukosta

5.3.1 Resurssin tietojen muokkaus resurssitaulukossa

5.3.9 Resurssin attribuuttien automaattinen päivitys

5.4.3 URI:n muokkaus

5.9 Resurssiselaimen statuspaneeli

5.13 Pikanäppäinten toteutus

5.14 Tekstikenttä hakusanoille

6.1 HSQLDB rajapinta

6.2 GUESS-rajapinta

7.6.3 Toteutetaan, jos jää aikaa

5.1.4 Verkkonäkymän tallennus kuvaksi tai tiedostoon

5.2.6 Viimeisimmän haun toistaminen

5.2.10 Resurssin yhteyksien avaaminen uutena osajoukkona

5.2.11 Resurssitaulukon asettelun tallentaminen

5.3.2 Monen resurssin yhtäaikainen muokkaus resurssitaulukossa

5.3.3 Resurssin tietojen muokkaus verkkonäkymässä

5.3.6 Resurssien yhdistäminen

5.3.7 Resurssien puolittaminen

5.3.8 Resurssin poisto

5.3.10 Resurssien välisten viitteiden eheyden tarkastus

5.5 Vapaamuotoinen SQL-kysely

5.10.2 Leikkaaminen

5.10.4 Lisääminen

5.10.5 Vaihtaminen

5.10.6 Joukko-opin mukainen leikkaus

5.10.7 Suodattaminen

6.3 Hakutuloksien järjestämisen ja vierittämisen nopeus

6.4 Connections, fromX, toX avaamisen nopeus

6.5 Hakujen nopeus

7.6.4 Ei toteuteta

5.6 Verkkohakukoneet

5.7 Vapaasanahaku dokumenttien sisällön perusteella

5.12 Resurssin metadatan automaattinen luonti metadatan sisältöä jäsentämällä

8. Resurssien attribuutit ja mahdolliset arvot

Activity	Views + Loads + "Clicks" painokertoimella.
Added	Milloin lisätty tietokantaan.
CanoName	RawName:n päivittäjän kanonisoima muoto (ei näy käyttäjälle), merkkijono < 128.
Created	Dokumentin luontiajankohta.
Connections	Resurssiin suoraan liittyvien yhteyksien lukumäärä.
Found	Milloin oli ensimmäisen kerran hakutuloksessa.
FromX.. ToX..	Resurssiin suoraan liittyvien tietyn tyyppisten yhteyksien lukumäärä. (ToType, FromType – monta)
Imported	Milloin metadata tuotu omalle koneelle.
InSourceId	Metadatalähteen sisäinen tunniste (ei näy käyttäjälle), merkkijono < 256.
Loads	Kuinka monta kertaa ollut hakutuloksessa.
Loaded	Milloin viimeksi ollut hakutuloksessa.
Name	Tiedoston nimi.
Rank	Value + Connections + Activity
RawName	Alkuperäinen nimi (ei näy käyttäjälle), merkkijono < 512.
Recency	Activity, jota painotetaan tapahtumien tuoreudella.
ResourceClass	Merkkijono < 128, max 32 erilaista. Esim. Actor
Source	Metadatalähde (reader), merkkijono < 128.
Tag	Käyttäjän määrittelemä merkintä.
Type	Resurssin ResourceClass -kohtainen tyyppi. Esimerkiksi Person.
URIs	Resurssiin liittyvien URIen lukumäärä.
Value	Käyttäjän antama arvio välillä -5, +5
Viewed	Milloin viimeksi katseltu.
Views	Katselujen lukumäärä.

8.1 URI-taulukon attribuutit

Jokaisella resurssilla on lista siihen liittyvistä URI:sta. URI-lista sisältää jokaiselle URI:lle seuraavat attribuutit.

Type: mailto / homepage (index.html) / html / tiedostopäätte / ?

Name: nimi

Location: sijainti

Created: luontipäivämäärä

Imported: lisäyspäivämäärä

Found: löytämisspäivämäärä

URI: yksikäsitteinen URI

9. Järjestelmäarkkitehtuuri

Korkeimmalla tasolla järjestelmä jakautuu kolmeen komponenttiin, *Resurssikantaan*, *Lokittietojen kerääjään* ja käyttöliittymään. Lisäksi järjestelmän kanssa on yhteistoiminnassa rajapintojen välityksellä kolme ulkoista komponenttia.

Resurssikannasta on JDBC-rajapinnan välityksellä yhteys varsinaiseen tietokantaan, joka on HSQLDB-niminen valmis komponentti. Rajapinta tietokantaan on siis kaksiosainen, ensin tiedonkäsittelypyynnöt osoitetaan Resurssikannalle rajapinnan kautta, joka puolestaan osoittaa pyynnön tietokantaan JDBC:n kautta.

Lokittietojen kerääjä tallentaa tietoja joistain käyttäjän toiminnoista.

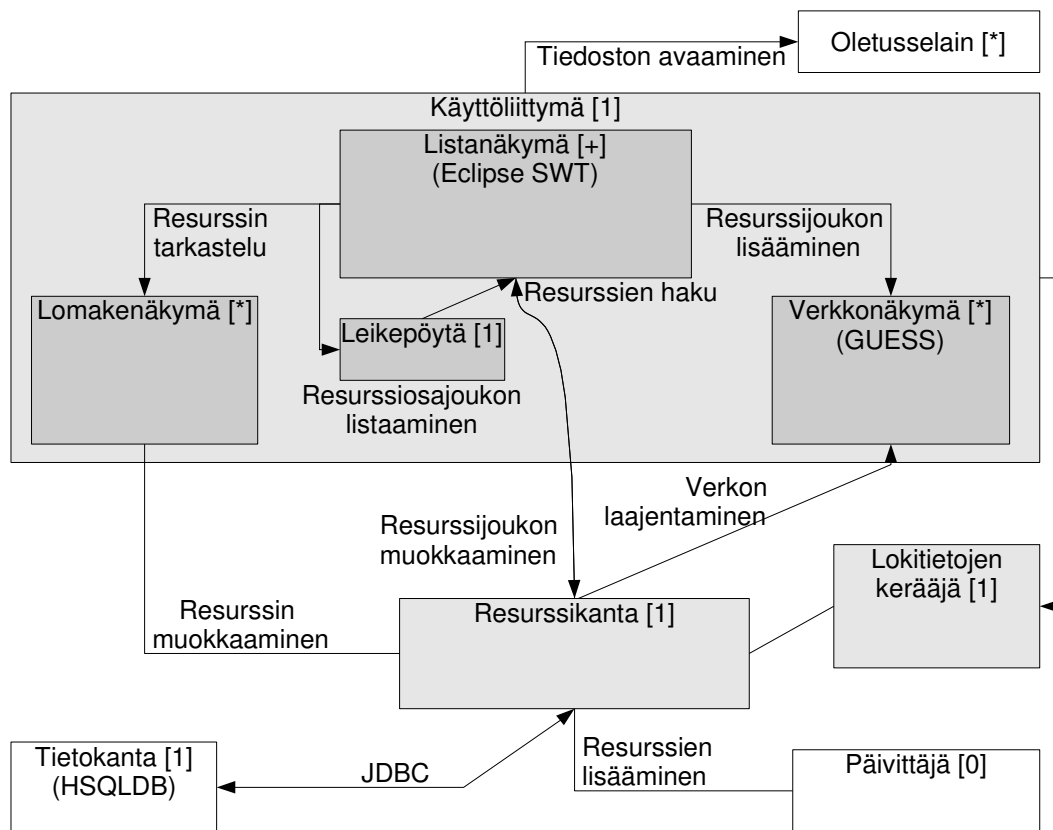
Käyttöliittymä muodostuu neljästä alikomponentista jotka ovat *Listanäkymä*, *Verkkonäkymä*, *Lomakenäkymä* ja *Leikepöytä*. Listanäkymää käytetään resurssihakujen muodostamiseen ja tulosjoukkojen tarkasteluun; komponentissa käytetään apuna Eclipsen SWT:n kirjastoa. Lomakenäkymässä voidaan tarkastella ja muokata yksittäistä resurssia. Verkkonäkymässä resurssijoukko esitetään kaksiulotteisena verkkona; toteutuksessa käytetään apuna GUESS-nimistä valmista komponenttia. Leikepöytä koostuu sovelluksen ja/tai käyttöjärjestelmän leikepöytätoiminnoista.

Päivittäjä on ulkoinen komponentti, joka esikäsittelee resurssitietoa järjestelmän vaatimaan muotoon, ja jolla voidaan lisätä resursseja järjestelmään Resurssikannan rajapinnan kautta.

Oletusselain on ulkoinen komponentti, joka on se oletusarvoinen ohjelma, jolla resurssin viittaama tietosisältöä tarkastellaan järjestelmän ulkopuolella (esimerkiksi resurssi viittaa PDF-dokumenttiin, jonka oletusselain on Adobe Reader).

Tietokanta on ulkoinen, HSQLDB niminen komponentti.

*Kuvassa kaaret kuvaavat datavirtoja, jos kaari on suuntaamaton, ei datavirran suuntaa ole määrätty. Hakasuissa ovat komponenttien lukumäärät sovelluksen suoritusajana (+ = "vähintään yksi", * = "mikä tahansa määrä").*



10. Järjestelmän elinkaari

Projektiryhmä suunnittelee, testaa ja toteuttaa ohjelmiston dokumentissa esitettyjen vaatimusten pohjalta. Vaatimukset on kerätty kokouksissa ja asiakastapaamisissa tehtyjen pöytäkirjojen perusteella ja ne ovat asiakkaan vahvistamia.

Projektin toteutusvaiheen loppupuolella pidetään lyhyt betatestausvaihe, jonka perusteella voidaan varmentaa asiakkaan vaatimusten toteutumista.

Projektin päättymisen jälkeen asiakas vastaa ohjelmiston ylläpidosta. Projektiryhmä pyrkii tekemään ohjelmistosta sellaisen, että ohjelmiston jatkokehitys olisi myös mahdollista. Jatkokehitystä silmällä pitäen on tärkeää, että ohjelmisto dokumentoidaan ja testataan hyvin.

Ominaisuuksien toteuttamisvaiheessa on muistettava, että on parempi tuottaa pieni hyvin toimiva ohjelmisto kuin huonosti toimiva monipuolinen ohjelmisto.

11.Liitteet

11.1 Järjestelmään liittyvät komponentit

11.1.1 GUESS – the Graph Exploration System

Lyhyesti: avoimeen lähdekoodiin perustuva ohjelma 2-ulotteisten verkkojen luomiseen.

Kotisivu: <http://graphexploration.cond.org/>

11.1.2 Eclipse SWT Table

Lyhyesti: Eclipsen oma komponentti taulukkojen luomiseen.

Kotisivu: <http://www.eclipse.org/swt/>

API:

<http://help.eclipse.org/help31/nftopic/org.eclipse.platform.doc.isv/reference/api/org/eclipse/swt/widgets/Table.html>

11.1.3 HSQL database engine

Lyhyesti: GUESS:in käyttämä Java-pohjainen tietokanta.

Kotisivu: <http://www.hsqldb.org/>