

SAIMAAN AMMATTIKORKEAKOULU
Tekniikka Lappeenranta
Tietotekniikan koulutusohjelma
Ohjelmistotekniikan suuntautumisvaihtoehto

Claire Tamper

SINSEWEB - SEMANTTINEN HAKUKONE

Opinnäytetyö 2011

TIIVISTELMÄ

Claire Tamper

Sinseweb - semanttinen hakukone, 81 sivua, 2 liitettä

Saimaan ammattikorkeakoulu, Lappeenranta

Tekniikka, Tietotekniikan koulutusohjelma

Ohjelmistotekniikan suuntautumisvaihtoehto

Opinnäytetyö 2011

Ohjaajat: Lehtori Martti Ylä-Jussila, Saimaan ammattikorkeakoulu

Toimitusjohtaja Mikko Mäkelä, Tiksis Technologies Oy

Internet ja sen sisällön määrä kasvaa jatkuvasti ja tiedon hakeminen Internetistä muuttuu jatkuvasti haastavammaksi. Luotettavan ja tiedonhakijan haluaman tiedon löytämiseksi on käytettävä monimutkaisia hakustrategioita ja valittava oikeanlainen hakupalvelu. Internetin suosituimmiksi hakupalveluiksi ovat nousseet hakukoneet. Viime vuosien aikana on alkanut tulla ihmisten käyttöön semanttisia hakukoneita.

Semanttinen hakukone poikkeaa perinteisestä hakukoneesta siihen lisätyn semantiikan avulla. Semantiikalla tarkoitetaan tässä yhteydessä, että hakukone ymmärtää hakusanojen ja lauseiden merkityksen. Tämän opinnäytetyön tarkoituksena on rakentaa yksinkertainen semanttinen hakukone nimeltä Sinseweb. Sinseweb-projekti on tehty yhteistyössä Tiksis Technologies Oy nimisen yrityksen kanssa.

Sinseweb-projektiryhmä koostui Tiksis Technologies Oy:n työntekijöistä. Opinnäytetyö keskittyy omaan panokseeni koskien tuotteen suunnittelua, testausta ja toteutusta. Projektinhallinnassa sovellettiin erilaisia metologioita useista erilaisista elinkaarimalleista, kuten esimerkiksi vesiputousmalli ja RUP.

Projektin kussakin vaiheessa käytettiin erilaisia työkaluja ja menetelmiä. Esitutkimusvaiheessa todettiin Sinseweb-järjestelmä toteutuskelpoiseksi. Määrittelyvaiheessa määriteltiin projekti ja sen lopputuote. Suunnitteluvaiheen alkuvaiheessa tehtiin myös pieni tutkimus semanttisesta Internetistä ja hakukoneista. Toteutusvaiheessa valmistettiin suunnitelmien ja määritelmien mukainen lopputuote käyttämällä hakukoneen rakennukseen erilaisia web-teknologioita, kuten Drupal ja PHP. Viimeisenä vaiheena projektissa oli tuotteen testaus. Testaus suoritettiin opiskelijavoimin ja se onnistui.

Projektin lopputuloksena saatiin aikaiseksi semanttinen hakukone. Sinseweb-projektin aikana kuitenkin Sinseweb-järjestelmä saatiin keskeisiltä osilta määriteltä, suunniteltua, toteutettua ja testattua. Ongelmitta projektissa ei selvitty, mutta loppujen lopuksi saatiin aikaan lupaava lopputuote. Lopputuotteessa on kuitenkin vielä varaa kehitykselle vaikka se onkin jo nyt lupaava hakukone.

Asiasanat: Hakukone, semantiikka, Drupal, Internet

ABSTRACT

Claire Tamper

Sinseweb - a semantic search engine, 81 pages, 2 appendices

Saimaa University of Applied Sciences, Lappeenranta

Technology, Information Technology

Software Engineering

Final Year Project 2011

Instructors: Lecturer Martti Ylä-Jussila, Saimaa University of Applied Sciences

CEO Mikko Mäkelä, Tiksis Technologies Oy

The internet is continuously growing along with the quantity of the information it embodies. Searching information from the internet keeps getting more challenging. To find information that is both reliable and wanted by the searcher requires him to use complex search strategies and correct search services. The most popular search services nowadays are the search engines. Within the last few years people have begun to use the semantic search engines.

Semantic search engines differ from classical search engines because they take into account the semantics of the search. In this context semantics means that the search engine is able to understand the meaning of the words and sentences put into the search engine. The purpose of this thesis was to build a simple semantic search engine called Sinseweb. The Sinseweb project was carried out in collaboration with a company called Tiksis Technologies Oy .

The project team comprised of employees of Tiksis Technologies Oy. The thesis concentrated on the author's contributions to the design, implementation and testing of the product. The project's life cycle used appropriate methodologies from several different development models such as waterfall model and RUP.

In each phase of the project different tools and methods were used. In the feasibility study the Sinseweb system was confirmed to be viable. In the requirements specification phase the project and the end product were defined. The first task of the design phase was to make a study about the semantic web and search engines. In the implementation phase the defined and designed product was produced. It was done by using different web technologies such as Drupal and PHP. The last phase of the project was the testing phase. The testing phase was completed successfully with the help of other students.

In the end the semantic search engine was created. During the Sinseweb project the Sinseweb system's central parts got defined, designed, implemented and tested. Despite of the problems being encountered along the way a promising end product was created. There is still room for development in the end product even though it is already looking like an up-and-coming search engine.

Keywords: Search engines, semantics, Drupal, Internet

SISÄLTÖ

1	JOHDANTO	10
1.1	Tausta ja tarkoitus	10
1.2	Rajaus	11
1.3	Asiakkaan toiminnan kuvaus	11
2	TIEDON HAKU INTERNETISSÄ	13
2.1	Internetin hakupalvelut.....	14
2.1.1	Aiuehakemistot.....	15
2.1.2	Organisaatio- tai palvelinhakemistot.....	16
2.1.3	Erikoishakemistot ja tietokannat.....	17
2.1.4	Portaalit.....	18
2.1.5	Hakukoneet.....	18
2.2	Tiedonhakustrategiat	20
2.3	Internetin tiedonhakuun liittyviä ongelmia	21
2.4	Ongelmista ratkaisuihin	21
3	SEMANTIikka.....	23
3.1	Semantiikka käsitteenä.....	23
3.2	Semantiikka tietoteknisissä sovelluksissa.....	24
3.2.1	Ontologia.....	24
3.2.2	Semanttinen verkko.....	25
3.3	Semanttiset hakukoneet	26
4	OHJELMISTOPROJEKTIN HALLINTA.....	28
4.1	Vaihejakomallit ja elämäнкаari	28
4.1.1	Vesiputousmalli	28
4.1.2	Prototyyppimalli.....	29
4.1.3	EVO-malli	31
4.1.4	RUP.....	32
4.2	Ohjelmistokehityksen vaiheet	33
4.2.1	Esitutkimus.....	33
4.2.2	Määrittely.....	34
4.2.3	Suunnittelu	34
4.2.4	Toteutus ja testaus	35
5	TYÖSSÄ KÄYTETYT TEKNIIKAT	37
5.1	Käsiteanalyysi.....	37
5.2	UML	38
5.3	Ohjelmointikielet	40
5.3.1	HTML	40
5.3.2	PHP.....	41
5.3.3	JavaScript ja jQuery	42
5.3.4	Ajax	42
5.4	SQL, RDMS ja MySQL	44
5.5	Ohjelmat, komponentit ja järjestelmät.....	45
5.5.1	Internet-selain	45
5.5.2	Mozilla Firefox	45
5.5.3	Internet Explorer.....	46
5.5.4	Google Chrome	47
5.5.5	Vim editor	48
5.5.6	Drupal.....	48
5.5.7	Yahoo! Boss API	50

5.5.8	PhpMyAdmin.....	51
6	SINSEWEB – PROJEKTIN VAIHEET	52
6.1	Organisointi	52
6.2	Vaiheistus	54
6.3	Projektin kulku	54
6.3.1	Esitutkimus.....	54
6.3.2	Projektin aloitus.....	54
6.3.3	Määrittely.....	55
6.3.4	Suunnittelu	55
6.3.5	Toteutus	57
6.3.6	Testaus	59
6.3.7	Projektin lopetus.....	59
7	SINSEWEB – JÄRJESTELMÄN ESITTELY	61
7.1	Rakenne	61
7.1.1	Fyysinen rakenne	61
7.1.2	Järjestelmän rakenne	63
7.2	Toiminta.....	64
7.2.1	Hakeminen	65
7.2.2	Hakutavoitteen määrittäminen.....	65
7.2.3	Haun tekeminen	66
7.2.4	Hakutuloksen katselu ja selaus	67
7.2.5	Hakuasetusten määrittäminen.....	70
8	YHTEENVETO	72
	KUVAT	75
	LÄHTEET.....	76

TERMISTÖ

CSS	CSS (Cascading Style Sheets) on kieli dokumentin ulkoasun muotoilua varten. Dokumentilla tarkoitetaan esimerkiksi web-sivua. CSS-muotoilukielen avulla on mahdollista muuttaa web-sivun elementtien (kuten painikkeiden, tekstikenttien, tekstin tai kuvien) ulkoasun asettelua ja muotoilua. (Meyer 2006.)
Drupal	Drupal on WWW-pohjainen ohjelmistokehitys- ja sisällyönhallintajärjestelmä. Ohjelmisto on kirjoitettu PHP-ohjelmointikielellä ja toimii erilaisissa tietokantaympäristöissä, esimerkiksi MySQL. (Wikipedia (suomi) sisällyönhallintajärjestelmät 2009.)
Graafi	Graafi on solmuista ja kaarista koostuva tietorakenne. Jokaisella solmulla voi olla yksi tai useampi kaari. Visuaalisoitu graafi on puolestaan verkko, joka koostuu kaarista ja solmuista. Siinä kaaret on kuvattu viivoina ja solmut esimerkiksi suunnikkaina. Jokainen graafin solmu voi sisältää jonkun merkkijonon. (Lilja.)
Haittaohjelma	Haittaohjelma (malware) on ohjelma, joka on luotu käyttäjän huomaamatta muun muassa muokkaamaan tai vahingoittamaan muita tietokoneella olevia ohjelmia. Tietokonevirukset ja vakoiluohjelmat ovat esimerkkejä haittaohjelmista. (Christodorescu, Jha, Maughan, Song & Wang.)
Hakubotti	Hakubotti on ohjelma, joka etsii internetistä sivustoja, joita se tallentaa sille valittuun tietovarastoon (esimerkiksi tietokantaan). Tästä tietovarastosta hakukoneet etsivät hakustrategioiden avulla hakutuloksia käyttäjille.
Hakustrategia	Hakustrategia on toimintatapa. Sillä hakutavoittelauseesta muodostetaan hakulause, jolla haetaan suoraan vastaavia tuloksia järjestelmän omasta tietokannasta ja Yahoo! BOSS API –palvelusta. Hakustrategian tavoitteena on muodostaa hakutavoitetta mahdollisimman hyvin kuvaava hakulause, jolla löydetään mahdollisimman kattavasti hakutavoitetta vastaavia käyttökelpoisia hakutuloksia (eli linkkejä sivuille) internetistä ja järjestelmästä.
Hakutavoite	Käyttäjän määrittelemä haun tavoite. Toisin sanoen se on se mitä käyttäjä haluaa löytää. Hakutavoite voi olla esimerkiksi ”koiran ruokinta”.

Hakutavoitelause	Käyttäjän syöttämä hakutavoite järjestelmälle lauseke muodossa. Hakutavoitelause voi olla esimerkiksi ”Vie koira rokotettavaksi eläinlääkärille”.
Java	Java on Sun Microsystems -yhtiön kehittämä olioperustainen, syntaksiltaan C++ -kielen kaltainen ohjelmointikieli. (Jaakkola 2000.)
Käsite	Käsite on sanan tai termin merkityssisältö. Käsitteellä voidaan myös tarkoittaa semanttisten verkkojen solmuja, joita kuvataan kaavioissa niihin liittyvien suhteiden avulla.
Leivänmurupolku	Leivänmurupolku tai lyhyesti murupolku (Breadcrumbs) on haun yhteydessä laadittava navigoitava hakupolku, joka kuvaa sen kuinka käyttäjä on edennyt hakutavoitteensa määrittelyssä.
Malli	Malli on verbistä ja objektista koostuva kokonaisuus. Mallin avulla järjestelmä tietää mitkä sanat liittyvät toisiinsa. Malli voivat muodostaa esimerkiksi sanat ”koiran” ja ”ruokinta”, jolloin malli on nimeltään ”koiran ruokinta”.
Normi	Normi on jonkin yhteisön jäseniä sitova toimintaohje.
Objekti	Objekti on lauseen jäsen, joka ilmaisee tekemisen kohteen. Sinseweb-järjestelmässä objekti on osa muun muassa hakutavoitelauseita ja malleja.
Ontologia	Ontologiat ovat tietojenkäsittelyssä luokitteluja, joita on käytetty erityisesti automaattisen tietojenkäsittelyn yhteydessä. Ontologioilla määritellään tietyn aihealueen käsitteistöjä. Usein ne muodostetaan olemassa olevien sanastojen avulla. Ontologioilla määritellään käsitteet ja niiden suhteet. (Wikipedia (suomi) luokittelujärjestelmät 2009.)
Rajapinta	Rajapinta (interface) on liittymäpinta, jonka kautta on mahdollista siirtää tietoja ohjelmistojen välillä tai ohjelmiston ja käyttäjän välillä. Ohjelmistotekniikassa rajapinnalla tarkoitetaan myös operaatiokokeelmaa, joita käytetään määrittelemään ilmeentymän tarjoama palvelu. (Mäkinen 2008; Haikala & Märijärvi 2004.)
RSS-syöte	RSS-syöte (RSS feed) on dokumentti, joka sisältää tietoa web-sivun sisällöstä. RSS-syötteen avulla on mahdollista seurata päivityksiä ja muutoksia erilaisilla web-sivuilla. (Search Engine Partner 2010.)

Salaus	Salaus eli kryptaus on tiedon salaamista. Salaamisen tarkoituksena on saada tieto muotoon, josta (ideaalita-pauksessa) vain tiedon vastaanottaja saa palautettua alkuperäisen tiedon. (Ala-Mutka, Rintala, Savikko & Palviainen.)
Semantiikka	Semantiikka eli merkitysoppi on kielitieteen osa-alue, joka keskittyy merkitysten tutkimiseen. Tutkittavana voivat olla sanojen, kieliopillisten ainesten, lauseiden, tekstien ja myös ilmauksen merkitykset. Merkityksen tutkimisessa oleellisia ovat ilmaisujen väliset suhteet, kuten antonymia, homonymia, hyponymia, meronymia, polysemia ja synonymia. (Muikku-Werner & Savolainen 1999; WikiWiki 2005; Wikipedia (suomi) semantiikka 2009.)
Sinse	Sinse (Sinse is not a search engine) on projekti, jonka tarkoituksena on luoda konkreettisesti uudenlainen hakukonekonsepti tavallisille Internetin käyttäjille.
Sinseweb	Sinseweb on Sinse-projektin osaprojekti. Sen tarkoituksena on luoda yksi käyttöliittymä Sinse-projektille. Siinä myös aloitetaan semantiikan tutkiminen ja rakennetaan ensimmäiset semanttiset avusteet Sinse-järjestelmään.
Spiraalimalli	Spiraalimalli on yksi ohjelmistotuotannon elikaarimalleista. Spiraalimallissa ohjelmistoprojekti etenee spiraalinkaltaisesti aloittaen sen keskipisteestä koko ajan siitä loitoten projektin kehittyessä. Spiraalimalli keskittyy riskeihin ja niiden minimointiin. Siinä ohjelmistoa kehitetään inkrementtaalisesti kierroksittain. Jokaisella kierroksella kehityksen edetessä minimoidaan myös projektin riskejä. (Tsui & Karam.)
Taksonomia	Taksonomia on luokittelujärjestelmä, jolla tarkoitetaan erilaisten asioiden hierarkkista luokittelua, luokittelujärjestelmää tai luokittelun periaatteita. Melkein mitä tahansa, kuten eläviä ja elottomia objekteja, paikkoja tai tapahtumia voidaan luokitella jonkinlaisen taksonomisen järjestelmän mukaan. (Wikipedia (suomi) Taksonomia 2009.)
Verbi	Verbi on tekemistä ilmaiseva sana. Sinseweb-järjestelmässä verbi on muun muassa osa mallia ja hakutavoitelausetta.
Verkkourkinta	Verkkourkinnalla tarkoitetaan jonkun käyttäjän tietojen, kuten salasanojen ja tunnusten, hankkimista urkkijan omaan käyttöön. Urkkija yleensä tekeytyy joksikin luo-

tettavaksi tai julkiseksi toimijaksi automatisoidulla tavalla. Automatisoitu tapa voi olla esimerkiksi sähköpostin käyttäminen. Yleensä verkkourkinnan tarkoituksena on saada selville esimerkiksi käyttäjän luottokortti tunnukset tai salasanoja ja tunnuksia. Urkkijoiden lähettämisessä sähköpostiviesteissä voidaan pyytää käyttäjää luovuttamaan vastaavia tietoja urkkijalle. (Jakobsson & Myers 2007.)

Web-konsepti

Web-konsepti on idea web-sovelluksesta. Se voi olla esimerkiksi uudenlainen sovellus, jonkalaista internetissä ei vielä ole.

XML

XML on lyhenne sanoista eXtensible Markup Language ja sen tarkoituksena on kuvata dataa. XML:n tarkoituksena on toimia laitteisto- ja ohjelmistoriippumattomana tiedonvälitystyökaluna kaikilla alustoilla. (Mikkonen.)

1 JOHDANTO

Sinse on Tiksis Technologies Oy:n kehittämä uudenlainen hakukonsepti. Sinse-hakukonseptin tavoitteena on tarjota internetin käyttäjille mahdollisuus saavuttaa tiedonhakemiseen liittyviä tavoitteitaan nopeammin ja helpommin kuin nykyisillä hakukoneilla on mahdollista. Se tarjoaa käyttäjän antamien tietojen, semanttisen toimintalogiikan ja semanttisten käyttäjäavusteiden avulla käyttäjän hakutavoitteisiin täsmäviä hakutuloksia. Sinse-hakukonsepti määrittelee uudenlaisen semanttisen hakukoneen, joka on hakemisen aikana jatkuvassa vuorovaikutuksessa käyttäjiensä kanssa.

Tämän opinnäytetyön tavoitteena on kehittää Sinse-hakukonseptista käyttökel-poinen ja selainriippumaton järjestelmä, joka julkaistaan myöhemmin internetis-sä.

1.1 Tausta ja tarkoitus

Sinse-projektin idea sai alkunsa vuonna 2003 Tiksis Technologies Oy:ssä. Vuonna 2005 käynnistyi ensimmäinen projekti, jonka tavoite oli toteuttaa Sinse-hakukonseptista julkaisukelpoinen järjestelmä. Sinse-hakukonsepti ei kuitenkaan kehittynyt tarpeeksi hakukonseptina, jotta siitä olisi saatu käyttäjiä puoleensavetävä järjestelmä. Aiempien toteutusyritysten tuloksena on syntynyt uudelleenkäytettäviä lähdekooditiedostoja ja jalostuneempia ideoita järjestelmän toteutuksesta. Keväällä 2009 Sinse-projekti käynnistettiin uudelleen. (Mäkelä, M. Toimitusjohtaja. Tiksis Technologies Oy. Lappeenranta. 7.8.2009. Henkilökohtainen tiedonanto.)

Sinse-projekti määriteltiin keväällä 2009 täsmällisesti. Määrittelyn yhteydessä järjestelmä jaettiin useampaan osakokonaisuuteen ja toteutus osaprojektiin. Näitä osaprojekteja ovat Scrollsearch, Sinseweb ja Weave-intergaation toteu-

tus. Tässä opinnäytetyössä keskitytään Sinne-hakukonseptin ja Sinseweb-osaprojektin version 1.0 kehittämiseen.

Sinseweb-osaprojektin tarkoituksena on kehittää selainriippumaton käyttöliittymä, joka tarjoaa tiedonhakupalveluita ja mahdollistaa sosiaalisen kanssakäynnin linkkien jakamisen muodossa järjestelmässä. Projektin tarkoituksena on myös tutustua, suunnitella ja lopuksi toteuttaa semanttinen tiedonhakupalvelu eli hakukone.

1.2 Rajaus

Tämä opinnäytetyö on rajattu toteuttamaan Sinseweb-järjestelmän versio 1.0, jossa keskitytään toteuttamaan yksinkertainen semanttinen hakukone, jonka avulla voidaan myös jakaa linkkejä käyttäjien kesken. Järjestelmän versio 1.0 hyödyntää järjestelmän semantiikkaa ja käyttäjien lisäämiä suosikkeja täsmällisempien hakutulosten hakemiseen. Järjestelmän käyttämät hakustrategiat on myös rajattu mahdollisimman yksinkertaisiksi.

Semantiikan suunnittelu ja toteuttaminen Sinsewebin versiossa 1.0 rajataan tietomallin suunnitteluun, semantiikan toteutuksen tutkimiseen ja semanttisen hakuavusteen toteuttamiseen. Hakuavuste auttaa käyttäjää määrittelemään hakutavoitteensa niin, että järjestelmä pystyy tulkitsemaan sen.

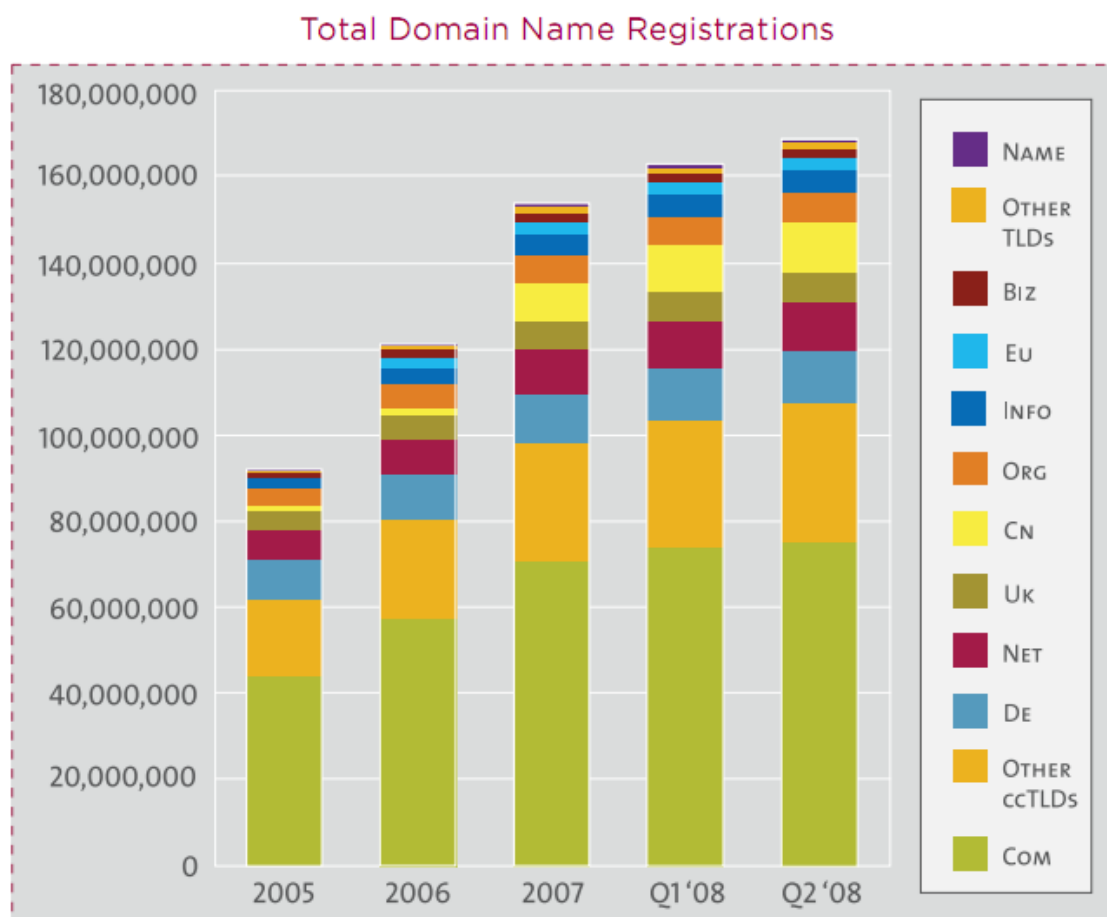
1.3 Asiakkaan toiminnan kuvaus

Järjestelmä tehdään Tiksis Technologies Oy:lle. Tiksis Technologies Oy tai lyhyesti vain Tiksis on pienikokoinen tietotekniikka alan yritys, joka suunnittelee ja kehittää muun muassa web-portaaleja, web-palveluja ja web-konsepteja asiakkaidensa tarpeisiin. Tämän lisäksi Tiksis Technologies Oy tarjoaa konsultointi- ja internetmarkkinointi palveluita. (Mäkelä, M. Toimitusjohtaja. Tiksis Technologies Oy . Lappeenranta. 7.8.2009. Henkilökohtainen tiedonanto.)

Tiksis Technologies Oy:n on perustanut Mikko Mäkelä vuonna 2004. Yrityksessä työskentelee tällä hetkellä 8 työntekijää, joista 7 tekee ohjelmistokehitystä ja suunnittelua. Yrityksen toimitusjohtajana toimii Mikko Mäkelä. (Mäkelä, M. Toimitusjohtaja. Tiksis Technologies Oy. Lappeenranta. 7.8.2009. Henkilökohtainen tiedonanto.)

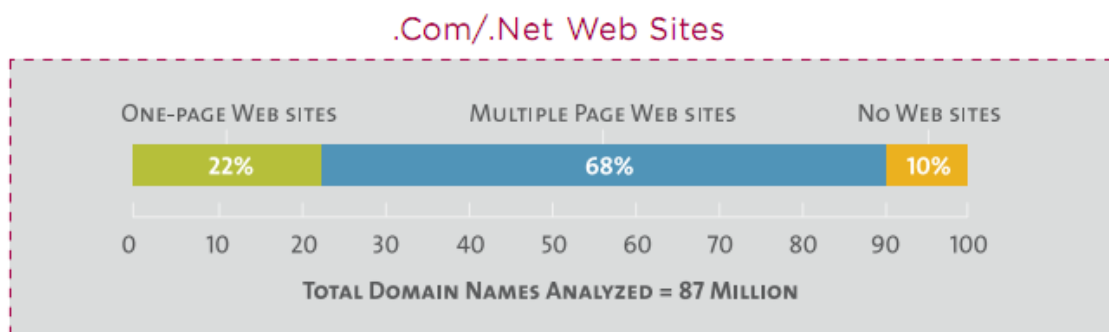
2 TIEDON HAKU INTERNETISSÄ

Internet on täynnä informaatiota. Siellä on tietoa lähes kaikilta aloilta ja kaikista aiheista. Internetissä on arvioitu olevan liki 200 miljoonaa verkkotunnusta. Erään verkkotunnusten määrää päivittäin tilastoivan sivuston mukaan 31.8.2009 menessä internetissä oli pelkästään .com päätteisiä verkkotunnuksia yli 81 miljoonaa kappaletta. Seuraavassa kuvassa on esitetty verkkotunnusten määrän kasvua vuodesta 2005 vuoden 2008 toiselle neljännekselle asti. (VeriSign 2008; DomainTools 2009; Kokkolan Kaupunginkirjasto.)



Kuva 2.1 Verkkotunnusten määrä vuoden 2008 toiselle neljännekselle (Q2') saakka. (VeriSign 2008.)

Näistä verkkotunnuksista esimerkiksi .com ja .net päätteisistä sivustoista jopa arvioilta noin 68 % sisältää useamman kuin yhden web-sivun ja noin 10 %:lle verkkotunnuksista ei ole verkkosivustoja lainkaan. Loput 22 % verkkotunnusten sivustoista sisältää vain yhden sivun. Yhden sivun sivustoiksi tutkimuksessa oli laskettu myös sivuston keskeneräisyydestä ilmoittavat sivut, web-esitesivut (brochureware), parkkeeraussivut (parking pages) ja mainontaan tarkoitettut sivut. Seuraavassa on kuva sivustojen sivumääristä (kuva 2.2). (VeriSign 2008.)



Kuva 2.2 .com ja .net -päätteisten verkkotunnusten sivumäärät. (VeriSign 2008.)

Internetin verkkotunnusten web-sivujen määrä muuttuu joka päivä. Uusien sivujen luomisen lisäksi vanhoja sivuja poistetaan, sivuja linkitetään toisiinsa ja niiden sisältö muuttuu. Internet-tiedonhaussa ongelmaksi muodostuu luotettavan ja oikean tiedon löytäminen sekä haun rajaaminen niin, että sopivaa tietoa saadaan vain tiettyyn aiheeseen liittyen. Ongelmaton tiedonhaku internetissä edellyttää, että internetin käyttäjä tuntee oikeat välineet (eli hakupalvelut) ja hakutavat (eli tiedonhakustrategiat). (Ikonen ym. 2004; Kokkolan Kaupunginkirjasto; OAMK (Tiedonhaku).)

2.1 Internetin hakupalvelut

Tiedon etsimiseen internetistä on olemassa erilaisia hakupalveluita. Hakupalvelut voidaan jakaa seuraaviin ryhmiin: aihehakemistot, organisaatio- tai palvelinhakemistot, erikoishakemistot, tietokannat, portaalit, hakukoneet. (Tampereen kaupungin kirjasto.)

2.1.1 Aihehakemistot

Aiuehakemistot ovat linkkikokoelmia, jotka on järjestetty aiheiden mukaan. Aiuehakemistot lajittelevat tiedon kategorioihin. Tällöin käyttäjät voivat etsiä tietoa selaillemalla läpi heille tarjottuja erilaisia kategorioita. Perinteisiä aihehakemistopalveluja tarjoavat internetissä muun muassa Makupalat ja Yahoo. Yahoo! tarjoaa hakemistojen selailun lisäksi myös mahdollisuuden etsiä aihehakemistoista tietoa hakusanojen avulla. (Tampereen kaupungin kirjasto.)

Seuraavassa kuvassa on esimerkki aihehakemistosta: Makupalat-aiuehakemisto (kuva 2.3).

Vapaahaku [ohje]
 Hae

MAKUPALAT
: HÄMEENLINNAN KAUPUNGINKIRJASTO :

: UUTUUKSIA : HAKUPALVELUT : KÄSIKIRJASTO

Vapaa-aika	Yhteiskunta	Taiteet	Pikalinkit
Matkailu, Maat Ruoka ja juomat Retkeily, Kalastus Autot, Prätäkät... Urheilu Kädentaidot Lemmikkieläimet Muoti Ajanviete, Keräily Huumori Peliarkistot Rajatieto Terveys ja kauneus Verkkokaupat	Politiikka Euroopan unioni Julkishallinto Kaupungit, Kunnat Laki, Oikeus, Poliisi Talous Liikenne, Aikataulut Sosiaalipalvelut Maa- ja metsätalous Maanpuolustus Yritykset, Kauppa Työ, Työnhaku Koulut, Opiskelu Kirjastot Kirkko, Uskonnot, Teologia Tilastot Tietoyhteiskunta Tasa-arvo, Ihmisoikeudet Muuttoliike	Musiikki Kirjallisuus Sarjakuvat Teatteri Tanssi Elokuva Kuvataiteet Arkkitehtuuri Taideteollisuus, Muotoilu Valokuva Multimedia Kulttuuri - yleistä	YLE Uutiset » Sää - Ilmatieteen laitos » TV-ohjelmat » ExpressBus - aikataulut » VR - aikataulut » Häme-Wiki » Suomi.fi » Wikipedia - suomi » Wikipedia - englanti » Kartat ja reitit » Luontoon.fi » Sanojen aika » Otakantaa.fi » Answers.com »
Koti Asunnot, Remontti Puutarhanhoito Lapsi ja perhe Lasten linkit Kodin juhlat	Viestintä Uutiset, lehdet TV, Radio Posti, Puhelin Internet Verkkójulkaiseminen Viestintä, Journalismi	Eri tieteenaloja Historia, Sukututkimus Kansatiede Biologia Lääketiede, Sairaudet Matematiikka Kemia Fysiikka Filosofia Sosiologia Psykologia Kielitiede, Sanakirjat Kasvatustiede Tiede - yleistä	
Ympäristö Maantiede, Kartat Maaperä, Ilmasto ja sää Tähtitiede Luonto, Ympäristö Eläimet Kasvit	Tekniikka Tekniikka, teollisuus Tietotekniikka, atk		

: INFO : POSTIA : MEDIAKORTTI : GOOGLE : KIRJASTOON

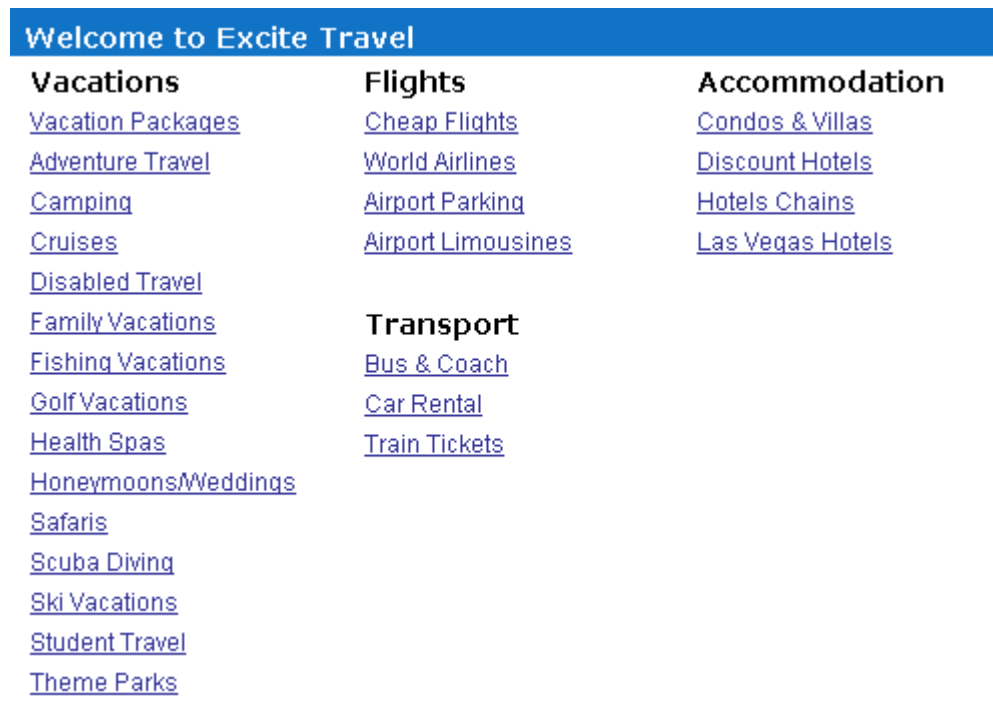
Kuva 2.3 Makupalat-aiuehakemisto.

Kuten yllä olevan kuvan esimerkissä, aihehakemistot kootaan manuaalisesti. Käyttäjät tai dokumenttien laatijat ilmoittavat löytämistään tai tekemistään hyvis- tä dokumenteista hakemiston ylläpitäjälle. Ylläpitäjät päättää lopuksi itse sen, lisätäänkö dokumentti hakemistoon vai ei. (Tampereen kaupungin kirjasto.)

2.1.2 Organisaatio- tai palvelinhakemistot

Organisaatio- tai palvelinhakemistot ovat hyödyllisiä, kun halutaan päästä kä- siksi esimerkiksi jonkun maan palveluihin tai halutaan päästä jonkin organisaa- tion palvelimelle. Tyypillisiä palvelinhakemistoja ovat muun muassa Viestintävi- rasto ja Excite Travel. Viestintäviraston palvelinhakemisto sisältää linkkejä tär- keisiin suomalaisiin yrityksiin ja Excite Travel -palvelinhakemisto puolestaan sisältää tietoa eri maista ja kaupungeista lähinnä turisteille. (Tampereen kau- pungin kirjasto.)

Seuraavassa kuvassa 2.4 on kuva Excite Travel -sivuston palvelinhakemistosta.



Welcome to Excite Travel		
Vacations	Flights	Accommodation
Vacation Packages	Cheap Flights	Condos & Villas
Adventure Travel	World Airlines	Discount Hotels
Camping	Airport Parking	Hotels Chains
Cruises	Airport Limousines	Las Vegas Hotels
Disabled Travel		
Family Vacations	Transport	
Fishing Vacations	Bus & Coach	
Golf Vacations	Car Rental	
Health Spas	Train Tickets	
Honeymoons/Weddings		
Safaris		
Scuba Diving		
Ski Vacations		
Student Travel		
Theme Parks		

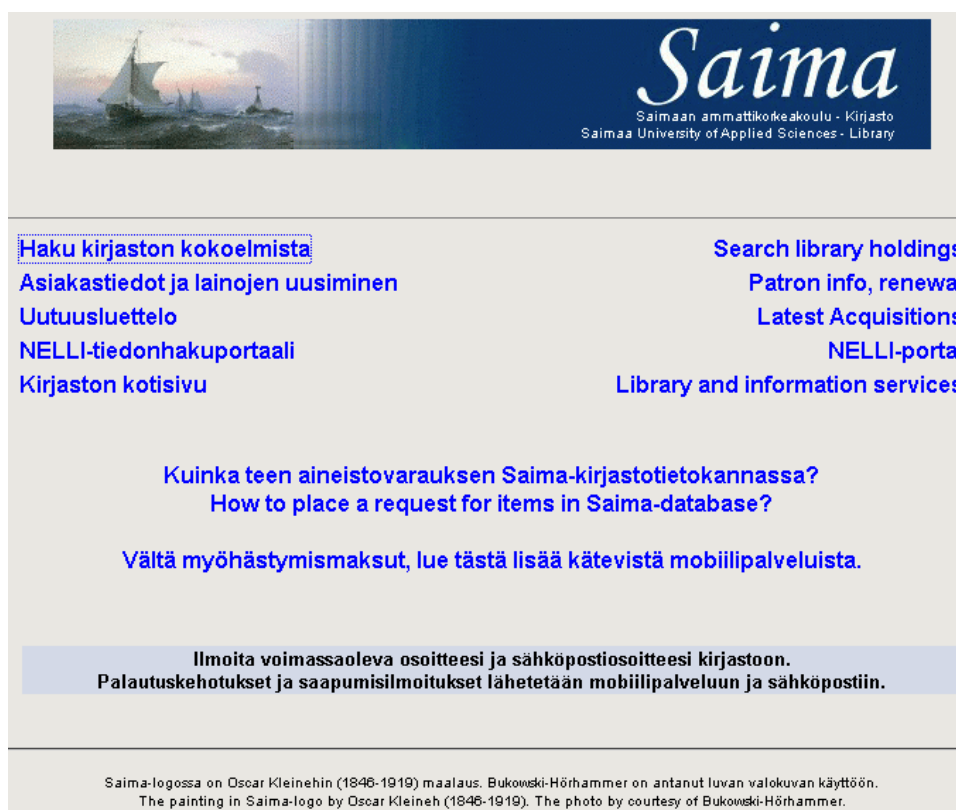
Kuva 2.4 Exite Travel –sivuston palvelinhakemistot.

Kuvan 2.4 sivulla palvelut on luokiteltu ominaisuuksien mukaan. Jokainen linkki johtaa hakemistoon, jossa on lueteltu erilaisia palveluita, kuten esimerkiksi Train Tickets-hakemisto sisältää linkkilistan junalippuja myyvien yritysten kotisivuille.

2.1.3 Erikoishakemistot ja tietokannat

Erikoishakemistoilla ja tietokannoilla (eli palvelukohtaisilla hakuvälineillä) tarkoitetaan joko hakuteoksia, järjesteltyjä tietokantoja verkossa (esimerkiksi kirjaston aineistotietokannat) tai hakupalveluita, jotka on luotu tietynlaisen tiedon (esimerkiksi sähköpostiosoitteiden) etsintään internetistä. Yleensä nämä palvelut ovat joko maksullisia tai vaativat käyttäjäänsä rekisteröitymään (ilmainen) palvelun käyttäjäksi. (Tampereen kaupungin kirjasto.)

Seuraavassa kuvassa 2.5 on Saimaan ammattikorkeakoulun kirjaston Saima-tietokanta.



Haku kirjaston kokoelmista

Asiakastiedot ja lainojen uusiminen

Uutuusluettelo

NELLI-tiedonhakuportaali

Kirjaston kotisivu

Search library holdings

Patron info, renewal

Latest Acquisitions

NELLI-portal

Library and information services

Kuinka teen aineistovarauksen Saima-kirjastotietokannassa?
How to place a request for items in Saima-database?

Vältä myöhästymismaksut, lue tästä lisää kätevästä mobiilipalveluista.

Ilmoita voimassaoleva osoitteesi ja sähköpostiosoitteesi kirjastoon.
Palautuskehotukset ja saapumisilmoitukset lähetetään mobiilipalveluun ja sähköpostiin.

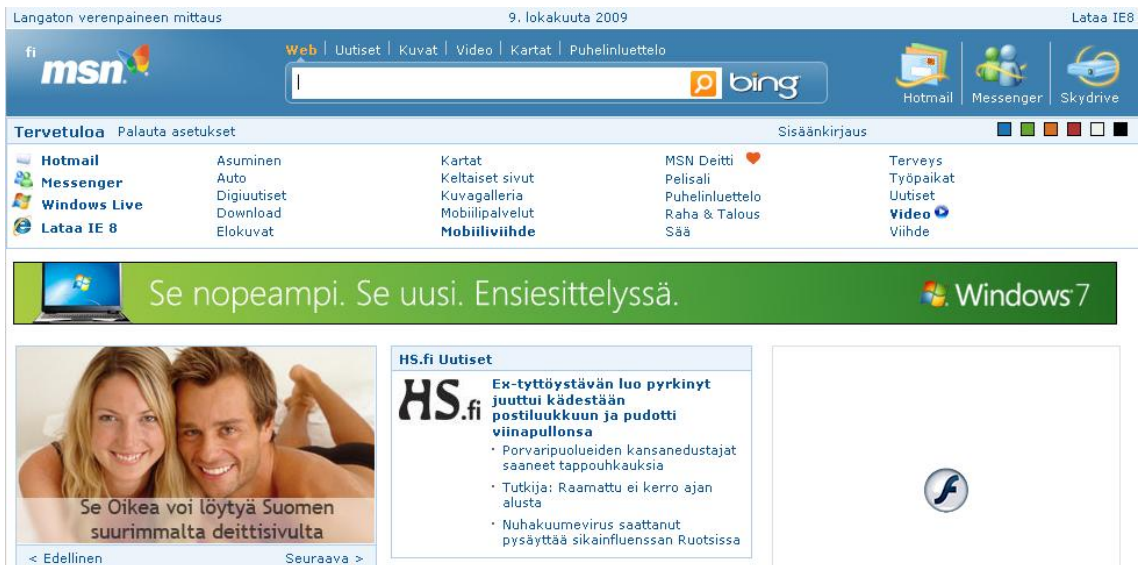
Saima-logossa on Oscar Kleinehin (1846-1919) maalaus. Bukowski-Hörhammer on antanut luvan valokuvan käyttöön.
The painting in Saima-logo by Oscar Kleineh (1846-1919). The photo by courtesy of Bukowski-Hörhammer.

Kuva 2.5 Saima-tietokanta.

2.1.4 Portaalit

Portaali on internetissä oleva aloitussivu, joka tarjoaa käyttäjilleen alkupisteen internetissä navigointiin ja josta voi mennä lukuisiin muihin palveluihin muistamatta niiden osoitetta. Portaaleista käyttäjä voi löytää esimerkiksi uutisia, sää-tietoja, TV:n ohjelmatietoja ja -esittelyitä, linkkikokoelmia ja hakupalveluita. Tunnettuja ja yleisiä portaaleja ovat muun muassa Suomi24, Plaza ja MSN. (OAMK (Tiedonhaku).)

Seuraavassa kuvassa 2.6 on MSN-portaali.



Kuva 2.6 MSN-portaali.

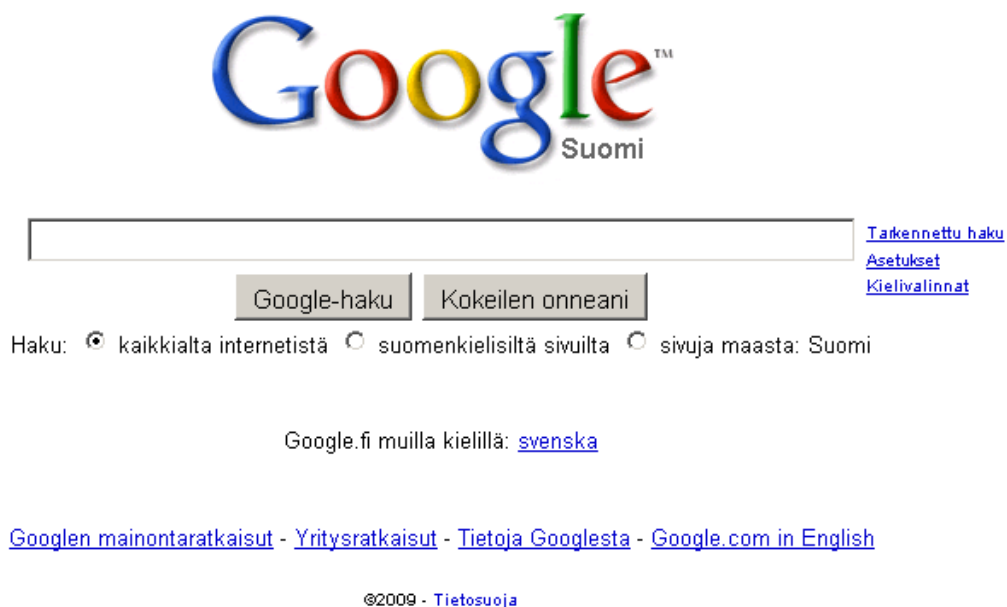
2.1.5 Hakukoneet

Hakukoneet ovat käyttäjien keskuudessa suosittuja. Hakukoneiden avulla voidaan yhdellä tai useammalla hakusanalla etsiä hakusanaan liittyvää tietoa verkosta. Hakukoneet keräävät automaattisesti hakubottien avulla web-sivuja verkosta, käyvät keräämänsä web-sivut eli dokumentit läpi ja lisäävät niitä suuren tietokantaan. Käyttäjät voivat sitten eri tavoilla tehdä hakuja hakukoneen tietokantaan ja saada vastaukseksi linkkiviitteitä. (Puskala 2002, Tampereen kaupungin kirjasto, Wikipedia Internet search engines (englanti) 2009.)

Hakukoneen hakutulokset esitetään yleensä viitelistana. Listan sisältö koostuu viitteistä web-sivuille, kuviin ja muun tyyppisiin tiedostoihin. Hakukoneiden listat eivät esitä viitelistojen viitteitä sekalaisessa järjestyksessä vaan listat ovat yleensä laitettu viitteiden paremmuusjärjestykseen tiedonhakijan käyttämien hakusanojen mukaan. Yleensä myös mainokset ja sponsoreiden linkit asetetaan tiedonhakijan hakusanasta riippuen listalla ensimmäisiksi. (Tampereen kaupungin kirjasto, Wikipedia Internet search engines (englanti) 2009.)

Googlea ja Altavistaa voidaan pitää niin sanottuina perinteisinä hakukoneina. Niillä tehtävät haut perustuvat avainsanapohjaiseen hakuun. Siinä web-sivujen sisältö ensin analysoidaan ja tämän jälkeen indeksoidaan. Indeksoinnilla pyritään löytämään sopivia termejä, jotka kuvaavat sivua parhaiten. Kun käyttäjä tekee haun tietyllä hakusanalla, hakukone etsii kyseistä sanaa indeksoiduista sanoista ja palauttaa sitten yleensä suuren määrän sivuja, joihin tämä sana on indeksoitu. (Tupala 2005.)

Seuraavassa kuvassa 2.7 on Google-hakukone.



Kuva 2.7 Google-hakukone.

2.2 Tiedonhakustrategiat

Internetin web-sivut eivät kaikki vielä käytä yhteistä standardia, jolla voidaan luokitella sivustoja esimerkiksi avainsanojen mukaan helpottamaan tiedon etsintää. Yhdellä hakusanalla löytyy useita käyttökeltvottomia ja käyttökelpoisia linkkejä, jotka on järjestelty hakusanan todennäköisen tarkoituksperän mukaan. Tällöin hakutuloslistalta hakusanalla "apple" löytyy tietoa muun muassa yrityksestä ja hedelmästä vuoronperään. Jotta voidaan löytää hakusanaan paremmin liittyviä linkkejä, on laadittava tiedonhakustrategia. (OAMK (Tiedonhaku), Nikkilä 2001.)

Tiedonhakustrategiaa laadittaessa on pohdittava seuraavia asioita:

1. Mistä tieto voisi löytyä? Kuka on todennäköisin tiedontuottaja?
2. Löytyykö mahdollisen tiedontuottajan sivuilta linkkejä muille web-sivuille, jotka sisältävät tietoa asiasta?
3. Etsi tiedon aihepiiriin sopivia aihehakemistoja
4. Jos edellä mainituista palveluista ei ollut apua on turvauduttava hakukoneisiin, jonne voidaan syöttää hakusanan lisäksi lukuisia aiheeseen ja hakusanaan liittyviä avain sanoja (esimerkiksi "apple fruit" tai "apple company"). Hakusana voidaan myös korvata sen synonyymillä ja rajata hakua kirjoittamalla synonyymiin ja hakusanaan liittyviä termejä.
5. Hakukoneiden asetuksia muuttelemalla voidaan rajata hakutuloslista koostumaan vain esimerkiksi Suomalaisista tai suomenkielisistä hakusanoista.
6. Hakukoneita on useita, joten kannattaa käyttää useampaa kuin yhtä. (OAMK (Tiedonhaku), Nikkilä 2001.)

Useasti myös nopea tapa löytää tietoa on yrittää kirjoittaa dokumentin osoite (esimerkiksi www.apple.com) osoiteriville selaimessa tai hakukoneen hakukenttään. Hakukoneilla voidaan myös etsiä muita hakupalveluita, kuten ainehakemistoja tai tietokantoja, joista voidaan etsiä tietoa paremmin. Nykyään kannattaa käyttää myös internetissä olevia virtuaalisia tietosanakirjoja asiatiedon etsimiseen. Näistä tunnetuin on Wikipedia. (OAMK (Tiedonhaku), Nikkilä 2001.)

2.3 Internetin tiedonhakuun liittyviä ongelmia

Internetiä ei ole alunperin rakennettu ja organisoitu järjestelmällistä tiedonhakua varten, vaan se on nimenomaan vapaa ja hajautettu linkkien järjestelmä. Tämän vuoksi tarvitaan tiedon hakemiseen edellä jo mainittujen tiedonhakustrategioita ja välineitä. Nämäkään välineet ja tiedonhakustrategiat eivät ole täydellisiä ja oikean tiedon löytämiseksi internetistä tarvitaan jonkun verran aikaa ja vaivannäköä. Tämän lisäksi nykyiset perinteiset hakukoneet eivät ymmärrä, mitä sanat tarkoittavat, ja jos sanalla on lukuisia homonyymejä, hakutulostista (viitelista) sisältää vuorotellen linkkejä liittyen hakusanan homonyymeihin. Esimerkiksi hakusanalla ”apple” voisi ensimmäisenä viitteenä olla linkki sivulle, jolla kerrotaan hedelmistä, ja toinen viite veisi Apple-nimisen yrityksen sivulle. Kolmantena olisi puolestaan taas linkki sivulle, jossa kerrottaisiin hedelmistä. (OAMK (Tiedonhaku).)

Edellä mainittujen tiedon hakuun liittyvien ongelmien lisäksi on myös otettava huomioon tiedon luotettavuus. Internetistä löydetty tieto voi olla muun muassa puutteellista tai vanhentunutta, koska sitä ei valvota. Nykyään tosin muun muassa poliisi pystyy kieltämään joillekin sivuille menemisen, jos sivuilla julkaistaan laitonta materiaalia levitykseen. (OAMK (Tiedonhaku).)

2.4 Ongelmista ratkaisuihin

Kuten edellisissä kappaleissakin on jo mainittu: tiedon haku internetistä on perinteisillä hakupalveluilla monimutkaista ja ongelmallista. Tämän takia onkin alettu panostaa viime aikoina yhä enemmän hakukoneiden toimintaan lisäämällä niihin semantiikkaa ja automatiikkaa. Semantiikan ja automatiikan avulla on mahdollista lisätä helppokäyttöisyyttä ja nopeuttaa hakukoneita. Semantiikkaa hyödyntäviä hakukoneita kutsutaan semanttisiksi hakukoneiksi.

Semanttinen hakukone eroaa perinteisestä hakukoneesta siinä, että se ratkaisee semantiikan avulla paremmin sen, mitä käyttäjä yrittää etsiä hakukoneella. Jos perinteisellä hakukoneella suorittaa haun häiden järjestelemisestä tietyssä kaupungissa, tarjoaa hakukone linkkejä sivustoille, jotka mahdollisesti sisältävät häitä järjestävien yritysten tietoja kyseisellä alueella. Semanttisella hakukoneella tehty haku antaa myös tietoja sillä alueella häitä järjestävistä yrityksistä, mutta tarjoaa tämän lisäksi muita häätapahtumaan liittyviä asioita, kuten esimerkiksi eri pitopaikkojen tietoja tai vaikkapa erikoisia hääautoja vuokraavien yritysten sivustoja. Semanttisuuden avulla hakukone pystyy luomaan asiayhteyksiä ja näin tarjoamaan hakijalle oikeita vastauksia oikeisiin hakusanoihin liittyen. (Tupala 2005.)

Lisättäessä hakukoneeseen semantiikkaa voidaan myös ratkoa edellisen luvun homonymiongelmia ja parantaa hakutulosten laatua. Semanttiset hakukoneet osaavat päätellä, mitä esimerkiksi hakusana "apple" tarkoittaa missäkin asiayhteydessä eikä hakutulostista sisällä enää vuoroperään hakutuloksia omenista ja yrityksestä nimeltä Apple.

3 SEMANTIikka

Edellisessä luvussa käsiteltiin jo alustavasti, mitä semantiikka tuo hakukoneisiin. Luvussa ei kuitenkaan perehdytty siihen, mitä semantiikka on. Tämän luvun tarkoituksena on tutustua semantiikkaan käsitteenä ja sen käyttöön tietoteknisissä laitteissa ja ohjelmissa. Lopuksi tässä luvussa myös perehdytään tarkemmin myös semanttisiin hakukoneisiin ja niiden toimintaan.

3.1 Semantiikka käsitteenä

Semantiikka eli merkitysoppi on kielitieteen osa-alue, joka keskittyy merkitysten tutkimiseen. Tutkittavana voivat olla sanojen, kieliopillisten ainesten, lauseiden, tekstien ja myös ilmauksen merkitykset. Merkityksen tutkimisessa oleellisia ovat ilmaisujen väliset suhteet, kuten antonymia, homonymia, hyponymia, meronymia, polysemia ja synonymia. (Muikku-Werner & Savolainen 1999; WikiWiki 2005; Wikipedia (suomi) semantiikka 2009.)

Ilmausten tulkinnassa tulkitsija vaikuttaa oleellisesti siihen, miten ilmaus ymmärretään. Ilmauksen tulkitsijan ollessa ihminen ilmauksen merkitys on se, mitä ihminen ymmärtää ilmauksesta. Ihmisen ymmärtämiseen vaikuttavat muun muassa kulttuuri, ympäristö ja normit. Tietokoneen ollessa tulkitsijana semantiikka ymmärretään kapeammin (koska koneet eivät ymmärrä mitään): ohjelmoinnissa ohjelmointikielen ilmauksen semantiikka on määritys siitä, mitä kone tekee, kun sille annetaan kyseinen ilmaus. (Muikku-Werner & Savolainen 1999, WikiWiki 2005)

Esimerkiksi Java-kielen ilmauksen (tai lausekkeen) `3 + 5` semantiikka on se, että kone laskee yhteen luvut 3 ja 5. Java-kielen lausekkeen `System.out.println(x)`, missä `x` on mikä tahansa lauseke, semantiikka on se,

että ohjelma tulostaa käyttäjälle lausekkeen x semanttisen arvon. (WikiWiki 2005.)

3.2 Semantiikka tietoteknisissä sovelluksissa

Kun semantiikka lisätään tietokoneohjelmaan (esimerkiksi semanttiset hakukoneet) tai laitteeseen (esimerkiksi robotti), niin saadaan lopputulokseksi ohjelma tai laite, joka ymmärtää tiettyjä sanojen merkityksiä ja niiden välisiä asiayhteyksiä. Tällaiset laitteet tai ohjelmat yleensä on rajattu ymmärtämään vain tietyn sanaston sanoja (esimerkiksi lääkäreiden ammattisanasto). Tietotekniset laitteet ja ohjelmat eivät kuitenkaan ymmärrä sanojen merkitystä ja asiayhteyksiä samalla tavalla kuin ihminen ymmärtää lukemalla sanakirjasta. Tämän takia ne on opetettava ymmärtämään sanojen merkityksiä käyttämällä ontologioita ja semanttisten verkkojen kaltaisia tietomalleja.

Seuraavissa luvuissa käsitellään sitä, miten tietoteknisille laitteille ja sovelluksille voidaan opettaa ymmärtämään sanoja tutustumalla ontologiaan ja semanttisiin verkkoihin.

3.2.1 Ontologia

Tietokoneelle pelkät sanat ovat vain symboleja vaikka ihmiselle tietyllä sanalla on joku tietty merkitys. Jotta hakusanalle voidaan lisätä semanttisessa hakukoneessa jokin merkitys, on turvauduttava ontologiaan. Tietojenkäsittelyn yhteydessä ontologia tarkoittaa luokittelua ja kategorisointia. Ontologioilla voidaan laatia tietyn aihealueen käsitteistöjä, joka usein muodostetaan olemassa olevien sanastojen avulla. Ontologioiden avulla pystytään kuvaamaan eri käsitteitä ja niiden välisiä suhteita virallisesti, jolloin tietokone pystyy käsittelemään niitä. Ontologian peruskäsitteitä ovat muun muassa luokka, yksilö, ominaisuus ja omaisuus. Peruskäsitteistä luokat voidaan vielä jakaa ali- ja yliluokkiin, jolloin voidaan myös luokitella käsitteitä. (Tupala 2005, Wikipedia (suomi) luokittelujärjestelmät 2009.)

Teknisesti tarkasteltuna ontologiaan kuuluu yhteinen sanasto, joka kuvaa olioita ja niiden välisiä suhteita määrämuotoisella tavalla, sekä kielioppi, jonka avulla sanastoa käyttäen voidaan kuvata reaalimaailman eri aihepiirejä. (Profium.)

Internetin semanttinen web (semantic web) käyttää ontologioita olennaisena osana, joiden avulla tietokoneohjelmat (robotit, botit ja hakukoneet) pystyvät tehokkaammin paikantamaan tai tunnistamaan oikeaan ryhmään kuuluvaa sisältöä. (Wikipedia (suomi) luokittelujärjestelmät 2009.)

3.2.2 Semanttinen verkko

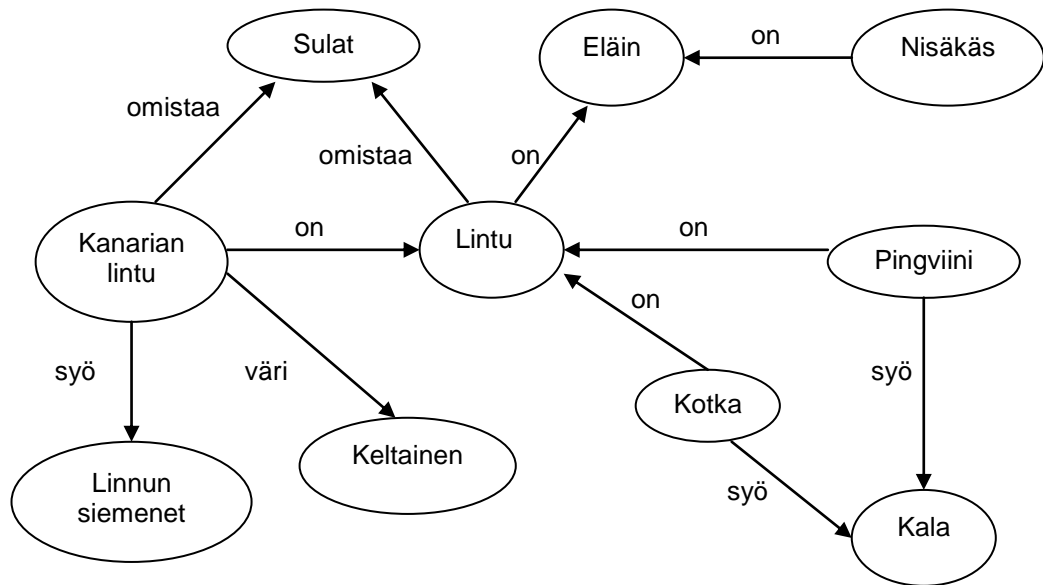
Semanttinen verkko (Semantic network) on graafinen notaatio tiedon esittämiselle (eli tietomalli). Rakenteena semanttisella verkolla tarkoitetaan suunnattua graafia, jonka solmut edustavat käsitteitä (ontologioita) ja kaaret käsitteiden välisiä semanttisia suhteita. Semanttisia verkkoja käytetään esimerkiksi sanastojen esittämiseen tietokoneella käsiteltävässä muodossa. Niiden avulla tietotekniset ohjelmat ja laitteet oppivat ymmärtämään, mitä sanat tarkoittavat missäkin asiayhteydessä. (Wikipedia (suomi) tekoäly 2009, Sowa 2006.)

Semanttisissa verkoissa käytettyjä suhdetyyppejä ovat:

- synonyymisuhteet (samaa tarkoittavat sanat)
- antonyymisuhteet (vastakohtat)
- meronyymi- ja holonyymisuhteet (osakokonaisuus)
- hyponyymisuhteet (alakäsitteet)
- hyperonyymisuhteet (yläkäsitteet) (Obitiko 2007.)

Näiden yllä olevien suhteiden lisäksi semanttisen verkon käsitteisiin voi liittyä myös sitä kuvaavia ja tekemiseen liittyviä käsitteitä. Esimerkiksi kuvassa 3.1 on kanarialinnulla muitakin suhteita kuin pelkästään ylä- ja alakäsitteisiin, osakokonaisuus käsitteisiin ja synonyymeihin liittyviä suhteita. Kanarialintu on keltainen ja sille voidaan syöttää siemeniä. Kuvassa väriin liittyvä suhteen nimeksi on an-

nettu väri. Näiden attribuuttien lisääminen verkkoon tekee siitä rikkaamman ja monipuolisemman. Seuraavassa kuva semanttisesta verkosta. (Yaratan 2003.)



Kuva 3.1 Semanttinen verkko. (Yaratan 2003.)

Suhdetyypit ja niiden lisääminen yllä olevaan kaavioon tekeekin perinteisestä hierarkkisesta kaaviosta moniulotteisen. Moniulotteisuutensa ansiosta muun muassa hakukoneet voivat suoriutua seuraavista tehtävistä:

- voidaan vastata suoriin kysymyksiin
- voidaan vertailla käsitteitä
- ja voidaan tehdä erilaisia päätelmiä käsitteistä. (Yaratan 2003.)

3.3 Semanttiset hakukoneet

Semanttinen hakukone eroaa tavallisesta siinä, että se antaa ensin haetulle sanalle merkityksen, jota se sitten käyttää haun suorittamiseen. Siinä missä perinteinen hakukone etsii internet-sivuista vain tiettyyn sanaan liittyviä tuloksia, semanttinen hakukone pystyy hyödyntämään ontologiaa ja etsimään sanaan liittyviä käsitteitä ja näiden avulla hakemaan tarkempia tuloksia tietylle hakusanalle. (Tupala 2005.)

Semanttisuus mahdollistaa myös sen, että eri sivustojen sisältö ja riippuvuudet toisista sivustoista pystytään kuvaamaan ja löytämään helpommin. Jos tavallisella hakukoneella suorittaa haun juhlien järjestelemisestä tietyssä kaupungissa, tarjoaa hakukone linkkejä sivustoille, jotka mahdollisesti sisältävät juhlia järjestävien yritysten tietoja kyseisellä alueella. Semanttisella hakukoneella tehty haku antaa myös tietoja sillä alueella juhlia järjestävistä yrityksistä, mutta tarjoaa myös muita juhliin liittyviä asioita, kuten esimerkiksi eri pitopaikkojen tietoja. Semanttisuuden avulla hakukone pystyy luomaan asiayhteyksiä ja näin tarjoamaan hakijalle oikeita tuloksia oikeisiin hakusanoihin liittyen. (Tupala 2005.)

Nykyisiä semanttisia hakukoneita ovat muun muassa Bing ja Haku. Myös Googlesta on yritetty tehdä semanttista hakukonetta, mutta yritykset eivät ole vielä ainakaan lyöneet läpi käyttäjien keskuudessa. Googlea voidaan kuitenkin mukauttaa niin, että siitä saa semanttisen hakukoneen. Googlelle löytyy runsaasti lisäosia, jotka tuovat semantiikkaa hakukoneeseen. Nimekkäin näistä lisäosista on Smartlogic, Nasan ja Googlen yhteistyössä tekemä Semaphore. (Smartlogic 2009.)

Semaphore-lisäosa on välitason ohjelma, joka on integroitu Googlen hakupalvelun osaksi. Se sijoittuu Google-hakukoneessa käyttöliittymän ja tieto- sekä hakulähteiden väliin. Lisäosan tehtävänä on lisätä merkitys ja konteksti käyttäjäkemukseen sekä lisätä luokittelun metatieto hakuindeksiin ja/tai tietolähteisiin. (Smartlogic 2009.)

Semaphore käyttää semanttisia malleja taksonomioiden ja ontologioiden muodossa ohjatakseen ja kuvastaakseen käyttäjän näkökulmaa subjektista. Semaphore kutsutaankin ontologiakeskeiseksi lisäosaksi. (Smartlogic 2009.)

4 OHJELMISTOPROJEKTIN HALLINTA

Sinseweb-projektin tekemiseen on jokaisessa vaiheessa hyödynnetty erilaisia menetelmiä ja tekniikoita. Tekniikat ja menetelmät on valittu sen mukaan, miten ne sopivat vaiheisiin. Tässä luvussa käydään läpi, mitä tekniikoita ja menetelmiä työssä on käytetty sen eri vaiheissa.

4.1 Vaihejakomallit ja elämänkaari

Ohjelmiston elämänkaarella tarkoitetaan aikaa, joka kuluu ohjelmiston kehittämisen aloittamisesta sen poistamiseen käytöstä. Vaihejakomallilla tarkoitetaan tapaa, jolla ohjelmiston kehitystyö tai koko elämänkaari jaetaan vaiheisiin. Seuraavassa on esitelty muutamia vaihejakomalleja, joita on käytetty tässä projektissa. (Haikala & Märijärvi 2004.)

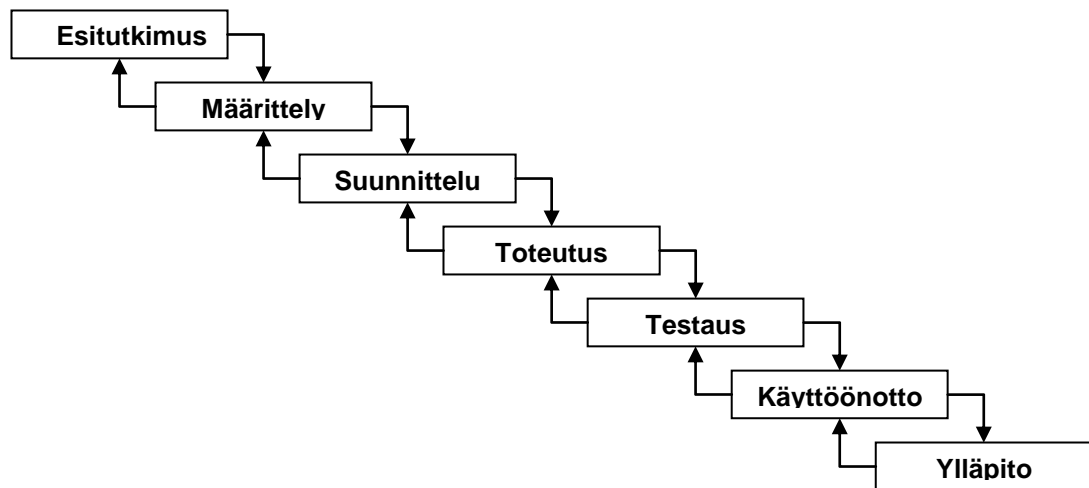
4.1.1 Vesiputousmalli

Vesiputousmalli on 1960-luvun lopulla perinteisten fyysisten prosessimallien pohjalta kehitetty vaihejakomalli. Vesiputousmallissa ohjelmistojen kehittäminen nähdään eteenpäin kulkevana prosessina, jossa on hankalaa tai jopa turhaa peruuttaa taaksepäin. Vesiputousmallissa kehittämisen vaiheet seuraavat suoraviivaisesti toisiaan alkaen esitutkimuksesta ja päättyen ylläpitoon. (Pohjonen 2002.)

Vesiputousmalli ei ole täysin ongelmaton, ja voidaankin sanoa, että vesiputousmalli kuvaa ohjelmistoprosessille tyypillisen iteratiivisuuden huonosti. Esi-merkkinä mallin ongelmallisuudesta on se, että usein tietyn vaiheen suoritus paljastaa edellisessä vaiheessa tehtyjä virheitä ja syntyneitä ongelmia, jolloin prosessissa on peruutettava taaksepäin ja korjattava virheet. Toinen mallin on-

gelma on se, että varsinaisia tuloksia voidaan esittää asiakkaalle vasta myöhäisessä prosessin vaiheessa. (Pohjonen 2002.)

Ongelmistaan huolimatta vesiputousmalli on kuitenkin tunnetuin ja edelleen käytännössä yleisimmin sovellettu vaihejakomalli. (Pohjonen 2002.)



Kuva 4.1 Vesiputousmalli. (Pohjonen 2002.)

4.1.2 Prototyypimalli

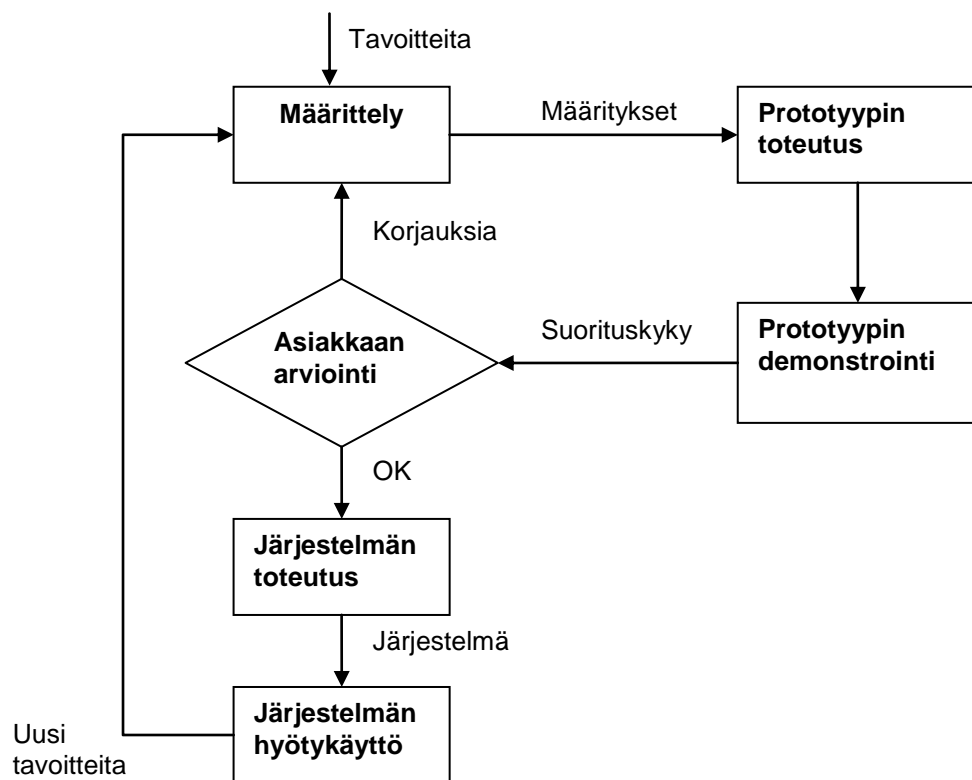
Toisin kuin vesiputousmalli, prototyyppi malli mahdollistaa sen, että asiakas pääsee nopeammin näkemään ohjelmistoprojektin tuloksia ja kokeilemaan mahdollisesti myös projektissa kehittyvää tuotetta. Prototyypimallissa ohjelmistonkehitys on prosessi. Se alkaa vaatimusten ja tavoitteiden määrittelystä. Tämän jälkeen siirrytään tuotteen toteuttamiseen. Tavoitteita ja vaatimuksia vastaavan tuotteen prototyypin ollessa valmis tuote viedään asiakkaalle arvioitavaksi. Asiakas arvioi tuotteen prototyypin, jota kehitetään eteenpäin asiakkaalta saadun palautteen mukaisesti. (Pohjonen 2002.)

Prototyypin kehitysprosessi, joka edellä jo kuvattiin, jatkuu niin kauan, että asiakas on tyytyväinen prototyyppiin. Tämän jälkeen lopputuote (ts. järjestelmä, ohjelmisto) toteutetaan prototyypin pohjalta käyttäen hyväksi prototyypin karkeita

määrittämiä, sen toteutuksessa saatuja kokemuksia sekä rakenteellisia ja toiminnallisia ratkaisuja. (Pohjonen 2002.)

Prototyypimalli ei ole myöskään täysin ongelmaton. Sen ongelmana muun muassa on mallin käyttämien resurssien suuruus. Tämän lisäksi prototyypimalli ei välttämättä tuo esille kaikkia järjestelmän piileviä ongelmia. Ongelmiensa takia prototyypimalli ei välttämättä sovellu järjestelmien toiminnallisuuden kehittämiseen. Tämän sijaan sitä suositellaan käytettäväksi muun muassa käyttöliittymän määrittelyn yhteydessä. (Pohjonen 2002.)

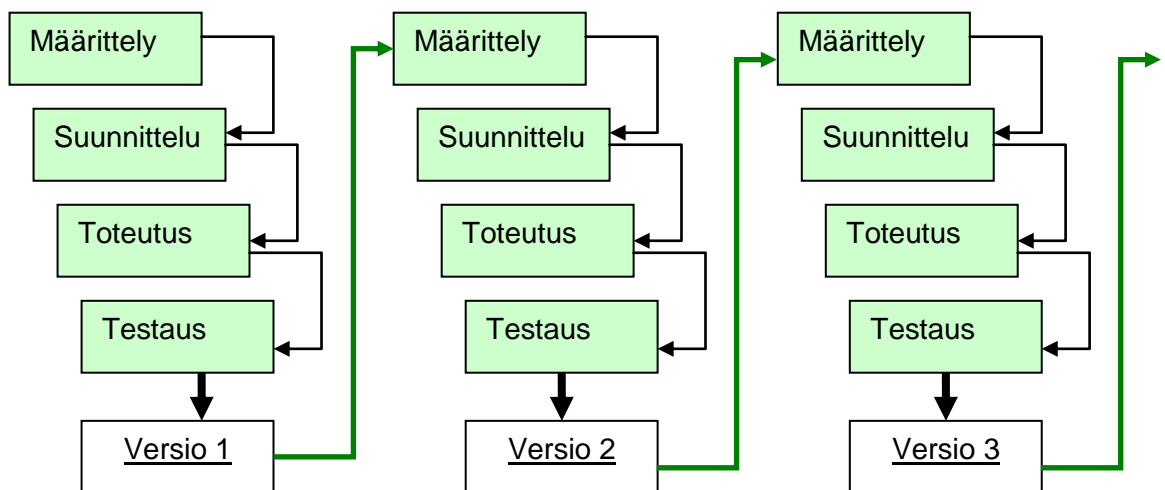
Seuraavassa kuvassa 4.2 on kuvattu prototyypimallia vuokaaviolla.



Kuva 4.2 Prototyypimalli. (Pohjonen 2002.)

4.1.3 EVO-malli

EVO-malli (evolutionary delivery) on inkrementaalinen malli. Inkrementaalisilla malleilla tarkoitetaan yleensä ohjelmistokehitystä, jossa projektin sisällä lopputuotetta kehitetään pieninä inkrementteinä. Tällöin projekteissa, joissa käytetään inkrementaalisia malleja, lopputulos valmistuu esimerkiksi kolmella inkrementtikierroksella. EVO-mallissa jokaisella inkrementtikierroksella syntyy toimiva versio järjestelmästä. Edellistä kierrosta seuraavalla inkrementtikierroksella syntyvällä järjestelmän versiolla on aina enemmän ominaisuuksia kuin edeltäjällään. EVO-mallissa kukin inkrementtikierros kestää yleensä muutamasta viikosta muutamaan kuukauteen. Seuraavassa kuvassa on esitetty EVO-malli. (Haikala & Märijärvi 2004.)

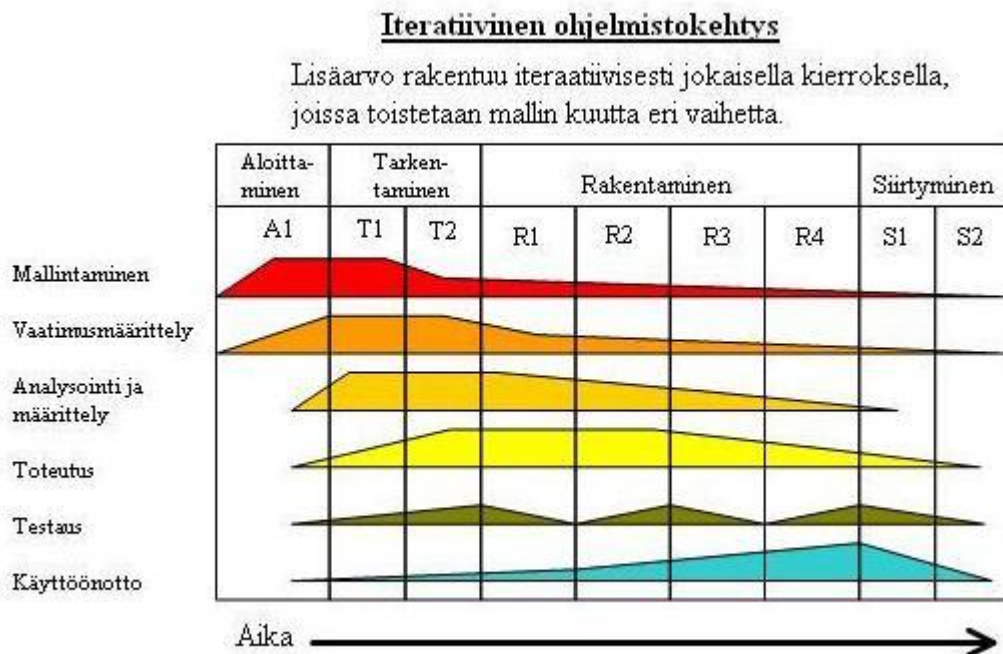


Kuva 4.3 EVO-malli. (Haikala & Märijärvi 2004.)

EVO-malli ei ole prototyyppi- ja vesiputousmallin tapaan täysin ongelmaton malli. EVO-malli voi aiheuttaa muun muassa projektinhallinnallisia ongelmia. Jokainen EVO-mallin inkrementtikierroksella syntynyt järjestelmän versio voidaan luovuttaa asiakkaan käyttöön. Jos asiakas löytää versiosta vakavia virheitä, voi projektiryhmän aika kulua ongelmien ja virheiden korjailuun. Tällöin seuraavan inkrementin kehittäminen voi pysähtyä kokonaan. Myös projektin jakaminen liian pieniin inkrementteihin voi aiheuttaa kokonaisarkkitehtuurin rappautumista ja ohjelman sirpaloitumiseen. (Haikala & Märijärvi 2004.)

4.1.4 RUP

RUP (Rational Unified Process) on prosessi, jota voidaan pitää spiraalimallin ja EVO-mallin sovelluksena. Sen on kehittänyt Ivar Jacobson Rational-yhtiössä. EVO-mallin tapaan RUP perustuu sykleihin, joista jokainen pitää sisällään vesiputousmallin tärkeimmät vaiheet. Tämän lisäksi RUP-prosessissa ohjelmistokehitys jaetaan neljään päävaiheeseen: aloittaminen, tarkentaminen, rakentaminen ja siirtyminen. Kukin näistä vaiheista koostuu yhdestä tai useammasta iteraatiosta. Vaiheiden lopputuloksena syntyy vaiheesta riippuen erilaisia dokumentteja ja lopuksi valmis järjestelmä. Seuraavassa kuvassa 4.4 on kuvattu RUP-prosessia. (Haikala & Märijärvi 2004; Wikipedia (suomi) Ohjelmistotuotanto 2009b.)



Kuva 4.4 RUP. (Wikipedia (suomi) Ohjelmistotuotanto 2009b.)

Kuten kuvasta 4.4 voidaan huomata aloitusvaiheessa keskitytään tuotekonseptin mallintamiseen ja asiakasvaatimusten analysointiin. Tarkentamisvaiheessa keskitytään lähinnä tuotteen vaatimusmäärittelyn laatimisen lisäksi myös perusrakenteiden toteutukseen. Rakennusvaiheessa toteutetaan kuvan 4.3 mukaisesti jokaisen iteraation lopputuloksena uusi beta-versio järjestelmästä. Versiot

annetaan asiakkaan käyttöön niiden valmistuttua. Viimeinen vaihe on Siirtymävaihe. Siirtymävaiheen (siirtyminen) lopputuloksena syntyy patentointikelpoinen järjestelmä ja siihen liittyvät käyttöohjeet, ylläpitopalvelut, asennuspaketit, lisenssisopimukset ja muut vastaavat dokumentit. (Haikala & Märijärvi 2004.)

Siinä missä vesiputousmalli, prototyypimalli ja EVO-malli, ei ole RUP-prosessikaan säästynyt ongelmilta. RUP-prosessi omistaa kaikki EVO-mallin ongelmat, kuten muutkin EVO-mallin kaltaiset ratkaisut. RUP-prosessia pidetään myös hyvin raskaana, koska prosessin aikana on toteutettava useita kymmeniä erilaisia kaavioita ja dokumentteja. (Haikala & Märijärvi 2004.)

4.2 Ohjelmistokehityksen vaiheet

4.2.1 Esitutkimus

Ohjelmistokehitys alkaa, kun ideakynnys järjestelmän tai ohjelmiston kehittämisen suhteen on ylitetty. Ensimmäinen ohjelmistokehityksen vaihe on esitutkimus. Esitutkimuksen tarkoituksena on selvittää edellytykset hankkeen toteuttamiselle. Esitutkimuksen tehtävänä on muun muassa asiakasvaatimusten määrittäminen, hankkeen vaatimien resurssien määrittäminen (esimerkiksi kustannukset, aika ja työkalut) sekä hankkeen hyötyjen ja riskien määrittäminen. (Haikala & Märijärvi 2004, Pohjonen 2002.)

Esitutkimuksessa ei toteuteta vielä mitään teknistä ratkaisua ongelmiin, joita asiakkaalla on. Esitutkimus on tarkoitettu päättäjille. Sen perusteella on pystyttävä tekemään päätös hankkeen toteuttamisesta tai toteuttamatta jättämisestä. Tämän lisäksi esitutkimus toimii perustana hankkeen toteutuksen suunnittelulle. (Pohjonen 2002.)

4.2.2 Määrittely

Elämänkaarimallistakin tutuksi tulleessa määrittelyvaiheessa kootaan ja analysoidaan asiakkaan vaatimuksia ja niiden pohjalta laaditaan ohjelmistovaatimukset, jotka määrittelevät toteutettavan järjestelmän. Määrittelyvaiheessa syntyneitä dokumentteja kutsutaan toiminnalliseksi määrittelyksi (eli määrittelydokumentiksi). (Haikala & Märijärvi 2004.)

Toiminnallisessa määrittelyssä kuvataan ohjelmiston toteutukselle asetettavat ei-toiminnalliset ja toiminnalliset vaatimukset sekä kaikki toteutusta koskevat rajoitukset. Toiminnallisten vaatimusten (eli toimintojen) määrittelyn yhteydessä määritellään myös muut ohjelmiston toteutettavat ominaisuudet, käyttöliittymä ja vuorovaikutus muiden järjestelmien kanssa. Ei-toiminnallisia vaatimuksia ovat muun muassa suoritusteho, vasteaika ja käytettävyys. Rajoituksia ovat käytävissä oleva muistitila, toteutus tietyllä ohjelmointikielellä ja niin edelleen. Lyhyesti sanottuna toiminnalliseen määrittelyyn dokumentoidaan vaatimukset ja vaatimukset täyttävän järjestelmän kuvaus. (Haikala & Märijärvi 2004.)

4.2.3 Suunnittelu

Suunnitteluvaiheessa suunnitellaan määrittelyvaiheessa kuvattujen toimintojen toteutus. Suunnitteluvaihe jaetaan yleensä kahteen vaiheeseen: arkkitehtuurisuunnitteluun ja moduulisuunnitteluun. Näistä vaiheista arkkitehtuurisuunnittelun tuloksena syntyy tekninen määrittely. Teknisessä määrittelyssä ei enää tehdä valintoja tai päätöksiä siitä, millaisia ominaisuuksia järjestelmässä on. Teknisen määrittelyn tarkoituksena on kuvata järjestelmän tekninen arkkitehtuuri mahdollisimman tarkasti. Tekninen määrittely ei kuitenkaan sisällä ohjelmakoodia vaan se kuvaa järjestelmää muun muassa UML-kaavioiden avulla. Syötteenä tekniselle määrittelylle toimii toiminnallinen määrittely, ja se kuvaa ne järjestelmän komponentit, jotka toteuttavat määrittelyn vaatimat toiminnot. (Haikala & Märijärvi 2004; Wikipedia (suomi) Ohjelmistotuotanto 2009a.)

Teknisessä määrittelyssä määritellään esimerkiksi käytetty ohjelmointikieli, järjestelmän komponentit kuten kirjastot, ohjelmistokomponenttien rakenne ja keskinäinen hierarkia, käytetyt tietorakenteet ja niiden väliset sovellusrajapinnat. (Wikipedia (suomi) Ohjelmistotuotanto 2009a.)

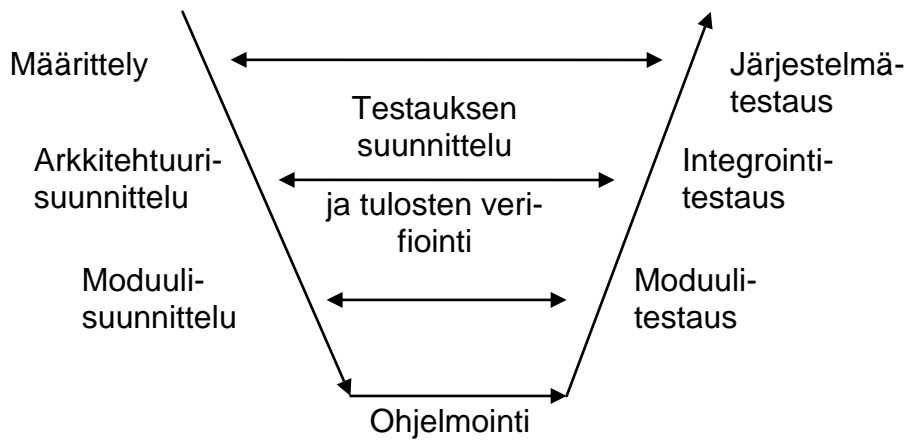
4.2.4 Toteutus ja testaus

Suunnitteluvaiheen jälkeen ohjelmistotuotannossa on toteutusvaihe. Toteutusvaiheeksi kutsutaan ohjelmankirjoitusvaihetta ensimmäiseen virheettömään käännökseen saakka. Toteutusvaiheessa toteutetaan edellisissä vaiheissa määritelty ja suunniteltu järjestelmä vaiheissa syntyneiden dokumenttien mukaisesti. Toteutusvaiheen lopputulos on yleensä virheetön käännös ohjelmistosta. (Haikala & Märijärvi 2004.)

Toteutusvaiheeseen jälkeen tulee testausvaihe. Testausvaiheen tarkoituksena on etsiä virheitä ohjelmistosta suunnitelmallisesti ja tavoitteena todistaa ohjelman toimivuus. Kaikki testausvaiheen aikana löytyneet virheet kirjataan ylös niiden ilmetessä ja kriittiset virheet korjataan testausvaiheen lopussa. Testausvaihe loppuu, kun kaikki suunnitellut testit on suoritettu, kriittiset virheet korjattu ja testaus on dokumentoitu. (Haikala & Märijärvi 2004.)

Itse testaus voi käsittää useita erilaisia testauksen kohteita, mutta yleensä testattaessa ohjelmistoa ja sen suunnitelmia. Ohjelmistotestauksessa edetään vaiheittain V-mallin mukaisesti. V-mallissa testaus jaetaan kolmeen vaiheeseen: moduulitestaukseen, integrointitestaukseen ja järjestelmätestaukseen. Alfatestauksessa testataan, täyttävätkö tilaustyönä tehty ohjelmisto asiakkaan asettamat vaatimukset. Beta-testaus suoritetaan alfatestauksen jälkeen. Beta-testauksessa asiakas suorittaa itsenäisesti järjestelmän testauksen omissa tiloissaan. Moduulitestauksessa etsitään yksittäisten moduulien virheitä. Integraatiotestauksessa testataan moduulien toimintaa yhdessä ja järjestelmätestauksessa testataan, täyttävätkö valmis ohjelmisto määrittelyn asettamat vaatimukset. Järjestelmätestauksessa testataan myös ohjelmiston suorituskykyä ja tur-

vallisuutta. Seuraavassa kuvassa on esitetty testauksen V-malli. (Haikala & Märijärvi 2004; Kautto 1996)



Kuva 4.5 Testauksen V-malli (Haikala & Märijärvi 2004)

Kuten kuvasta 4.5 huomataan, jo järjestelmän määrittely- ja arkkitehtuurisuunnitteluvaiheessa on muun muassa pohdittava sekä vaiheissa syntyvien dokumenttien testausta että järjestelmätestausta ja integrointitestausta. Määrittelyvaiheessa esimerkiksi laaditaan ensimmäiset testaussuunnitelmat ja testitapaukset. Myöhemmin testausvaiheessa ja testaustilaisuudessa näitä suunnitelmia hyödynnetään testejä tehtäessä. Huolellisesti suunniteltu ja toteutettu testaus johtaa yleensä parempiin tuloksiin kuin suunnittelematon testaus. (Haikala & Märijärvi 2004.)

Ohjelmistolle voidaan suorittaa erilaisia toiminnallisia ja ei-toiminnallisia testejä. Esimerkiksi jo aiemmin mainittiin, että systeemitestauksessa voidaan tehdä ei-toiminnallisia testejä, kuten turvallisuus- ja suorituskykytestejä ohjelmistolle. Tämän lisäksi muita ei-toiminnallisia testejä ovat muun muassa käytettävyydestausta ja kuormitustestausta. Toiminnallisia testejä ovat puolestaan toiminnallisuustestausta, rinnakkaistestausta, yhteensopivuustestausta ja alustatestausta. (Pyhäjärvi & Pöyhönen 2004.)

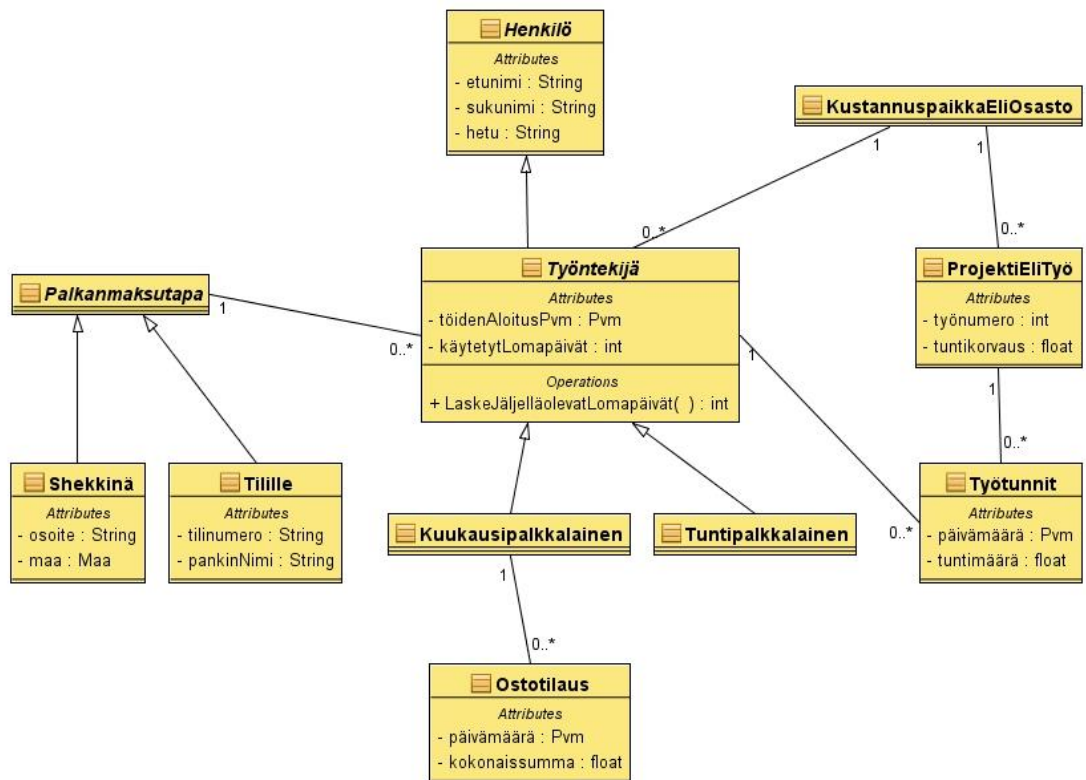
5 TYÖSSÄ KÄYTETYT TEKNIIKAT

5.1 Käsiteanalyysi

Käsitteellisen tason tietosisältömäärittelyn muodostamisesta käytetään yleisesti nimitystä käsiteanalyysi (conceptual modeling). Käsiteanalyysi on työskentelymenetelmä, jonka tarkoituksena on nimetä ja määritellä järjestelmän kehityksessä käytettävä käsitteistö. Se on myös järjestelmän tietokannan suunnittelun ensimmäisiä vaiheita, jonka tarkoituksena on saada aikaan riittävän tarkka kuva järjestelmän tietosisällöstä. Tätä tietosisältöä voidaan jatkossa käyttää pohjana myös tietojärjestelmien toimintojen määrittelylle. (Honkasaari & Fontell 1992; Hovi ym. 2005; Laine 2005.)

Käsiteanalyysissä syntyy käsittemalli, jossa kuvataan käsiteanalyysin lopputulosta eli tietojärjestelmän käsitteistöä, niiden välisiä suhteita ja ominaisuuksia. Riippuen siitä, mihin käsittemallia käytetään, se voi olla hyvinkin karkea kuvaus järjestelmän tietosisällöstä. Käsittemalli kuvataan yleensä käsitekaaviona eli ER-kaaviona, jollakin sovitulla esitystavalla. Esimerkiksi Sinseweb-järjestelmässä käsittemalleja oli kahdenlaisia. Osalla pyrittiin kuvaamaan tietorakenteita ja osalla hahmottamaan tietorakenteiden muotoja. Tietorakenteiden hahmotteluun käytetyt käsittemallit olivat yleensä semanttisia verkkoja, joissa käytettiin solmuja kaarinotaatiota. (Honkasaari & Fontell 1992; Hovi ym. 2005.)

Seuraavassa kuvassa 5.1 on esimerkillinen käsittemalli.



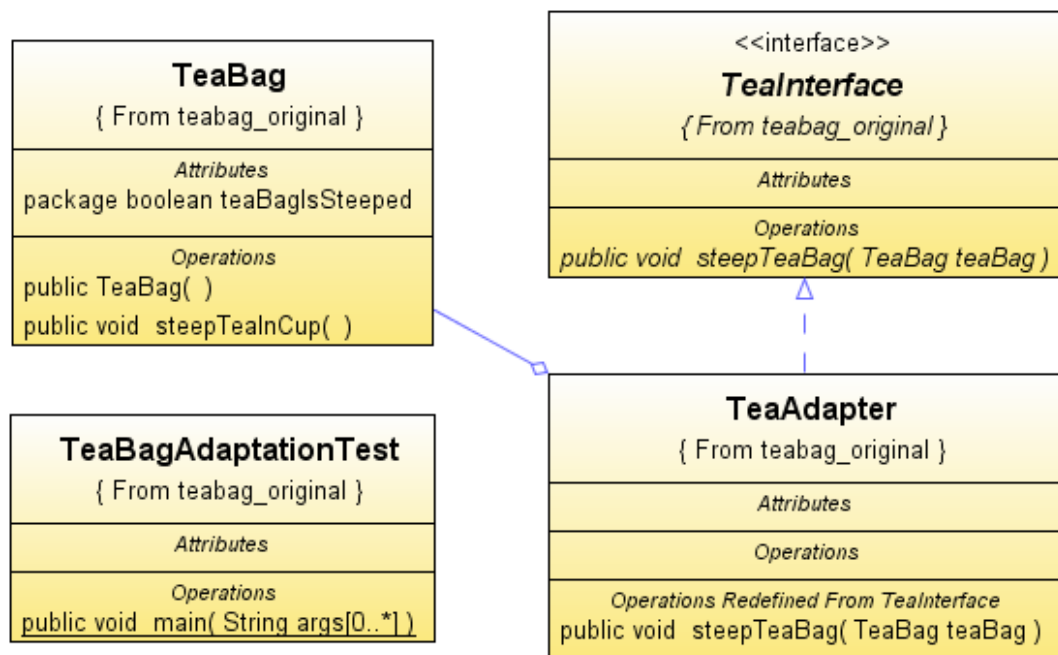
5.1 Käsittemalli.

Käsiteanalyysiä tehtäessä pyritään siihen, että tietty tieto esiintyy käsittemallissa vain kerran. On myös tärkeää, että käsittemalli on työhön osallistuneiden yhdistetty näkemys tutkitun alueen käsitteistöstä. Sen avulla on pystyttävä kommunikoimaan eri alojen ammattilaisten kanssa, jotka liittyvät järjestelmään ja sen kehitykseen jollain tavalla. Käsiteanalyysin käsittemallia tarkennetaan aina käsiteanalyysin jokaisella kierroksella. Tämän prosessin lopputuloksena syntyy kaavio, jonka pohjalta tietokanta voidaan rakentaa. (Honkasaari & Fontell 1992; Hovi ym. 2005.)

5.2 UML

Yhdistetty mallinnuskieli (The Unified Modeling Language) eli UML on standardoitu mallinnuskieli ohjelmisto- ja järjestelmäkehitykseen. UML-kielen on kehittänyt OMG (Object Management Group). Sillä voidaan muun muassa kuvata kehitettävän järjestelmän ympäristöä, rakennetta, toimintaa, arkkitehtuuria, liiketoimintaprosesseja ja tietorakenteita. Seuraavassa kuvassa 5.2 on luokkakaa-

vio, joka on yksi tavallisimpia UML-kielen kaavioita. (OMG 2009, Miles & Hamilton 2006.)



Kuva 5.2 Luokkakaavio.

UML-kieltä käytetään ohjelmistokehityksessä ja varsinkin sen suunnitteluvaiheessa järjestelmän tai ohjelmiston mallintamiseen. Mallintamalla järjestelmää ja sen osia UML-kielellä eri näkökulmista saadaan aikaan tarkka ja täsmällinen kuvaus järjestelmästä. UML-kielellä voidaan laatia käyttäytymistä, vuorovaikutusta ja rakennetta kuvaavia kaavioita. Käyttäytymiskaavioita ovat muun muassa aktiviteettikaaviot, tilakaaviot ja käyttötapauskavioita. Vuorovaikutuskaavioita ovat puolestaan sekvenssikaaviot, kokoavat vuorovaikutuskaaviot, ajoituskaaviot ja kommunikaatiokaaviot. Rakennetta kuvaavia kaavioita ovat muun muassa sijoittelukaaviot, luokkakaaviot ja komponenttikaaviot. Laatimalla luokkakaavioita järjestelmästä voidaan määritellä järjestelmän rakennetta yhdestä näkökulmasta. Käyttämällä luokkakaavioiden lisäksi vielä komponenttikaavioita, jossa luokkakaavion luokat on organisoitu erilaisten komponenttien sisään, saadaan aikaan ylläpidettävämpi ja yksikertaisempi kuva järjestelmän rakenteesta. Sijoittelukaaviolla voidaan vielä jakaa kukin komponenttikaavion komponentti omalle palvelimelleen ja kuvata, miten palvelimet ovat toisiinsa yhteydessä erilaisten

laitteiden avulla. Sekvenssikaavioilla voidaan puolestaan kuvata järjestelmän tapahtumankulkua eri tilanteissa.

Tämän lisäksi käyttämällä UML-kieltä voidaan välttää esimerkiksi työntekijöiden välisiä järjestelmän suunnitelmiin ja kaavioihin liittyviä väärinkäsityksiä. UML-kielen avulla voidaan myös hallita järjestelmän suunnittelussa monimutkaisuutta. (Miles & Hamilton 2006.)

5.3 Ohjelmointikiel

5.3.1 HTML

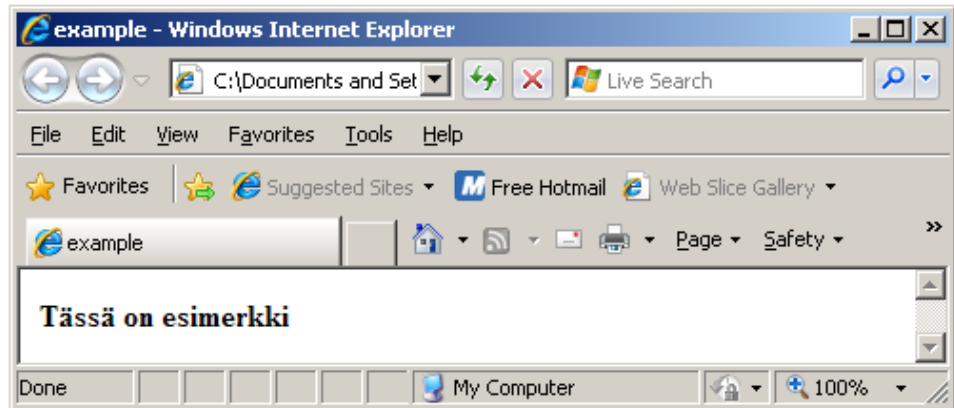
HTML (Hypertext Markup Language) on kieli, jota käytetään web-sivujen määrittelyyn. HTML-koodi muodostuu tageista. Tagi on HTML-kielessä kielen elementti, joka määrittelee tiedon (esimerkiksi kuvat, teksti tms.) ulkomuodon web-sivulla. Web-selaimet puolestaan osaavat tulkita tageja ja tulostaa käyttäjälle web-sivun tagien määrittelemällä tavalla. (Groh ym. 2007.)

HTML-kielen tagit voidaan tunnistaa HTML-dokumentista ulkomuodostaan. HTML-tagia ympäröi kulmasulkumerkit. Tämän lisäksi tagilla on lopetus- ja aloitustagit. HTML-kielessä aloitustageilla tarkoitetaan tageja, jotka aloittavat tietyn elementin kuvaamisen. Lopetustagit puolestaan ilmaisevat elementin kuvaamisen loppumista. Aloitustagi ei eroa edellämäinitusta kuvauksesta niinkään, mutta lopetustageilla on tämän lisäksi vielä ensimmäisen kulmasulkeisen jälkeen tulee aina kauttaviiva. Seuraavassa kuvassa 5.3 on esimerkki HTML-kielestä ja HTML-kielen tagien käytöstä. (Groh ym. 2007.)


```

<html>
  <head>
    <title>example</title>
  </head>
  <body>
    <strong>Tässä on esimerkki</strong>
  </body>
</html>

```



Kuva 5.3 HTML-koodi ja sitä vastaava HTML-dokumentti.

Viimeisin versio 4.01 HTML-kielestä on julkaistu vuonna 1999. Tämän jälkeen HTML-kielestä on alkanut muodostumaan monia muita versioita, kuten XHTML, joka hyödyntää XML-kieltä. (Wikipedia HTML (englanti) 2009.)

5.3.2 PHP

PHP (PHP Hypertext Preprocessor) on dynaamisten web-sivujen kehityksessä käytetty ohjelmointikieli. PHP-kieli sai alkunsa vuonna 1994 ja viimeisin kielen versio on PHP6. PHP on avoimen lähdekoodin kieli ja sitä voidaan siis käyttää maksutta web-kehityksessä. PHP-kieli on palvelinpuolen kieliä, ja sitä käytetään luomaan web-sovelluksia, jotka käyttävät web-palvelimia (esimerkiksi Apache) toimiakseen. PHP-kieltä voidaan tämän lisäksi käyttää myös komentoriviskriptinä. PHP-lähdekooditiedostot käyttävät yleisesti .php tiedostopäätettä. (Suehring, Converse & Park 2009, Welling & Thomson 2003.)

Dynaamisten web-sivujen kehityksessä PHP-koodia ei näy HTML-koodin tavoin asiakaspuolelta käyttäjille. PHP-moottori suorittaa ja lopuksi piilottaa php-dokumentista php-koodin pois, ja asiakas-puolella käyttäjälle näkyy vain HTML-koodi, josta web-sivu koostuu. PHP-kielellä voidaan tehdä web-kehityksessä

tekemään kyselyitä tietokantoihin, kuvien luomiseen, tiedostojen lukemiseen sekä kirjoittamiseen, palvelimien väliseen vuorovaikutukseen ja niin edelleen. (Zandstra, M. 2004.)

5.3.3 JavaScript ja jQuery

JavaScript on Netscapen kehittämä web-kehityksessä käytetty ohjelmointikieli, jota voidaan käyttää yhdessä HTML-kielen kanssa. JavaScript-kielen avulla voidaan luoda interaktiivisia web-sivustoja. JavaScriptin avulla on mahdollista käsitellä HTML-dokumentin kaikkia elementtejä. JavaScriptin on asiakaspuolen kieli, ja se käännetään ja ajetaan dynaamisesti web-selaimessa. JavaScript-kieltä voidaan joko sijoittaa suoraan HTML-kieltä sisältäviin tiedostoihin tai se voidaan sijoittaa kokonaan omaan tiedostoonsa. (Vander Veer 2005, Heilmann 2006.)

Vuonna 2006 julkaistiin jQuery niminen JavaScript-kirjasto, joka keskittyy JavaScriptin ja HTML:n väliseen vuorovaikutukseen. jQuery on kehitetty helpottamaan JavaScript-kielen käyttöä, ja se tarjoaa yksinkertaisemman tavan kirjoittaa JavaScript-koodia. jQuery:n sijoittaminen toimii samalla tavalla kuin JavaScript-koodi. Tämän lisäksi jQuery:n käyttö edellyttää, että tiedostoon, jossa sitä käytetään, lisätään viittaus jQuery-kirjastoon. jQuery-kirjastossa on määritelty kaikki jQuery:n funktiot, ja viimeisin versio tästä kirjastosta on 1.3.2, joka on julkaistu helmikuussa 2009. (Wikipedia JavaScript libraries (englanti) 2009, HTML-advisor 2009.)

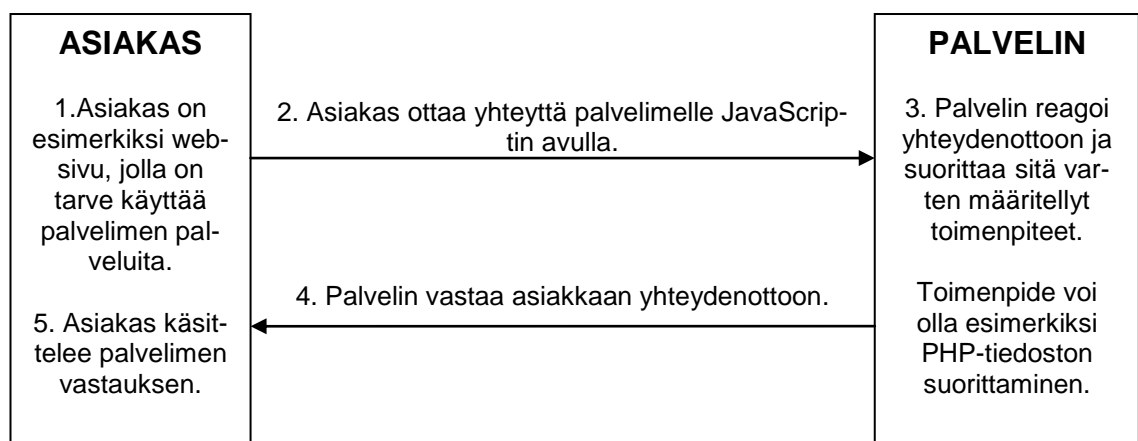
5.3.4 Ajax

Ajax (Asynchronous JavaScript and XML) on tekniikka, jota käytetään tiedonsiirtoprosessista asiakas- ja palvelinskriptin välillä. Tämän tekniikan avulla pystytään esimerkiksi noutamaan tieto web-palvelimelta lähettämättä uudelleen aktivoitua web-sivua. Ajaxia käyttämällä käyttäjää ei tarvitse keskeyttää ja voidaan

luoda sulavammin toimivaa vuorovaikutusta asiakkaan ja palvelimen välille. (Vohra 2008.)

JavaScript on tärkeä osa Ajax-tekniikkaa. JavaScript hoitaa vuorovaikutuksen asiakaspuolelta ja suorittaa yhteydenoton palvelinpuoleen. Palvelinpuolelle ei ole kuitenkaan määritelty mitään tiettyä kieltä. Yleensä Ajaxia käytettäessä palvelinpuolen tehtäviä hoidetaan PHP-kielellä.

Kun asiakaspuoli lähettää JavaScript-viestin palvelimelle, palvelin reagoi yhteydenottoon ja suorittaa toimenpiteet. Lopuksi palvelin palauttaa asiakkaalle XML-muotoisen viestin, jonka asiakaspuoli käsittelee. Tätä prosessia on kuvattu kuvassa 5.4, jossa tehtävät on myös numeroitu.



Kuva 5.4 Ajax-tekniikan toiminta.

Tunnetuimpia Ajax-tekniikkaa hyödyntäviä sovelluksia ovat Googlen Gmail ja Microsoftin Outlook Express –sähköpostipalvelut. Myös Drupal-sisällönhallintajärjestelmä käyttää Ajaxia täydentämään tietynlaisiin tekstikenttiin kirjoitettua tekstiä. Drupalissa on myös olemassa oma komponenttinsa (AHAH), jolla voidaan sujuvasti lisätä, poistaa ja muuttaa Drupal web-sivun (esimerkiksi lomakkeen) elementtejä. (Vohra 2008.)

5.4 SQL, RDMS ja MySQL

SQL (Structured Query Language) on strukturoitu kyselykieli, joka on tarkoitettu tietokantakyselyiden teon lisäksi myös tietokantojen luomiseen ja määrittelyyn, tietokannan tietosisällön päivittämiseen, lisäämiseen ja poistamiseen, valtuuksien ja turvallisuuden hoitoon, tapahtumankäsittelyn ohjaamiseen, upotettuun SQL:aan ja kohdistimien (eli kursorien) hallintaan. SQL:aa voidaan käyttää erityisen käyttöliittymän kautta tai upotettuna ohjelmointikieleen. (Hovi 2000.)

Seuraavassa kuvassa on esimerkki SQL-kielestä. Kuvan 5.5 ohjelmakoodilla luodaan tietokantaan taulu.

```
CREATE TABLE IF NOT EXISTS `A` (  
  `id` int(11) NOT NULL auto_increment,  
  `name` varchar(255) NOT NULL,  
  PRIMARY KEY (`id`)  
 ) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=13 ;
```

Kuva 5.5 SQL-kielen ohjelmakoodinäyte.

Tietokantojen hallintajärjestelmät (RDMS), kuten MySQL ja Microsoft Access, käyttävät hyväkseen SQL:aa standardoituna tietokantakielenä. Tosin riippuen järjestelmästä SQL-kielen syntaksi saattaa jonkun verran poiketa perus SQL-kielestä. MySQL noudattaa SQL-standardia pääpiirteittäin, ja siihen on tehty joitakin SQL-standardista poikkeavia laajennuksiakin. (OAMK (MySQL), Art Branch Inc 2004.)

MySQL on web-palveluiden kehittämisessä suosittu ilmainen tietokantojen hallintajärjestelmä. MySQL perustuu GNU-lisensointiin, ja siksi sitä voi käyttää kuka tahansa. (OAMK (MySQL), Wikipedia (suomi) SQL 2009.)

5.5 Ohjelmat, komponentit ja järjestelmät

5.5.1 Internet-selain

Selain on ohjelma, jolla on mahdollista käyttää hypermediadokumenttien verkostoa (eli World Wide Web), jossa yhdistyy hyperteksti, multimedia ja tietoverkot. Jotta selaimella pystytään käyttämään hypermediadokumenttien verkostoa, on sen tunnettava muun muassa CSS-, RSS-, HTML- ja XHTML-protokollia. Edellä mainittujen protokollien tunteminen on selaimen toiminnalle tärkeää, sillä niiden avulla se esittää web-sivuja käyttäjilleen. Riippuen selainten käyttämistä moottoreista, joilla web-sivujen ulkoasu määritetään, web-sivujen ulkoasu voi vaihdella selainkohtaisesti. Tämän lisäksi selaimen on pystyttävä muokkaamaan edellä mainittujen kielten tulostetta. (Multisilta; Tech-FAQ 2009.)

Selaimia on olemassa kahdenlaisia: graafisia ja tekstipohjaisia, joista graafiset selaimet ovat yleisempiä. Sinseweb-järjestelmän kehitystyössä on käytetty yleisimpiä graafisia selaimia. Näitä selaimia ovat muun muassa Internet Explorer, Mozilla Firefox ja Google Chrome. (Tech-FAQ 2009.)

Tekstipohjaiset selaimet ovat harvinaisia internetin käyttäjien keskuudessa. Tekstipohjaiset selaimet tarjoavat käyttäjilleen tekstipohjaisen käyttöliittymän, jolla voidaan selata internetiä. Nämä selaimet myös tarjoavat korkeamman suorituskyvyn matalan siirtonopeuden linkeille ja tukevat enemmän erilaisia laitteita kuin graafiset selaimet. Suosituimpia tekstipohjaisia selaimia ovat muun muassa Lynx ja e-links. (Tech-FAQ 2009.)

5.5.2 Mozilla Firefox

Mozilla Firefox on ilmainen avoimen lähdekoodin internet-selain, jonka on kehittänyt Mozilla Corp. Firefox on tilatojen mukaan toiseksi käytetyin web-selain ja vuoden 2009 elokuussa noin 22 % internetin käyttäjistä käytti Firefox-selainta.

Internetin suosituin selain on Internet Explorer. (Wikipedia Mozilla Firefox (englanti) 2009.)

Web-sivustojen esittämiseen Firefox-selain hyödyntää Gecko layout engine-moottoria, joka käyttää tämän hetken yleisempiä web-standardeja ja monia muita ominaisuuksia, joita oletetaan lisättäviksi standardien joukkoon. (Wikipedia Mozilla Firefox (englanti) 2009.)

Viimeisimmän Firefox-selaimen ominaisuuksia ovat muun muassa välilehtien avulla surffailu, oikeinkirjoituksen tarkastus, inkrementaalinen haku, live bookmarkkaus (eli reaaliajassa tapahtuva sivujen merkkkaus suosikiksi), latausmanageri, yksityinen surffailu, paikkatietoinen surffailu (ts. geolocation), joka perustuu Google palveluun, ja sisäänrakennettu haku, joka käyttää oletuksenaan Google-hakukonetta. Toimintoja voidaan lisätä selaimen asentamalla useiden eri toimittajien kehittämiä lisäosia. Lisäosat ovat Firefox-selaimessa yksi käyttäjiä puoleensa vetävä ominaisuus. (Wikipedia Mozilla Firefox (englanti) 2009.)

Firefoxista on monia eri versioita eri käyttöjärjestelmille, kuten Mac OS X, Microsoft Windows, Linux ja niin edelleen. Firefoxin viimeisin versio on 3.5.2, joka on julkaistu vuoden 2009 elokuussa. Firefoxin ohjelmakoodi on vapaasti luettavissa, koska se on julkaistu kolmen lisenssin alaisuudessa (GNU GPL, GNU LGPL, MPL). Viralliset versiot on puolestaan julkaistu EULA-lisenssin alaisuudessa. (Wikipedia Mozilla Firefox (englanti) 2009.)

5.5.3 Internet Explorer

Internet Explorer (IE) eli Microsoft Internet Explorer (MSIE), on käytetyin internet-selain. Viimeisimpien tilastojen mukaan noin 68 % internetin käyttäjistä käyttää Internet Explorer -selainta. Internet Explorer tulee Microsoft Windows -käyttöjärjestelmän mukana oletusselaimena. Selaimen voi myös ladata Microsoftin web-sivuilta. (Market Share 2009, SearchEnterpriseDesktop.com 2008.)

Internet Explorer-selaimesta on historiansa aikana ollut jo seitsemän versiota ja sille on tehty lukuisia tietoturvapäivityksiä ja paikkauksia, joilla on pyritty estämään selaimen kohdennettuja hyökkäyksiä, jotka hyödyntävät yleensä selaimen tunnettuja virheitä ja tietoturva-aukkoja. Viimeisin versio selaimesta on Internet Explorer 7, joka tarjoaa käyttäjilleen paljon samanlaisia ominaisuuksia, joita sen kilpailijat ovat kehittäneet (esimerkiksi Mozilla Firefox). Näitä ominaisuuksia on muun muassa välilehtien avulla surffailu, CSS-tuki, RSS-syötteet ja lupaus lisätä tietoturvaa haittaohjelmia ja verkkourkintaa vastaan. (SearchEnterpriseDesktop.com 2008.)

5.5.4 Google Chrome

Google Chrome on avoimen lähdekoodin internet-selain, joka käyttää WebKit-moottoria sivujen esittämiseen. Viimeisimpien tilastotietojen mukaan internetin käyttäjistä noin 2 % käyttää Google Chromea selaimenaan. (Market Share 2009, Linux.fi. 2009.)

Google Chrome-selaimen etuja ovat muun muassa nopea JavaScriptin suorittaminen ja helppokäyttöisyys. Chromessa on Firefoxin ja Internet Explorerin tapaan käytössä välilehdet. Chromessa välilehdet muodostavat omat prosessinsa, ja tällä tavalla Chrome saavuttaa myös tietoturvaetua ja vakautta, sillä yhdellä välilehdellä tapahtuva toiminnan jumiutuminen ei pysäytä koko selaimen käyttöä. Tämän lisäksi myös Chromessa on mahdollista asentaa siihen lisäosia, jotka eivät vaikuta muuten selaimen toimintaan. Lisäosat on suljettu omiin muusta selaimen toiminnallisuudesta eristettyihin ympäristöihin ja näin erotettu perusselaimesta. (Linux.fi. 2009.)

Google Chrome-selaimesta on olemassa nykyisin versiot Microsoft Windows Vista ja XP (SP2) -käyttöjärjestelmille. Google Chromesta on myös tulossa Linux-versio, joka on vielä toistaiseksi keskeneräinen. Linux-versio Chromesta on kuitenkin jo käyttäjien saatavilla. (Linux.fi. 2009.)

5.5.5 Vim editor

Vim on GPL-lisenssin alaisuudessa oleva kevyt tekstieditori, joka on rakennettu tehokasta tekstin editointia varten. Vim on paranneltu versio Vi-editorista, joka on käytössä suurimmalla osalla UNIX-järjestelmistä. Vim-editori on ilmaisohjelma ja sitä voi käyttää maksutta. Tosin editorin tekijät pyytävät käyttäjiltä, jotka pitävät editoria hyödyllisenä, lahjoituksia, jotka menevät Ugandan lapsille. (Vim online.)

Vim-editoria kutsutaan useasti ohjelmoijan editoriksi ja sitä pidetään hyödyllisenä ohjelmointityökaluna. Vim ei ole kuitenkaan pelkästään ohjelmoijille vaan se on hyödyllinen työkalu kaikessa tekstin editoinnissa, sähköpostiviestien laatimisessa ja konfiguraatitiedostojen muokkaamisessa. (Vim online.)

5.5.6 Drupal

Drupal on sisällönhallintajärjestelmä (CMS), jota käytetään web-sivustojen rakentamisessa. Se on modulaarinen vapaan lähdekoodin ilmaisjärjestelmä. Drupal sisältää oletuksena kaiken web-sivujen rakentamiseen tarvittavan perustoinnallisuuden, mutta sen toiminnallisuutta voidaan laajentaa asentamalla siihen erilaisia moduuleita. Drupalilla voidaan rakentaa muun muassa web-portaaleja, web-sivustoja, verkkokauppoja, hakemistoja, online-sanomalehtiä, kuvagallerioita ja niin edelleen. (VanDyk & Westgate 2007.)

Drupal noudattaa lähdekoodissaan hyvin tarkasti ohjelmoimisstandardeja. Tämän lisäksi tietoturvallisuudesta on huolehdittu. Kaikki Drupalin tarjoamat funktiokirjastot, komponentit ja muut valmiit ratkaisut on suojattu väärinkäytön varalta. (VanDyk & Westgate 2007.)

Drupal antaa myös käyttäjille mahdollisuuden käyttää HTML- ja PHP-ohjelmointikieliä. Drupalilla on käytössä tietokantoja; tyypillisesti Drupalin kanssa on käytössä joko MySQL- tai PostgreSQL-tietokanta. (VanDyk & Westgate 2007.)

5.5.6.1 Drupalin moduulit

Drigg

Drigg on Drupal-moduuli, jonka avulla on mahdollista laatia äänestettäviä sivuja eli Linkkejä (Scoop). Äänestettävyydellä tässä tapauksessa tarkoitetaan sitä, että käyttäjän voivat arvioida sivustoja ja antaa nille arvosanansa. Käyttäjä voi muun muassa antaa sivulle plus- tai miinusäänen. Tämän ominaisuuden avulla sivut voidaan järjestelmissä asettaa paremmuusjärjestykseen käyttäjien antamien äänien perusteella. Äänet ja muut Driggin vaatimat tiedot sivustoista tallennetaan Driggin omaan tauluun tietokannassa.

Drigg-moduuli on riippuvainen vain kahdesta muusta moduulista, joita ovat User_karma ja Voting form. Edellämainitut moduulit mahdollistavat Drigg-moduulin toiminnan. (Drupal.org Drigg 2007.)

Favorite Nodes

Favorite Nodes-moduulin avulla käyttäjät voivat merkitä internet-sivuja suosikeiksi. Suosikit kerätään Drupalin tietokantaan, jonne Favorite Nodes-moduuli automaattisesti lisää oman taulunsa suosikkien tallennusta varten. Drupal-järjestelmässä Favorite Nodes-moduuli mahdollistaa myös suosikkien katselun käyttäjäprofiilista, jossa näytetään lista kaikista käyttäjän lisäämistä suosikeista. (Drupal.org Favorite Nodes 2006.)

Taxonomy

Taxonomy-moduuli mahdollistaa sivustojen luokittelun valittujen kategorioiden perusteella. Drupal-kehittäjät, jotka käyttävät Taxonomy-moduulia, voivat luoda kategorioita ja järjestellä niitä ylä- ja alakategorioiksi halutessaan. Kategoriat voidaan myös asettaa omiin sanastoihinsa, ja yhdelle kategorialle voidaan määrittellä synonyymejä. Tämän ominaisuuden avulla on mahdollista lajitella sivustoja eri kategorioihin. Esimerkiksi musiikkiin liittyvät linkit voidaan asettaa kategoriaan musiikki. (Drupal.org Taxonomy 2009.)

Form API

Drupalin Form API-moduulin avulla on mahdollista laatia yksinkertaisia peruslomakkeita Drupal-sivuille helposti. Form API antaa mahdollisuudet laatia peruslomakkeille myös mukautettuja teemoituksia, lomakkeiden oikeellisuustarkistuksia ja lomakkeiden suorituksia. Lomake-elementtejä voidaan helposti lisätä, poistaa siirrellä ja järjestellä Form API-moduulin avulla. (Drupal.org Form API 2006.)

Comment

Drupalin Comment-moduuli mahdollistaa muun muassa linkkien, blogien, keskustelu foorumeiden aiheiden ja tarinoiden kommentoinnin. Kommentointi on tärkeä ominaisuus haluttaessa saada järjestelmän web-yhteisöt keskustelemaan ja harrastamaan järjestelmän välityksellä sosiaalista kanssakäyntiä. Aivan kuten muutamissa muissakin edellä mainituissa moduuleissa, niin myös tämäkin täydentää Drupalin omaa tietokantaa yhdellä lisätaululla, jonne käyttäjien omat kommentit talletetaan. (Drupal.org Comment 2009.)

Profile

Profile-moduulin avulla käyttäjät voivat tallettaa järjestelmään lisää tietoa itsestään. Moduuli laajentaa Drupalin jo olemassa olevaa käyttäjäprofiilia siten, että kehittäjät pystyvät luomaan käyttäjän profiiliin lisää kenttiä, jonne käyttäjät pystyvät syöttämään tietoa itsestään. (Drupal.org Profile 2009.)

5.5.7 Yahoo! Boss API

Yahoo! BOSS API (BOSS eli Build your Own Search Service) on Yahoon oma ja ilmainen vapaan lähdekoodin web-palvelu alusta. Palvelun tarkoituksena on edistää innovaatiota hakualalla. Kehittäjät, aloittelijat ja suuren internetet-alan yritykset voivat käyttää palvelua rakentamaan ja julkaista web-mittakaavan hakutuotteitaan, jotka käyttävät hyväkseen Yahoon-hakuindeksiä. Palvelun avulla

voidaan käyttää muun muassa Yagoon valmiita indeksointi-, järjestys- ja asi-
aankuuluvuusalgoritmejä sekä infrastruktuuria. Toisin kuin jotkin toiset vastaa-
vat palvelut se antaa käyttäjilleen mahdollisuuden uudelleenindeksoida ja järjes-
tellä hakutulostuloja. (Yahoo! Development network.)

5.5.8 PhpMyAdmin

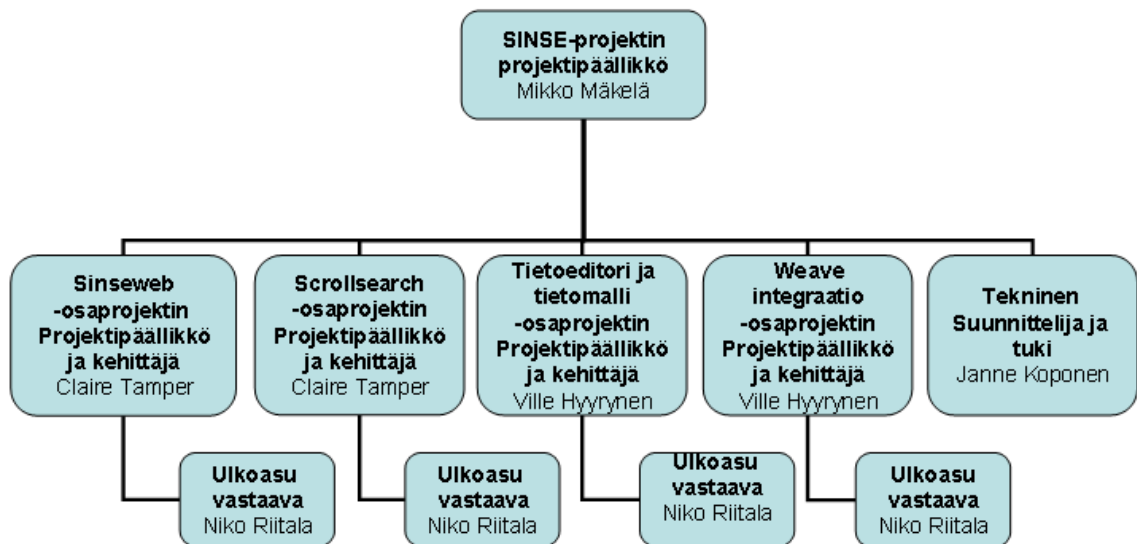
PhpMyAdmin on ilmainen työkalu, jonka avulla käyttäjät pystyvät käsittelemään
MySQL-tietokantojaan. phpMyAdmin on PHP-ohjelmointikielellä luotu järjestel-
mä, jota voidaan käyttää internet-selaimella. Tällä hetkellä phpMyAdmin-
järjestelmän avulla on mahdollista luoda ja ylläpitää tietokantoja ja niiden taulu-
ja. Tämän lisäksi voidaan myös tehdä MySQL-kyselyjä tietokantojen tauluihin.
Näiden ominaisuuksien lisäksi on myös mahdollista tallentaa tietokannat teksti-
tiedoihin, joiden avulla voidaan siirtää tietokantoja järjestelmästä esimerkiksi
toiseen phpMyAdmin-järjestelmään. (Haas.)

6 SINSEWEB – PROJEKTIN VAIHEET

Tässä luvussa käydään läpi Sinseweb-projektin organisointi ja vaiheet.

6.1 Organisointi

Sinse-projektin organisaatio on esitetty kuvassa 6.1.



Kuva 6.1 Projektiorganisaatio.

Sinse-projektin päällikkönä toimii Mikko Mäkelä. Sinse-projekti jakautuu neljään osaprojektiin: Sinseweb, Scrollsearch, Weave integraatio ja Tietoeditori ja tietomalli. Projektiryhmän muita jäseniä olivat muun muassa Niko Riitala, Ville Hyyrynen, Janne Koponen ja Mikko Mäkelä. Janne Koponen oli suunnittelija sekä tekninen tuki ja Niko Riitala vastasi ulkoasusta Sinse-projektissa. Sinseweb-osaprojektin lisäksi itse olin vastuussa myös toisesta osaprojektista nimeltä Scrollsearch ja Ville Hyyrynen oli vastuussa muista osaprojekteista.

Tehtävien jakamisen lisäksi työt oli priorisoitava ja jaettava ensisijaisiin ja toissijaisiin tehtäviin, joista ensisijaiset tehtävät oli suoritettava ensin ja toissijaiset tehtävät myöhemmin. Ensisijaisia tehtäviä olivat kaikki perustoiminnallisuuden,

käyttöliittymän ja ominaisuuksien tekemiseen liittyvät tehtävät. Toissijaisia tehtäviä olivat puolestaan erilaisten hakutyyppeiden toteuttaminen, suosikkien hallinnan osatehtävät, kuten suosikkien poisto ja muokkaus, sekä hakuasetussivu. Seuraavassa taulukossa 5.1 on lueteltuna opinnäytetyöhön kuuluvat priorisoidut tehtävät. Taulukossa tehtävät on priorisoitu numeroasteikossa 1 - 2, jossa 1 on ensisijainen tehtävä ja 2 toissijainen tehtävä. Samaisessa taulukossa on myös kerrottu tekijät, jotka tehtäviin liittyvät, koska jotkut tehtävät on laadittu yhdessä Tiksis Technologies Oy:n muiden työntekijöiden kanssa. Tekijöiden nimissä on käytetty lyhenteitä: CT eli Claire Tamper, MM eli Mikko Mäkelä, JK eli Janne Koponen, NR eli Niko Riitala ja VH eli Ville Hyyrynen.

Taulukko 5.1 Priorisoitu tehtävälista.

Tehtävä	Tekijät	Prioriteetti
Järjestelmän määrittely	CT, MM	1
Järjestelmän arkkitehtuurin suunnittelu ja kuvaaminen	CT, JK	1
Järjestelmän tietomallin suunnittelu ja kuvaus	CT, JK, VH	
Suunnittelu- ja määrittelydokumentaatio	CT	1
Hakukäyttöliittymän suunnittelu	CT	1
Hakukäyttöliittymä-prototyypin toteutus	CT	1
Hakukäyttöliittymän hienosäätö	CT, NR	2
Hakukäyttöliittymän perustoiminnallisuus		
- Hakujen tekeminen web-haulla	CT	1
- Hakujen tekeminen web- ja kuvahaulla.	CT	2
- Hakutulosten hakeminen Yahoo:n ja Drupalin tietokannoista	CT	1
- Hakutulosten indeksointi	CT	1
Hakustrategian suunnittelu ja toteutus	CT	1
Hakuasetukset-sivun toteutus	CT	2
Semanttisen parserin toteuttaminen	CT	1
Semanttisen avusteen luominen	CT	1
Hakutuloksen lähettäminen sähköpostitse	CT	2
Hakutuloksen kommentointi	CT	2
Suosikkien hallinta		
- Hakutuloksen lisääminen suosikkeihin	CT	1
- Hakutuloksen poistaminen suosikeista	CT	2
- Suosikin muokkaus	CT	2

6.2 Vaiheistus

Priorisoinnin ja tehtävien jaon lisäksi Sinseweb-projektin tehtävät jaettiin eri vaiheisiin. Vaihejako tapahtui aluksi elämänkaarimallien mukaisesti, ja myöhemmin elämänkaarimallin mukaan jaotellut tehtävät purettiin vielä pienempiin tehtäviin. Projektin vaiheistus on kuvattu liitteessä 3, joka sisältää projektin aikataulut.

6.3 Projektin kulku

Sinseweb-projektissa projekti eteni pääpiirteittäin vesiputousmallin mukaisesti. Projektin toteutusvaiheessa edettiin suurimmaksi osaksi prototyypimallin mukaisesti. Projektin aikana ilmeni myös muutamia elämänkaarimallien tyypillisiä ongelmia. Seuraavissa luvuissa on kuvattu elämänkaarimallien vaiheet ja niiden kulku.

6.3.1 Esitutkimus

Esitutkimusvaiheessa laadittiin toukokuun 2009 aikana tulevasta Sinseweb-projektista esitutkimusraportti. Vaiheen tarkoituksena oli selvittää, onko kannattavaa aloittaa Sinseweb-projekti. Kannattavuutta tarkasteltiin projektin arvioidun käytettävän ajan, resurssien, konseptin, riskien ja toteutustekniikoiden avulla. Tämän vaiheen lopputuloksena syntyi esitutkimusraportti, jossa selvitetty, mitä projektin toteuttaminen vaatisi ja mitkä ovat sen vaatimat resurssit.

6.3.2 Projektin aloitus

Sinseweb-projekti aloitettiin kesäkuussa 2009. Projektin aloituksen jälkeen projektiryhmä oraganisoitui kuvan 6.1 mukaisesti.

6.3.3 Määrittely

Sinseweb-projektin ensimmäinen varsinainen tehtävä oli määrittellä itse tuote. Määrittelyvaiheessa laadittiin Sinseweb-järjestelmän versio 1.0 toiminnallinen määrittely (liite 1). Määrittelydokumentissa kuvataan järjestelmän tietomalli, käyttötapaukset, alustava käyttöliittymä, arkkitehtuuri, ominaisuudet, rajoitteet ja liittymät.

Ensimmäisenä määrittelyvaiheessa kuvattiin tietomalli. Tämän jälkeen kuvattiin järjestelmän arkkitehtuuri, joka oli jo osittain määritelty esitutkimusvaiheessa. Määrittelyvaiheen aikana järjestelmän arkkitehtuurista laadittiin UML-kaaviot ja sanalliset kuvaukset.

Käyttötapausten määrittely aloitettiin käyttötapausten luetteloinnilla. Käyttötapausta syntyi 7 kappaletta. Kaikki käyttötapaukset käytiin läpi ja niistä laadittiin yksityiskohtaiset käyttötapauskuvaukset. Samalla myös laadittiin alustava kuvaus käyttöliittymästä.

Viimeiseksi määrittelyvaiheessa määriteltiin järjestelmän ominaisuudet, rajoitteet ja liittymät. Mikko Mäkelä oli mukana ominaisuuksien ja liittymien määrittelyssä ja Janne Koponen auttoi rajoitteiden kartoituksessa. Kaikkein suurimpana rajoitteena ja riskinä pidettiin Drupalia. Määrittelyvaiheessa ei kuitenkaan havaittu Drupalin aiheuttavan vaikeuksia toteuttamisessa.

6.3.4 Suunnittelu

Suunnittelu käynnistyi Sinseweb-projektissa jo määrittelyvaiheen aikana kesäkuussa 2009. Suunnittelu tuloksena syntyi suunnitteludokumentti, joka laadittiin heti toiminnallisen määrittelyn valmistuttua. Suunnitteluvaiheessa käytimme kaavioiden piirtämisessä UML-kieltä.

Sinseweb-järjestelmän suunnitteluvaiheessa ensimmäisenä tehtävänä oli tehdä lyhyt tutkimus semantiikasta internetissä. Tutkimuksen tarkoituksena oli tutkia jo

olemassa olevia semanttisia hakukoneita, semantiikan toteutusta niissä ja tutustua alan terminologiaan. Tutkimus saavutti melkein kaikki tavoitteensa, mutta sen avulla ei löydetty valmiita tekniikoita tai menetelmiä toteuttaa semantiikkaa järjestelmän versiossa 1.0. Ainut hyöty tutkimustuloksista oli tietomallin suunnittelussa. Sen suunnittelussa hyödynnettiin valmiita semanttisia verkkoja, joita semantiikan tutkijat olivat laatineet.

Ensimmäinen suunnittelun kohde oli järjestelmän tietomalli, joka suunniteltiin jo määrittelyvaiheen aikana. Tietomallin huolellinen suunnittelu oli tärkeää, sillä koko järjestelmä rakennettiin sen varaan. Tietomallin suunnittelussa olivat itseni lisäksi mukana myös Tiksis Technologies Oy:n työntekijöistä Ville Hyyrynen ja Mikko Mäkelä. Lähdimme liikkeelle tietomallin suunnittelussa Mikko Mäkelän alkuperäisistä ideoista. Pohdimme myös Drupalin miten tietokanta voisi palvella järjestelmäämme.

Suunnittelussa vaadittiin asiantuntemusta työkaluna käytettyjen tietomallien ja käsittemallien laatimisessa. Loimme semanttisia verkkoja erilaisten hakusanan määrittelyyn liittyvien testitapausten avulla. Loppujen lopuksi onnistuimme luomaan järjestelmälle toimivan tietomallin ja käsittekaavioita.

Järjestelmän tietomallin suunnittelun jälkeen jatkettiin järjestelmän suunnittelua käyttötapausten kuvaamisella ja realisoinnilla. Käyttötapausten etenemistä tehtiin lyhyet sanalliset kuvaukset ja käyttötapauskaaviot sekä sekvenssikaaviot.

Käyttötapausten jälkeen suunnittelussa siirryttiin järjestelmän rakenteen ja arkkitehtuurin kuvaamiseen. Suunnitteluvaiheessa keskityttiin arkkitehtuurin kuvaamisessa enimmäkseen järjestelmän moduulien ja komponenttien kuvaamiseen tarkemmin. Rakenteen ja arkkitehtuurin suunnittelua seurasi semantiikan käytön suunnittelu, hakustrategian suunnittelu ja käyttöliittymän suunnittelu. Semantiikan käytön suunnittelussa perehdyttiin pohtimaan tietomallin hyödyntämistä järjestelmässä. Hakustrategian suunnittelussa pohdittiin sitä miten käyttäjä voisi saada enemmän itselleen sopivia tuloksia järjestelmän ensimmäisessä versiossa. Hakustrategian suunnittelussa hyödynnettiin asiakkaan valmiita

ideoita ja tutkittiin hieman, millaisilla hakusanoilla hakukoneet löytävät parhaimpia tuloksia.

Viimeisenä suunniteltavana oli käyttöliittymän suunnittelu. Käyttöliittymän suunnittelussa otettiin ensisijaisesti mallia Googlen ja muiden hakukoneiden käyttöliittymistä. Käyttöliittymistä piirrettiin ensin luonnokset paperille. Tämän lisäksi suunniteltiin järjestelmän navigointi ja erilaisten elementtien hyödyntäminen käyttöliittymärakennuksessa. Suunnitelmien valmistuttua käyttöliittymästä alettiin rakentamaan käyttöliittymäprototyyppiä suunnitelmien perusteella.

6.3.5 Toteutus

Järjestelmän toteutusvaihe käynnistyi heinäkuussa 2009 heti toiminnallisen määrittelyn, suunnitteludokumentin ja käyttöliittymäprototyypin valmistuttua. Järjestelmän toteuttamiseen käytettiin Drupalin lisäksi seuraavia ohjelmointikieliä ja kirjastoja: PHP, HTML, JavaScript ja jQuery. Vim-editoria käytettiin puolestaan ohjelmointityökaluna. Myös MySQL-kieltä käytettiin toteuttamisessa. MySQL-lausekkeet testattiin ja laadittiin aina phpMyAdmin-sovelluksella. Itse järjestelmän toimintaa testattiin toteutusvaiheen aikana erilaisilla selaimilla, joita olivat Internet Explorer, Mozilla Firefox ja Google Chrome.

Itse toteutusvaihe oli alustavasti jaettu kolmeen vaiheeseen, joista ensimmäisessä laadittiin järjestelmälle hakukäyttöliittymän perustoiminnallisuus, hakuasetukset sivu, hakutulosten kommentointi ja hakutuloksen lähettäminen sähköpostitse. Ensimmäisessä vaiheessa toteutettiin ensisijaisesti vain ne ominaisuudet, jotka oli merkitty ensisijaisiksi (prioriteetti 1) taulukossa 5.1. Toissijaiset ominaisuudet toteutettiin vain, jos aikaa oli riittävästi toteuttaa ne. Ominaisuuksien toteuttamisen lisäksi jokaiseen vaiheeseen sisältyi myös niiden toiminnan testaamista.

Ensimmäisessä vaiheessa tehtävät ja ongelmat olivat yksinkertaisia. Ne sujivat Drupalin opiskelun ohessa suhteellisen mutkattomasti. Kaikki ratkaisut, joita tämän vaiheen aikana tehtiin, eivät kuitenkaan olleet standardien mukaisia

vaikkakin toimivia. Tämä johtui siitä, että minulla tai kenelläkään muulla heinäkuussa paikalla olleella työntekijällä ei ollut vielä tarpeeksi kokemusta Drupalista. Ne, jotka tiesivät Drupalista enemmän, olivat lomalla eikä heihin saanut yhteyttä. Heidän palattuaan jouduimme tekemään joitakin asioita uudestaan Drupalin standardien mukaisesti.

Toisessa vaiheessa oli tehtävänä laatia suosikkien hallinta, joka käsitti suosikkien lisäyksen, poiston ja muokkauksen. Tässä vaiheessa tehtävät jaettiin ensi- ja toissijaisiin tehtäviin samalla tavalla kuin ensimmäisessä vaiheessa. Toinen vaihe sujui kaikista parhaiten eikä ongelmia ilmennyt kehityksen aikana. Kuten ensimmäisen vaiheen myös toisen vaiheen aikana syntyneet tuotokset jouduttiin muokkaamaan Drupalin standardien mukaisiksi, joka oli hyvin pieni tehtävä.

Kolmannessa vaiheessa, joka alkoi elokuussa 2009, oli tarkoitus laatia hakukoneesta semanttinen sekä toteuttaa hakustrategiat. Kolmannessa vaiheessa tehtävät jaettiin myös ensimmäisen vaiheen mukaisesti ensi- ja toissijaisiin tehtäviin. Järjestelmän kolmannen vaiheen tehtävät osoittautuivat monimutkaisemmiksi kuin ensimmäisen vaiheen tehtävät. Drupalin joustamattomuus aiheutti toisinaan ongelmia, joihin piti keksiä monimutkaisia ratkaisuja. Toteutuksen loppuvaiheilla voitiin jo todeta, että Drupalista oli ollut yhtä paljon haittaa kuin hyötyäkin järjestelmän kehityksessä.

Monimutkaisuuden ja Drupalin aiheuttamien ongelmien takia elokuun loppupuolella kyseenalaistettiin koko järjestelmän toiminta ja pohdittiin uusia toimintaratkaisuja siihen, koska järjestelmän toteutuksen loppuvaiheessa ei haluttu laittaa opinnäytetyöhön kuulunutta määrittelyä uusiksi, sain mahdollisuuden kehittää sen loppuun yksin. Tiksis Technologies Oy:n palvelimelle luotiin opinnäytetyöprojektia varten oma ympäristö, jossa sain jatkaa työn loppuun. Opinnäytetyön määrittelyn mukainen järjestelmä kehitettiin loppuun.

6.3.6 Testaus

Testaus aloitettiin järjestelmän toteutuksen loppuvaiheessa, syyskuussa 2009. Ohjausryhmässä päätettiin, että järjestelmälle tehdään toiminnallisuustestaus ja käytettävyydestestaus. Näistä kahdesta toiminnallisuustesti suoritettiin opinnäytetyön ohessa, ja käytettävyydestesti suoritettiin myöhemmin opinnäytetyönulkopuolisena työnä. Toiminnallisuustestin suorittaminen jätettiin minun tehtäväkseni. Laadin testaukselle aluksi testaussuunnitelman, testitapaukset ja alustavan pöytäkirjan testaustilaisuutta ja tulosten analysointia varten.

Toiminnallisuustestaus suoritettiin 24.9.2009 järjestelmän toteutuksen päätyttyä samoissa olosuhteissa, joissa järjestelmä oli rakennettukin. Testaajiksi testaus-tilaisuuden valitsin yhden henkilön, jolla ei ollut aiempaa kokemusta Sinseweb-järjestelmän käytöstä. Testaaja valittiin satunnaisesti.

Toiminnallisuustestin testaustilaisuudessa varattiin tunti aikaa suoritua ennalta määritellyistä tehtävistä. Testaajan toimia seurattiin ja sattuneista virheistä pidettiin samalla kirjaa. Virheet kirjattiin toiminnallisuustestauksen testauspöytäkirjaan.

Toiminnallisuustestaus sujui hyvin ja järjestelmästä löytyi joitakin virheitä. Virheistä osa oli korjattavissa, mutta järjestelmään jäi virheitä, joiden korjaaminen olisi ollut hankalaa. Osa korjaamattomista virheistä johtui Drupalin moduuleiden vajaavaisuudesta. Toiminnallisuustestauksen lopputulos oli virheistä huolimatta hyvä. Testaajan mielestä järjestelmä olisi saanut olla nopeampi ja sanasto olisi saanut olla laajempi. Myös hakutulokset olisivat saaneet olla enemmän hakusanaan täsmäviä.

6.3.7 Projektin lopetus

Lokakuussa 2009 järjestelmä julistettiin valmiiksi. Se oli läpäissyt kaikki testit ja kaikki korjattavissa olleet virheet oli korjattu. Projektin päättyessä tehtiin koko projektista loppuraportti ja pidettiin projektin päättävä kokous kaikkien projektin

osapuolten kanssa. Viimeisessä kokouksessa katselmoitiin loppuraportti ja katseltiin projektin aikana syntynyttä tuotetta.

7 SINSEWEB – JÄRJESTELMÄN ESITTELY

Tässä luvussa esitellään Sinseweb-järjestelmä. Luvussa käsitellään sen fyysistä rakennetta ja toiminnallisuutta kuvien ja kaavioiden avulla.

7.1 Rakenne

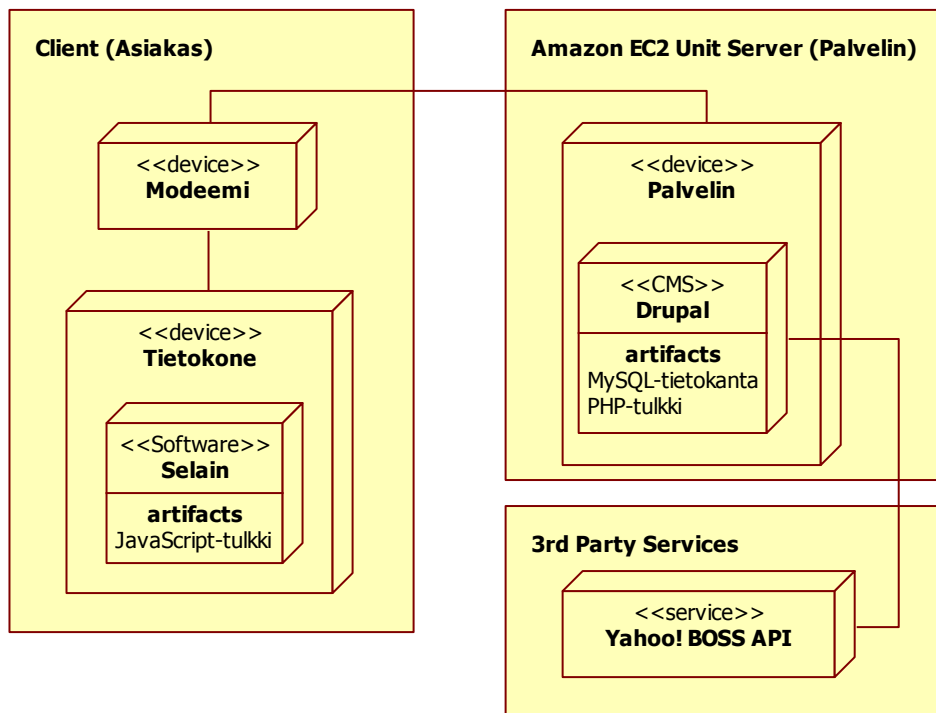
7.1.1 Fyysinen rakenne

Sinseweb-järjestelmä on sijoitettu Tiksis Technologies Oy:n vuokraamalle palvelimelle. Sitä käytetään selaimen avulla osoitteessa <http://unelmakone.tiksis.fi/sinse>. Sinseweb-järjestelmää käytetään sen graafisen käyttöliittymän (UI) avulla. Selain välittää viestin käyttäjän suorittamasta toimenpiteistä palvelimelle. Sinseweb-järjestelmän PHP:llä toteutetut lähdekooditiedostot sijaitsevat palvelimella. PHP-ohjelmakoodia sisältävät tiedostojen suorittaminen delegoidaan PHP-tulkille. PHP-tulkin tehtävä on lukea PHP-tiedostot ja tulkita niitä, jotta se voi pyytää järjestelmää suorittamaan tiedostoihin kirjatut toimenpiteet. Sinseweb-järjestelmän PHP-tiedostot sisältävät tavallisten toimenpiteiden (kuten tiedon käsittely ja laskutoimitusten tekeminen) lisäksi myös tietokantakyselyiden tekemisen ja Yahoo!BOSS API-palvelun käytön.

Sen jälkeen, kun PHP-tulkki on suorittanut toimenpiteet, PHP-tulkki käsittelee toimenpiteiden lopputulokset ja generoi HTML-sivun, jolla toimenpiteiden tulokset on esitettävissä. Jos toimenpide on ollut esimerkiksi tiedon hakeminen Yahoo! BOSS API-palvelun avulla generoi PHP-tulkki sivun, jolla esitetään hakutulokset. Lopuksi HTML-sivu palautetaan selaimelle, joka esittää sen käyttäjälle Sinseweb-hakukoneen käyttöliittymässä. Tässä tapauksessa käyttöliittymä joudutaan lataamaan aina uudelleen, jotta selain pystyy esittämään tulokset käyttäjälle.

Kuten aiemmin on jo kerrottu, Sinseweb-järjestelmä ei ole toteutettu pelkästään PHP- ja HTML-kielillä. Toteuttamiseen on käytetty myös JavaScript-kieltä

ja sen jQuery-kirjastoa. JavaScript ja jQuery-kirjastoa suoritetaan selaimessa selaimen JavaScript-tulkin avulla. Kaikki JavaScriptillä ja jQueryllä toteutetut toimenpiteet suoritetaan käyttäjän selaimessa. Tulkki tulkitsee JavaScript-tiedostot selaimessa ja suorittaa tiedostoissa määritellyt toimenpiteet. Lopuksi tulkki välittää selaimelle PHP-tulkin tapaan lopputulokset, jotka selain esittää käyttäjälle Sinseweb-järjestelmän käyttöliittymällä. Kuvassa 7.1 on esitetty järjestelmän rakenne edellisten kuvausten mukaisesti.



Kuva 7.1 Fyysinen rakenne.

Sinseweb-järjestelmässä käytetään myös edellämainittujen ohjelmointikielten lisäksi Ajax-tekniikkaa. Ajax-tekniikkaa hyödynnetään Sinseweb-järjestelmässä lähinnä sellaisissa tehtävissä, joissa käyttöliittymä pitää pystyä päivittämään ilman koko sivun uudelleen lataamista. Tällaisia tehtäviä ovat esimerkiksi suosikin lisäys ja kommentin tallennus järjestelmään.

7.1.2 Järjestelmän rakenne

Sinseweb-järjestelmä on Drupalin avulla rakennettu järjestelmä. Osana Sinseweb-järjestelmää on myös sen oma Sinseweb-moduuli. Sinseweb-moduuli sisältää Sinseweb-järjestelmän toiminnallisuuden. Se käyttää hyväkseen muita Drupalin Drigg, Taxonomy, Comment, Profile ja Favorite Nodes -moduuleita. Sinseweb-järjestelmän rakentamisessa on käytetty myös Drupalin Form API-moduulia, mutta ei samalla tavalla kuin muita moduuleita. Form API-moduulia käytettiin Sinseweb-järjestelmän web-sivujen luomiseen (kuten esimerkiksi Hakuasetukset-sivu).

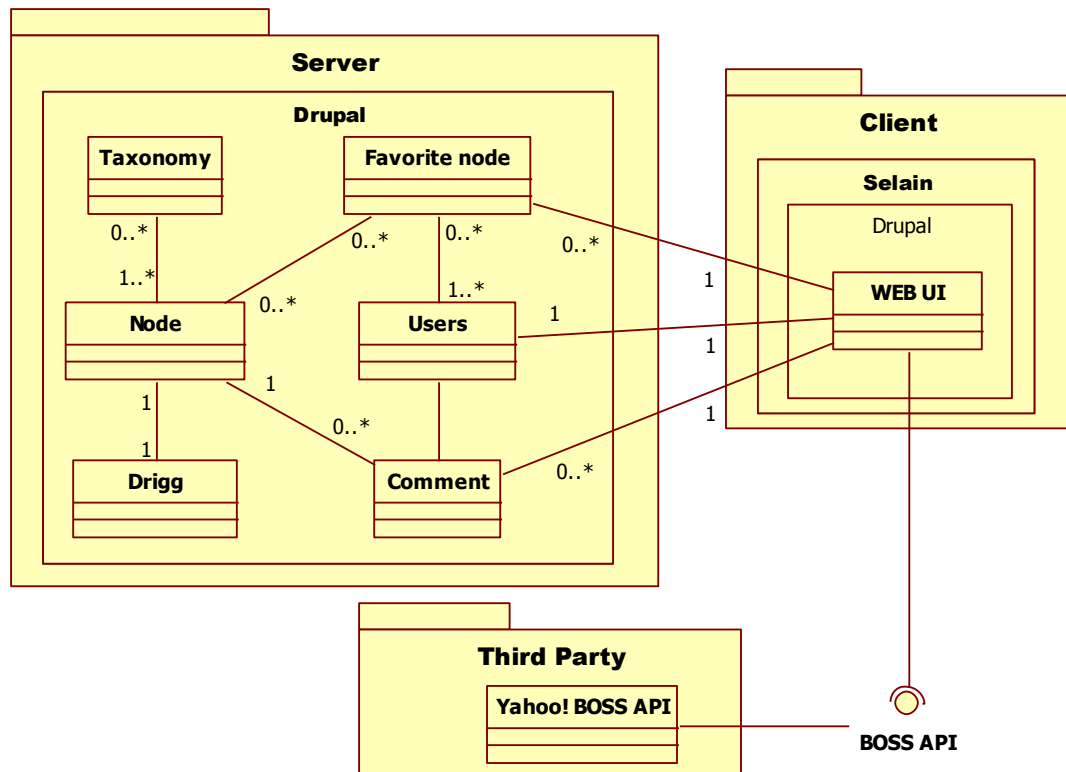
Muita edellä mainittuja moduuleita käytetään Sinseweb-järjestelmän oleellisten toimintojen toteuttamiseen. Drigg-moduulilla on laajennettu Drupalia siten, että hakutulokset voidaan tarpeen mukaan tallentaa järjestelmään linkkeinä. Taxonomy-moduulia on käytetty tietomallissa ja semantiikan luomisessa. Taxonomy-moduuli sisältää tietorakenteita, joiden avulla on mahdollista muodostaa ontologioita. Sinseweb-moduulin tietomallissa Taxonomy-moduulin tietomallia laajennettiin järjestelmään sopivaksi. Comment-moduulia käytetään Sinseweb-järjestelmässä käyttäjäkommenttien tallentamiseen ja esittämiseen. Profile-moduulin tehtävänä järjestelmässä on yksilöidä käyttäjät, ja Favorite Nodes-moduulin avulla tallennetaan järjestelmään suosikit.

Moduuleidensa lisäksi Drupal tarjoaa myös MySQL-tietokannan järjestelmälle. Kuten edellä on jo mainittu, tietokantaa on laajennettu järjestelmälle sopivaksi. Tietokanta kokonaisuudessaan ja laajennuksineen on kuvattu tarkemmin toiminnallisenmäärittelyn (liite 1) luvussa 3.

Drupalin ja sen moduulien lisäksi Sinseweb-järjestelmä käyttää myös Yahoo! BOSS API-palvelua. Yahoo! BOSS API -komponentti on Yahoo-hakupalvelun tuottama hakumoottorikomponentti, jonka avulla kehittäjät voivat itse laatia helposti hakukoneita. Yahoo! BOSS API tarjoaa kehittäjille Yahoo-hakutuloksia ja Yahoo-hakumoottorin palveluita, kuten esimerkiksi vaihtoehtoisten hakusanojen ehdottelu. Sinseweb-projektissa Yahoo! BOSS API:a hyödynnetään vain hakumoottorina, joka hakee tietyillä ehdoilla järjestelmälle hakutuloksia. Yahoo!

BOSS API-palvelua käytetään sen rajapinnan kautta. Palvelun antamat tulokset indeksoidaan ja muotoillaan tulosjoukoksi Sinseweb-järjestelmässä tulosten esittämistä varten.

Seuraavassa kuvassa 7.2 on esitetty järjestelmän rakenne.



Kuva 7.2. Järjestelmän rakenne.

7.2 Toiminta

Tässä luvussa on tarkoituksena esitellä järjestelmän toiminta ja sen perustoiminnot. Luvussa esitellyt toiminnot on kuvattu tarkemmin toiminnallisen määrittelyn (liite 1) luvussa 6.

7.2.1 Hakeminen

Sinseweb-järjestelmän keskeisin lomake on Haku-lomake. Haku-lomakkeella käyttäjä voi suorittaa tiedonhakemista internetistä. Haku-lomakkeella on hakemiseen tarvittavat kentät ja linkit toisille lomakkeille. Haku-lomakkeelta voi siirtyä seuraaville lomakkeille ja sivuille: Hakuasetukset-lomake ja Ohje-sivulle.

Haku-lomakkeella voidaan hakea uutisia (News) ja tietoa (Web) internetistä. Lomakkeella on heti haku-kentän ja haku-painikkeen alla aina kaksi linkkiä, joita painamalla voidaan milloin vain muuttaa hakuaineiston luonnetta uutisista perinteiseen tiedon hakuun ja toisin päin.

Tiedonhakeminen on jaettu Sinseweb-projektissa kolmeen toimenpiteeseen: hakutavoitteen määrittely, haun tekeminen ja hakutulosten katselu ja selaus. Kaikki nämä kolme toimenpidettä suoritetaan Haku-lomakkeella.

7.2.2 Hakutavoitteen määrittäminen

Hakutavoitteen määrittämisellä tarkoitetaan Sinseweb-järjestelmässä käyttäjän oman tavoitteen kirjoittamista Haku-kenttään, kuten alla olevassa kuvassa tehdään (ks. Kuva 7.3).



The image shows a search interface. At the top, there is a search input field containing the text "geneva". To the right of the input field is a button labeled "Haku". Further to the right are two links: "[Asetukset]" and "[Ohje]". Below the search input field is a dropdown menu with a light blue background. The menu is divided into two sections: "Templatelinks" and "Tasks". Under "Templatelinks", there are several search suggestions: "find education", "find housing", "find shopping center", "get education", "learn about administration", "learn about education", "learn about history", "learn about housing", "learn about shopping center", and "use transportation". Under "Tasks", there is one suggestion: "learn about". In the center of the dropdown menu, the text "Web | News" is displayed, indicating the current search scope.

Kuva 7.3. Hakutavoitteen määrittäminen.

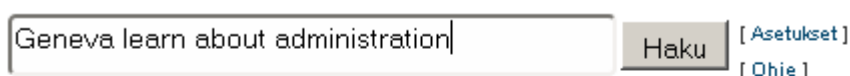
Käyttäjän tullessa Haku-lomakkeelle, hän syöttää käsitteen (tekemisen kohde) tai tavoitesanan (tekeminen) hakukenttään. Hakutavoite-kenttä tunnistaa sanan ja hakee sanalle joukon sanaan liittyviä malleja ja tavoittesanoja tai käsitteitä palvelimella sijaitsevasta tietokannasta. Tietokanta palauttaa järjestelmälle tulosjoukon, joka koostuu malleista ja käsitteistä tai tavoittesanoista. Tämän jälkeen tulosjoukkolista tulostetaan käyttäjälle, kuten kuvassa 7.3 on tehty. Listalla esiintyy kahdenlaisia ehdotuksia, joista ensimmäisiä ovat templatelinks eli mallit. Mallit ovat suoria ehdotuksia siitä, mitä käyttäjä voisi haluta etsiä kirjoittamallaan sanalla. Toiset listalla esiintyvät ehdotukset ovat kirjoitettuun sanaan liittyvät tavoittesanat (Tasks) tai käsitteet (Things). Kuvassa 7.3 järjestelmä ehdottaa käyttäjälleen tavoitteita.

Käyttäjä voi valita listalta yhden vaihtoehdon tai kirjoittaa sanan itse ja suorittaa saman iteraation uudelleen alusta loppuun, kunnes hakutavoite on saatu määritettyä. Toistaiseksi hakusanan määrittelyn ja hakemisen yhteydessä ei tallenneta uusia tuntemattomia tavoitteita ja käsitteitä järjestelmään. Tuntemattomien mallien, käsitteiden ja tavoitteiden esiintymistä on pyritty osittain välttämään syöttämällä järjestelmään sen kehityksen aikana niin paljon malleja, käsitteitä ja tavoitteita kuin mahdollista.

Hakutavoitteen määrittelyn jälkeen siirrytään haun suorittamiseen.

7.2.3 Haun tekeminen

Kuten missä tahansa hakukoneessa, suoritetaan tässäkin hakukoneessa hakeminen hakukentän perässä olevan Haku-painikkeen avulla. Kuvassa 7.4 on kuvattu hakulomake, jolla hakeminen tehdään.



Geneva learn about administration | Haku [Asetukset]
[Ohje]

Web | News

Kuva 7.4. Haun tekeminen.

Hakutavoitteen määrittelyn jälkeen käyttäjä painaa Hae-painiketta ja järjestelmä muodostaa hakutavoitteesta hakustrategian. Tämän jälkeen hakustrategialla haetaan tuloksia kahdesta paikasta. Ensimmäinen on Yahoo! BOSS API-hakukonemoottori ja toinen palvelimella oleva tietokanta, joka sisältää käyttäjien suosikkeja ja muita linkkejä. Molemmat lähteet palauttavat tulosjoukon, joka käsitellään. Käsitellyssä tulosjoukossa palvelimen tietokannasta haetut tulokset sijoitetaan ensimmäisiksi tulosjoukkoon ja Yahoo! BOSS API -palvelun palauttama tulosjoukko tulee tämän jälkeen. Yahoo! BOSS API -tulosjoukosta karsitaan pois Sinseweb-järjestelmän tietokannasta löytyneet tulokset.

Lopuksi järjestelmä tulostaa hakutulosityoukon käyttäjälle.

7.2.4 Hakutuloksen katselu ja selaus

Hakemisen jälkeen tulostuu käyttäjälle tulosjoukko näytölle (kuva 7.5). Hakutuloksessa on eroteltu Sinseweb-järjestelmästä löydetty tulokset ja Yahoo! BOSS API -palvelun kautta saadut tulokset. Koska Sinseweb-järjestelmään syötetyt tulokset ovat käyttäjien suosikkeja, on ne merkitty keltaisella tähdellä, joka on hakutuloksen oikealla puolella keltaisen nuolen vieressä. Muut tulokset, jotka saadaan Yahoo! BOSS API -palvelun kautta, on merkitty keltaisen tähden sijaan harmaalla tähdellä. Tämän lisäksi käytetystä hakulauseesta on muodostettu hakukentän alle leivänmurupolku, jolla voidaan mennä taaksepäin hakutavoitteen määrittelyssä. Esimerkiksi painamalla (kuva 7.5) leivänmurupolun linkkiä geneva suoritetaan haku hakusanalla geneva. Leivänmurupolun ja hakutulosten lisäksi käyttöliittymällä on myös ilmoitettu vihreässä laatikossa, kuinka tuottava haku oli.

geneva learn about administration

Haku

[Asetukset]

[Ohje]

geneva >> learn about administration

Web | News

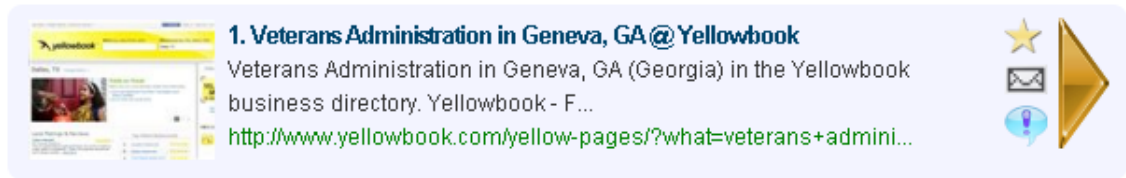
Showing results 1 - 50 from a total of 924433 results.

-  **1. Veterans Administration in Geneva, GA @ Yellowbook**
Veterans Administration in Geneva, GA (Georgia) in the Yellowbook business directory. Yellowbook - F...
<http://www.yellowbook.com/yellow-pages/?what=veterans+admini...>
-  **2. Public Health Program Administration in Geneva, KY**
Searching for Public Health Program Administration in Geneva, KY? Find Public Health Program Adminis...
<http://www.justclicklocal.com/citydir/geneva-ky--public-heal...>
-  **3. Print this page.**
Geneva College's Business Administration major prepares students with a broad training in the variou...
http://www.geneva.edu/object/business_administration?_templa...
-  **4. The Geneva School :: Administration Team**
At Geneva we seek to recover the lost art of teaching and the lost tools of learning and to instill ...
http://www.genevaschool.org/aboutus_administration.html

Kuva 7.5. Hakutulostista.

Hakutuloksen tulostamisen jälkeen käyttäjä voi katsella ja selaila hakutulosta näytöllään. Hakutulosta ei ole sivutettu näytölle vaan oletuksena hakutuloksia tulostetaan sivulle 50 kappaletta. Hakutulossivulta käyttäjä voi myös siirtyä myös suoraan linkin sivulle, lisätä linkin suosikkeihin (harmaa tähti), muokata suosikkilinkkiä (keltainen tähti), kommentoida linkkiä (puhekupla), lähettää linkin sähköpostitse toiselle henkilölle (kirjekuori) tai siirtyä tekemään uutta hakua.

Sinseweb-järjestelmässä hakutulos ei ole samanlainen kuin tyyppillisen hakukoneen tulos. Kuvassa 7.6 on esitelty Sinse-hakukoneen hakutulos.



Kuva 7.6. Hakutuloks.

Sinse-hakukoneen hakutuloksessa on kuva ja paljon kuvakkeita. Kuvakkeista jokainen toimii painikkeena ja niiden avulla voi suorittaa hakutulokseen liittyviä edellä mainittuja toimenpiteitä. Hakutuloksen otsikko, kuva ja nuoli ohjaavat klikattaessa hakutuloksen sivulle, joka avautuu selaimen asetuksista riippuen uudessa ikkunassa tai välilehdessä. Kun vie hiiren ison nuolen päälle, avautuu hakutuloksen vierelle uusi ikkuna, jossa on pidennetty kuvaus hakutuloksesta ja käyttäjien kommentit linkistä.

Linkin kommentointi

Linkkien kommentointi tehdään avaamalla näytölle yksinkertainen pieni lomake, johon voi käyttäjä kirjoittaa linkkiin liittyvät kommenttinsa. Lomake voidaan avata painamalla puhekuplakuvaketta. Kommentti kirjoitetaan lomakkeella oleviin kenttiin ja tallentaa lopuksi painamalla lomakkeen alareunassa olevaa Save-painiketta. Tämän jälkeen järjestelmä tallentaa kommentin tietokantaansa kytkien sen samalla automaattisesti linkkiin. Tallennuksen jälkeen järjestelmä ilmoittaa käyttäjälle tallennuksen onnistumisesta tai epäonnistumisesta. Lomake voidaan sulkea painamalla uudelleen puhekuplaikonia tai toista hakutuloksessa lomakkeen avaavaa painiketta (keltaista tähteä tai kirjekuorta).

Lähetä linkki kaverille

Linkin lähettäminen kaverille tehdään käyttäen yksinkertaista web-lomaketta, johon syötetään vastaanottajan tiedot, viestin sisältö ja otsikko. Lomake voidaan avata painamalla kirjekuorikuvaketta. Sähköposti viestin, osoitteen ja otsikon kirjoituksen jälkeen painamalla Send-painiketta voidaan avata asiakaspuolen sähköpostiohjelma (esimerkiksi Outlook), jossa on luotu syötetyistä tiedoista

valmis viestipohja. Tämän jälkeen viesti voidaan lähettää käyttäjän sähköpostiohjelman kautta vastaanottajalle. Sinseweb-järjestelmän oma viestinkirjoituslomake voidaan sulkea painamalla uudelleen kirjekuori-ikonia tai toista hakutuloksessa lomakkeen avaavaa painiketta (keltaista tähteä tai puhekuplaa).

Linkin lisäys suosikkeihin ja sen käsittely

Linkin lisääminen suosikkeihin on toimenpide, joka suoritetaan välittömästi harmaan Tähti-painikkeen painamisen jälkeen. Samalla linkki lisätään järjestelmään, jos sitä ei vielä ole siellä. Linkkiin lisätään tässä yhteydessä myös sen löytymiseen liittyviä hakusanoja hakutavoitelauseesta, jotta se löytyisi myös myöhemmin vastaavilla hakusanoilla haettaessa. Suosikin lisäyksen jälkeen kuvake muuttuu lisäyksen ajaksi oranssiksi tähdeksi ja lisäyksen onnistuttua keltaiseksi tähdeksi.

Suosikeista poistaminen tapahtuu painamalla keltaiseksi muuttunutta Tähti-painiketta uudelleen. Tämä avaa lomakkeen, jolla voidaan suosikin poistamisen lisäksi myös muokata suosikin tietoja. Poistaminen tapahtuu painamalla lomakkeella olevaa rasti-painiketta. Suosikkia voidaan myös muokata edellä mainitulla lomakkeella. Lomakkeella tehdyt muutokset voidaan tallentaa painamalla lomakkeen Save-painiketta. Lomake voidaan sulkea painamalla uudelleen keltaista tähteä tai toista hakutuloksessa lomakkeen avaavaa painiketta (puhekuplaa tai kirjekuorta).

7.2.5 Hakuasetusten määrittäminen

Hakuasetukset-lomakkeelle voi siirtyä hakulomakkeelta painamalla Haku-painikkeen vieressä olevaa Asetukset-linkkiä.

Hakuasetukset-lomakkeella käyttäjä voi määrittellä hakemiseen liittyviä asetuksia, kuten hakutulosten alueen ja kielen. Tämän lisäksi käyttäjä voi muuttaa lomakkeella hakutulosten oletusmäärää tai suodattaa hakutuloksia Safe Search – ominaisuuden avulla (joka poistaa hakutuloksista sopimattomat linkit). Hakuase-

tukset sivulla Hakutuloksen suodattavat ominaisuudet on ensin napautettava päälle ja vasta sen jälkeen on tehtävä valinnat.

Settings page

Results displayed:

50

▼ Filter search results

With safe search

By selected region

Regions:

Argentina

By selected language(s)

- | | | | | |
|------------------------------------|------------------------------------|-------------------------------------|----------------------------------|-----------------------------------|
| <input type="checkbox"/> katalaani | <input type="checkbox"/> Chinese | <input type="checkbox"/> tšekki | <input type="checkbox"/> tanska | <input type="checkbox"/> hollanti |
| <input type="checkbox"/> englanti | <input type="checkbox"/> suomi | <input type="checkbox"/> ranska | <input type="checkbox"/> saksa | <input type="checkbox"/> unkari |
| <input type="checkbox"/> heprea | <input type="checkbox"/> indonesia | <input type="checkbox"/> italia | <input type="checkbox"/> japani | <input type="checkbox"/> korea |
| <input type="checkbox"/> malaiji | <input type="checkbox"/> norja | <input type="checkbox"/> Portuguese | <input type="checkbox"/> romania | <input type="checkbox"/> venäjä |
| <input type="checkbox"/> espanja | <input type="checkbox"/> ruotsi | <input type="checkbox"/> thai | <input type="checkbox"/> turkki | <input type="checkbox"/> Vietnam |

Tallenna asetukset

Kuva 7.7. Hakuasetusten määrittäminen.

Hakuasetusten määrittäminen lomakkeelle siirtymisen jälkeen käyttäjä voi muuttaa hakuasetuksia hakuun sopivammaksi. Lopuksi käyttäjä tallentaa tehdyt muutokset järjestelmään ja siirtyy takaisin tekemään esimerkiksi uutta hakua.

8 YHTEENVETO

Opinnäytetyöni tuloksena syntyi ensimmäinen toimiva versio Sinseweb-järjestelmästä. Sinseweb-järjestelmä osaa tulkita semanttisen tietomallin ansiosta kaikki järjestelmään kirjatut mallit, yksittäiset sanat sekä semanttisen avusteen avulla määritellyt hakulauseet. Tämän ansiosta järjestelmän hakukone pystyy hakemaan käyttäjän määrittelemään hakulauseeseen sopivaa tietoa internetistä.

Sinseweb-järjestelmä ei ole kuitenkaan vielä valmis tuote, joka voidaan julkaista internetissä kaikkien käyttöön. Opinnäytetyöni tuloksia hyödynnetään uuden järjestelmän kehittämisessä. Asiakkaan toiveiden mukaisesti järjestelmän testusta jatketaan opinnäytetyön päätyttyä, jotta Sinse-konseptia voidaan testata paremmin.

Opinnäytetyön aikana Sinseweb-järjestelmä saatiin keskeisiltä osilta määriteltyä, suunniteltua, toteutettua ja testattua. Projektin ohjaus ja suunnittelu lukuun ottamatta aikataulujen laatimista onnistui hyvin. Projektin aikataulujen muodossa tehty tehtäviin kuluva ajankäytön arviointi oli ylioptimistinen. Tämän lisäksi projektin aikana toteutuksen loppuvaiheilla aikataulut joutuivat venymään, koska työtuntejani vähennettiin huomattavasti. Myös opinnäytetyötäni varten tehdyn oman ympäristön toteuttaminen vei useita viikkoja, jonka aikana työ ei edennyt yhtään. Näistä syistä aikatauluja saatiin korjailta ja uudelleen arvioida projektin loppuvaiheessa enemmän.

Aikataulujen lisäksi myös tehtäviä oli liikaa opinnäytetyöhön varattuun aikamäärään nähden. Lopussa tehtäviä jouduttiin karsimaan ja kaikkia ominaisuuksia ei saatu toteutettua lainkaan. Näitä ominaisuuksia olivat muun muassa kuvahaku, uusien mallien muodostaminen hakusanan määrittelyn yhteydessä ja hakujen perusteella tapahtunut laajempi käyttäjäprofiilin täydennys hakuihin liittyvällä tiedolla (esimerkiksi tilastoimalla käyttäjän yleisimmin käyttämiä hakusanoja hakulauseissa).

Projektissa onnistuttiin lopputuotteen suunnittelun aikana luomaan kelpo tietomalli, joka on helposti laajennettavissa. Tietomallin avulla pystytään ratkomaan suurin osa semantiikan ongelmista Sinseweb-järjestelmästä ja nopeuttamaan hakusanan määrittelyä huomattavasti. Suunnitteluvaiheen aikana tehtiin myös muutamia virheitä. Drupalin valitseminen toteutustyökaluksi oli yksi suurimmista virheistä. Drupal aiheutti melkein yhtä paljon ongelmia kuin hyötyä. Drupalin hyviin puoliin kuului se, että lomakkeiden ja muiden yksinkertaisten asioiden luominen oli helppoa ja hyvin organisoitua. Drupalin huonoja puolia olivat puolestaan sen joustamattomuus ja tiukat standardit. Nämä rajoittivat järjestelmän kehitystä jonkin verran ja aikaa kului enemmän monimutkaisempien ratkaisujen keksimiseen. Drupalissa ja sen moduuleissa oli myös paljon virheitä, jotka aiheuttavat vieläkin Sinseweb-järjestelmässä ongelmia. Näistä kaikista Drupaliin liittyvistä ongelmista onkin projektin aikana tultu siihen tulokseen, että Drupal ei sovi vastaavanlaisten järjestelmien kehittämiseen.

Projektin lopputuotteen toteutuksen aikana myös puutteellinen versioiden hallinta ja kommunikointi aiheutti joitakin ongelmia. Alustavasti projektin toteutuksen alkaessa käytössä oli versionhallintatyökalu, joka ei kuitenkaan toiminut oikeassa käytössä. Tämän jälkeen versionhallintaa hoidettiin ottamalla vähintään kerran viikossa kopiot järjestelmän tiedostoista toiseen paikkaan. Tämän lisäksi myös palvelimella oli ohjelma, joka otti palvelimen sisällöstä kopion sopivin väliajoin.

Projektin lopputuotteen testausvaiheessa huomattiin joitain uusia virheitä. Testauksessa löytyneistä virheistä suurin osa johtui Drupalista ja sen moduuleista. Näitä virheitä ei voitu korjata. Loput virheistä olivat itse aiheutettuja ohjelmointivirheitä, jotka olivat korjattavissa. Virheiden lisäksi testauksessa huomattiin, miten paljon lisäarvoa tietyt järjestelmän ominaisuudet toivat järjestelmälle. Samalla saatiin myös positiivista palautetta testaaajilta järjestelmästä. Opinnäytetyöhön kuuluvassa toiminnallisuustestauksessa saatiin selville vähän järjestelmän hyödyistä, mutta opinnäytetyön ulkopuolisessa käytettävyydestä saatiin paljon enemmän positiivisia tuloksia järjestelmästä.

Käytettävyydestä aikana huomattiin, että käyttöliittymän ulkoasu, hakukoneeseen laaditut lisätoiminnot (kuten suosikin lisäys, kommentointi ja kaverille lähettäminen) ja hakutuloksen kuvat saivat huomattavasti positiivista palautetta. Vastaavasti negatiivista palautetta saatiin lähinnä järjestelmän hitaudesta ja virheistä. Myös hakutulokset olisivat voineet olla joidenkin mielestä tarkempia.

Kaikista järjestelmä kehittämiseen liittyvistä ongelmista ja epäkohdista huolimatta projektin aikana opin paljon ryhmätyöskentelystä ja Drupalista. Suurimpana onnistumisena pidän Drupalin käytön oppimisen. Ryhmätyöskentelystä olen oppinut kommunikoinnissa sattuneiden virheiden ja epäonnistumisten kautta. Myös testausprosessista ja sen vaiheista olen oppinut paljon, koska projektin aikana olen lukenut jonkin verran aiheeseen liittyvää kirjallisuutta. Näiden lisäksi projektin toteutuksen aikana tutustuin jQuery-kirjastoon sen dokumentaation ja käytön kautta. Samalla syvensin myös aikaisempaa PHP- ja JavaScript-ohjelmointikielen osaamistani.

Kaikkien näiden työkalujen ja menetelmien oppimisen lisäksi olen myös tutustunut projektini aikana semantiikkaan ja sen käyttöön tietotekniikassa. Oppimani perusteella uskonkin, että semantiikka on luultavasti oleellinen osa tulevaisuuden tietojärjestelmiä. Tämän opinnäytetyön aikana tutuksi tulleista tekniikoista on varmasti minulle hyötyä tulevaisuudessa tällä alalla.

Sinseweb-järjestelmän kehitys ei pääty tähän. Uskon vahvasti, että järjestelmästä voidaan kehittää julkaisukelpoinen tuote, jos ensinnäkin toteutetaan kaikki ne ominaisuudet, jotka on tässä opinnäytetyössä jäi toteuttamatta. Tämän lisäksi parantamalla järjestelmän ominaisuuksia voidaan saavuttaa järjestelmälle paljon lisäarvoa.

KUVAT

- Kuva 2.1 Verkkotunnusten määrä vuoden 2008 toiselle neljännekselle (Q2') saakka, s. 13
- Kuva 2.2 .com ja .net -päätteisten verkkotunnusten sivumäärät, s. 14
- Kuva 2.3 Makupalat-aineistohakemisto, s. 15
- Kuva 2.4 Exite Travel –sivuston palvelinhakemistot, s. 16
- Kuva 2.5 Saima-tietokanta, s. 17
- Kuva 2.6 MSN-portaali, s. 18
- Kuva 2.7 Google-hakukone, s. 19
- Kuva 3.1 Semanttinen verkko, s. 26
- Kuva 4.1 Vesiputousmalli, s. 29
- Kuva 4.2 Prototyypimalli, s. 30
- Kuva 4.3 EVO-malli, s. 31
- Kuva 4.4 RUP, s. 32
- Kuva 4.5 Testauksen V-malli, s.36
- Kuva 5.1 Käsittemalli, s. 38
- Kuva 5.2 Luokkakaavio, s. 39
- Kuva 5.3 HTML-koodi ja sitä vastaava HTML-dokumentti, s. 41
- Kuva 5.4 Ajax-tekniikan toiminta, s. 43
- Kuva 5.5 SQL-kielen ohjelmakoodinäyte, s. 44
- Kuva 6.1 Projektioorganisaatio s. 52
- Kuva 7.1 Fyysinen rakenne, s. 62
- Kuva 7.2 Järjestelmän rakenne, s. 64
- Kuva 7.3 Hakutavoitteen määrittäminen, s. 65
- Kuva 7.4 Haun tekeminen, s. 66
- Kuva 7.5 Hakutulostulo s. 68
- Kuva 7.6 Hakutulos, s. 69
- Kuva 7.7 Hakuasetusten määrittäminen s. 71

LÄHTEET

Ala-Mutka, K. Rintala, M. Savikko V. & Palviainen, J. Tiedon salaaminen (kryptaus)

<http://www.cs.tut.fi/etaopetus/titepk/luku20/salaus.html>

(Luettu 28.5.2009)

Art Branch inc. 2004. SQL Tutorial.

<http://www.sql-tutorial.net/>

(Luettu 4.9.2009)

Christodorescu, M. & Jha, S. & Maughan, D. & Song, D. & Wang, C. 2007. Malware Detection. New York: Springer Science+Business Media, LLC.

DomainTools. 2009. Domain Counts & Internet Statistics.

<http://www.domaintools.com/internet-statistics/>

(Luettu 1.9.2009)

Drupal.org Comment. 2009. Comment: allow comments on content

<http://Drupal.org/handbook/modules/comment>

(4.9.2009)

Drupal.org Drigg. 2007. Drigg.

<http://Drupal.org/project/drigg>

(Luettu 4.9.2009)

Drupal.org Favorite Nodes. 2006. Favorite Nodes.

http://Drupal.org/project/favorite_nodes

(Luettu 4.9.2009)

Drupal.org Form API. 2006. Form API Quickstart Guide.

http://api.Drupal.org/api/Drupal/developer--topics--forms_api.html/4.7

(Luettu 4.9.2009)

Drupal.org. 2009. Profile: extending user account information.

<http://Drupal.org/handbook/modules/profile>

(Luettu 4.9.2009)

Drupal.org Taxonomy. 2009. Taxonomy: A way to organize content.

<http://Drupal.org/handbook/modules/taxonomy>

(Luettu 4.9.2009)

Groh, M. Stockman, J. Powell, G. & Prague, C. 2007. Access 2007 bible. Indianapolis: Wiley Publishing Inc.

Haas, J. Phpmyadmin.

<http://linux.about.com/cs/linux101/g/phpmyadmin.htm>

(Luettu 7.9.2009)

Haikala, I. & Märijärvi, J. 2004. Ohjelmistotuotanto. Hämeenlinna: Talentum Oyj.

Heilmann, C. 2006. Beginning JavaScript with DOM scripting and Ajax: from novice to professional. Berkeley: Apress.

Honkasaari, T. & Fontell, L. 1992. Tietokeskeisen suunnittelun menetelmiä. 1.-3. Painos. Helsinki: VAPK-kustannus.

Hovi, A. 2000. SQL-ohjelmointi Pro Training. Helsinki: Satku.

Hovi, A. Huotari J. & Lahdenmäki, T. 2005. Tietokantojen suunnittelu & indeksointi. Porvoo: Docendo.

HTML-advisor. 2009. What is jQuery?

<http://www.html-advisor.com/JavaScript/what-is-jquery/>

(Luettu 6.9.2009)

Ikonen, A. Kurki, E. Oinonen-Vallius, T. Kortelainen, J. Tuhkanen, H. 2004. Tiedonhaku Internetistä.

http://aikoledu.pkky.fi/matskunetti/materiaali/mto_materiaali_7.pdf

(Luettu 12.8.2009)

Jaakkola, M. 2000. www-selaimen ohjelmointi.

<http://edu.phkk.fi/Opiskelu/Internet-ohjelmointi/wwwSovOhj.htm#Java-ohjelmointikieli%20ja%20sovelmat>

(Luettu 29.6.2009)

Jakobsson, M. & Myers, S. Phising and Countermeasures. 2007. Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft. New Jersey: John Wiley & Sons, Inc.

JS. 2006. Pikajohdanto XML-skeemoihin.

<http://blite.iki.fi/artikkelit/pikajohdanto-xml-skeemoihin/>

(Luettu 29.6.2009)

Kautto, T. 1996. Ohjelmistotestaus ja siinä käytettävät työkalut.

<http://www.mit.jyu.fi/opiskelu/seminaarit/ohjelmistotekniikka/testaus/>

(Luettu 12.10.2009)

Kokkolan Kaupunginkirjasto. Tiedonhaku internetistä.

http://lib.kokkola.fi/kaveri/tiedonhaku/tiedonhaku_internetetista.html

(Luettu 12.8.2009)

Laaksonen, A. 2003. Käytännön PHP-opas (Osa 1 – Johdanto)

<http://www.ohjelmointiputka.net/opas.php?tunnus=phpj>

(Luettu 20.7.2009)

Laine, H. 2005. Tietokantojen perusteet.

<http://www.cs.helsinki.fi/u/laine/tkp/tietomallit/kasiteanalyysi.html>
(Luettu 9.10.2009)

Lilja, T. Graafin visualisoija.
<http://www.cs.hut.fi/Opinnot/T-93.210/2001/harjoitustyo/graphdraw.html>
(Luettu: 8.11.2010)

Linux.fi. 2009. Google Chrome.
http://linux.fi/wiki/Google_Chrome
(Luettu 7.9.2009)

Market Share. 2009. Browser Market Share.
<http://marketshare.hitslink.com/browser-market-share.aspx?qprid=0&qpmr=100&qpdt=1&qpct=3&qptimeframe=Y>
(Luettu 7.9.2009)

Meyer, E. 2006. CSS: Definitive Guide, Third Edition. Sebastopol: O'Reilly Media Inc.

Mikkonen, S. XML (Extensible Markup Language).
<http://aikoledu.pkky.fi/virtuaalikoulu/materiaali/XML.pdf>
(Luettu 27.5.2009)

Miles, R. & Hamilton, K. 2006. Learning UML 2.0. Sebastopol: O'Reilly Media Inc.

Muikku-Werner, P. & Savolainen, E. 1999. Selityksiä ja lisätietoja.
<http://www.internetetix.ofw.fi/opinnot/opintojaksot/8kieletkirjallisuus/pragmatiikka/selityks.html>
(Luettu 16.6.2009)

Multisilta, J. World Wide Web.
<http://matwww.ee.tut.fi/hmintro/www.html>
(Luettu 9.10.2009)

Mäkinen, K. 2008. ePortti luottaa ilmakuvienvetovoimaan.
www.maanmittauslaitos.fi/Tietoa_maasta_32008.pdf
(Luettu 27.5.2009)

Nikkilä, T. 2001. Internet 2001 - eilen, tänään ja huomenna. Jyväskylä: Docendo.

OAMK (MySQL). MySQL materiaali.
<http://www.ratol.fi/opensource/mysql/index.htm>
(Luettu 4.9.2009)

OAMK (Tiedonhaku). Tiedonhaku internetetistä.
<http://www.ratol.fi/opensource/tiedonhaku/>
(Luettu 12.8.2009)

- OBITIKO, M. 2007. SEMANTIC NETWORKS.
<http://www.obitko.com/tutorials/ontologies-semantic-web/semantic-networks.html>
(21.6.2009)
- OMG. 2009. Getting started with UML.
<http://www.uml.org/>
(Luettu 8.9.2009)
- Pohjonen, R. 2002. Tietojärjestelmien kehittäminen. Jyväskylä: Docendo.
- Profium. Ontologia.
<http://www.profium.com/index.php?id=672>
(Luettu 18.6.2009)
- Puskala, A. 2002. Tiedonhakustrategiat World Wide Webissä.
<http://ethesis.helsinki.fi/julkaisut/hum/psyko/pg/puskala/tiedonha.pdf>
(Luettu 24.8.2009)
- Pyhäjärvi, M. & Pöyhönen, E. 2004. Strategioista suunnitelmiin - selkeyttä käsitteiden sekamelskaan.
<http://www.pcuf.fi/sytyke/kerhot/testaus/arkisto/MaaretEki2004-09-06.ppt>
(Luettu 12.10.2009)
- Search Engine Partner. 2010. Search Engine (SEO) Glossary.
<http://www.searchenginepartner.com/SEO-glossary.html>
(Luettu 9.11.2010)
- SearchEnterpriseDesktop.com / Chuck Bury. 2008. Internet Explorer
http://searchenterprisedesktop.techtarget.com/sDefinition/0,,sid192_gci214105_00.html#
(Luettu 7.9.2009)
- Smartlogic. 2009. What we do and why it works.
<http://www.smartlogic.com/index.php/what-we-do-and-why-it-works>
(Luettu 12.8.2009)
- Sowa, J. 2006. Semantic Networks.
<http://www.jfsowa.com/pubs/semnet.htm>
(Luettu 18.6.2009)
- Suehring, S., Converse, T. & Park, J. 2009. PHP 6 and MySQL 6 Bible. Indianapolis: Wiley Publishing Inc.
- Tampereen maakuntakirjasto. Tiedonhaku internet-ympäristössä.
<http://www.tampere.fi/kirjasto/ansuhaku.htm>
(Luettu 12.8.2009)
- Tech-FAQ. 2009. What is a Web Browser?

<http://www.tech-faq.com/web-browser.shtml>

(Luettu 9.10.2009)

Tsui, F. & Karam, O. Essentials of Software engineering. Sudbury: Jones ad Barlett Publishers, LLC.

Tupala, T. 2005. Semanttiset hakukoneet.

www.it.lut.fi/kurssit/04-05/010850000/Tuomo_tupala.pdf

(Luettu 12.8.2009)

Vander Veer, E. 2005. JavaScript for dummies. Indianapolis: Wiley Publishing, Inc.

VanDyk, J. & Westgate, M. 2007. Pro Drupal Development. Berkley: Apress.

VeriSign. 2008. The Domain Name Industry Brief.

<http://www.verisign.com/static/044191.pdf>

(Luettu 1.9.2009)

Vim online. About Vim.

<http://www.vim.org/about.php>

(Luettu 7.9.2009)

Vohra, D. 2008. Ajax in Oracle JDeveloper. Berlin Heidelberg: Springer-Verlag.

Wikipedia (suomi) sisällönhallintajärjestelmät. 2009. Drupal.

<http://fi.wikipedia.org/wiki/Drupal>

(Luettu 20.7.2009)

Welling, L. & Thomson, L. 2003. PHP and MySQL Web development. USA: Sams Publishing.

Wikipedia HTML (englanti). 2009. HTML.

<http://en.wikipedia.org/wiki/HTML>

(Luettu 6.9.2009)

Wikipedia Internet search engines (englanti). 2009. Web search engines.

http://en.wikipedia.org/wiki/Search_engines

(Luettu 12.8.2009)

Wikipedia JavaScript libraries (englanti). jQuery. 2009.

<http://en.wikipedia.org/wiki/JQuery>

(Luettu 6.9.2009)

Wikipedia. 2009. Mozilla Firefox.

http://en.wikipedia.org/wiki/Mozilla_Firefox

(7.9.2009)

Wikipedia (suomi) Luokittelujärjestelmät. 2009. Ontologia (tietojenkäsittelytiede).

[http://fi.wikipedia.org/wiki/Ontologia_\(tietojenk%C3%A4sittelytiede\)](http://fi.wikipedia.org/wiki/Ontologia_(tietojenk%C3%A4sittelytiede))
(Luettu 18.6.2009)

Wikipedia (suomi) Ohjelmistotuotanto 2009a. Ohjelmistotuotanto.
<http://fi.wikipedia.org/wiki/Ohjelmistotuotanto>
(Luettu 3.9.2009)

Wikipedia (suomi) Ohjelmistotuotanto 2009b. RUP.
<http://fi.wikipedia.org/wiki/RUP>
(Luettu 12.10.2009)

Wikipedia (suomi) semantiikka. 2009. Semantiikka.
<http://fi.wikipedia.org/wiki/Semantiikka>
(9.10.2009)

Wikipedia (suomi) sisällönhallintajärjestelmät. 2009. Drupal.
<http://fi.wikipedia.org/wiki/Drupal>
(Luettu 20.7.2009)

Wikipedia (suomi) SQL. 2009. MySQL.
<http://fi.wikipedia.org/wiki/MySQL>
(Luettu 4.9.2009)

Wikipedia (suomi) Taksonomia. 2009. Taksonomia.
<http://fi.wikipedia.org/wiki/Taksonomia>
(Luettu 17.6.2009)

Wikipedia (suomi) tekoäly. 2009. Semanttinen verkko.
http://fi.wikipedia.org/wiki/Semanttinen_verkko
(Luettu 17.6.2009)

Yaratan, H. 2003. Intelligent Tutoring System: A Tool for Testing the Research
CURiosities of Artificial Intelligence Researchers.
<http://www.tojet.net/articles/236.htm>
(Luettu 18.6.2009)

Yahoo! Development network. Yahoo! Search BOSS.
<http://developer.yahoo.com/search/boss/>
(7.9.2009)

Zandstra, M. 2004. Sams teach yourself PHP in 24 hours. USA: Sams Pub-
lishing.