



TAMPEREEN TEKNILLINEN YLIOPISTO
Tietotekniikan koulutusohjelma

ANTTI MÄKI

MUKAUTUVAN VERKKOPALVELUN TIETOSISÄLLÖN
HALLINTA

Diplomityö

Tarkastajat: erikoistutkija Ossi Nykänen (TTY)
professori Seppo Pohjolainen (TTY)

Tarkastajat ja aihe hyväksytyt Tieto-
ja sähkötekniikan tiedekuntaneuvoston
kokouksessa 16.1.2008

Alkulause

Tämä diplomityö on tehty Tampereen teknillisen yliopiston Hypermedialaboratoriossa osana Aatu-hanketta. Kesäkuussa 2007 alkanut kirjoitustyö tuntui vielä kevään mittaankin uusien aiheeseen liittyvien kysymysten ilmetessä laajenevan lukemattomiin suuntiin. Jotta sain tutkintoni valmiiksi saattavan opinnäytteen lopulta kansiin, täytyi kirjoitustyö päättää hankkeen viimeistelyn ollessa kiireisimmillään. Lopputulokseen olen kuitenkin tyytyväinen.

Kiitän erikoistutkija Ossi Nykästä ja professori Seppo Pohjolaista työni ohjaamisesta ja tarkastamisesta sekä kaikkia lähipiirini jäseniä saamastani tuesta kuluneen vuoden aikana.

Tampereella 23.4.2008

Antti Mäki

Sisältö

Alkulause	ii
Sisältö	iii
Tiivistelmä	v
Abstract	vi
Lyhenteet	vii
Kuvat	viii
1 Johdanto	1
1.1 Määritelmät	2
1.2 Työn rakenne	4
2 Mukautuvat verkkopalvelut	6
2.1 Hypermedia ja hypermediajärjestelmät	6
2.2 Mukautuvat hypermediajärjestelmät	8
2.3 Mukautusprosessi	10
2.3.1 Tiedonkeruu	10
2.3.2 Käyttäjämallin rakentaminen	14
2.3.3 Mukautus käyttäjämallin perusteella	15
2.4 Web hypermediajärjestelmänä	18
2.4.1 Fyysinen ympäristö	19
2.4.2 Tekninen ympäristö	20
2.4.3 Sosiaalinen ympäristö	21
2.4.4 Mukautuminen käytännössä	22
3 Verkkopalvelun sisällönhallinta	24
3.1 Verkkopalvelun sisällönhallinnan kehitys	24
3.2 Blogit	26
3.3 Wikit	28
3.4 Web-pohjaisen käyttöliittymän erityispiirteet sisällönhallinnassa	31
3.4.1 Tekniset vahvuudet ja heikkoudet	31
3.4.2 Mahdollisuudet ja uhat sisällönhallinnalle	33
4 Tietomallin vaikutus sisällöntuotanto- ja mukauttamisprosesseihin	34
4.1 Tiedon mallintaminen	34
4.2 Olemassa olevat ruokaohjemallit	36
4.2.1 BigOven	38
4.2.2 RecipeML	41
4.2.3 Dish_Master	46
4.3 Mallintamisen vaikutus sisällöntuotantoon ja mukautusprosessiin	49

5	Aatu — Apua erikoisruokavalion hallintaan	51
5.1	Aatu-verkkopalvelun esittely	51
5.2	Käyttäjryhmät ja palvelun käytön tavoitteet	52
5.2.1	Ylläpitäjät	52
5.2.2	Sisällöntuottajat	53
5.2.3	Loppukäyttäjät	54
5.3	Toteutustekniikat	56
5.3.1	Django-verkkosovelluskehys	56
5.3.2	ECMAScript ja AJAX	59
5.4	Mukauttaminen Aatussa	59
5.5	Tiedon mallintaminen Aatussa	63
5.5.1	Lähtökohdat	63
5.5.2	Ratkaisut	64
5.6	Sisällönhallinta Aatussa	68
5.6.1	Ruokaohjeiden lisääminen palveluun	68
5.6.2	Ruokavalioprofilien lisääminen palveluun	74
5.6.3	Tietosisällön muokkaaminen	76
5.6.4	Julkaisutilojen hallinta	78
5.7	Verkkopalvelun tietosisältö ja tekijänoikeudet	81
6	Toteutuksen onnistumisen arviointi	83
6.1	Jatkokehitys	85
7	Yhteenveto	88
	Lähteet	90
A	Järjestelmän tietomalli	95

Tiivistelmä

TAMPEREEN TEKNILLINEN YLIOPISTO
Tietotekniikan koulutusohjelma

MÄKI, ANTTI:

Mukautuvan verkkopalvelun tietosisällön hallinta

Diplomityö, 94 sivua, 1 liitesivu
Kesäkuu 2008

Pääaine: Hypermedia

Tarkastajat: erikoistutkija Ossi Nykänen ja professori Seppo Pohjolainen

Avainsanat: mukautuva verkkopalvelu, tiedon mallintaminen, sisällönhallinta,
käyttöliittymä

Hypermediajärjestelmät ovat perinteisesti pyrkineet palvelemaan käyttäjiään mukauttamalla esittämäänsä sisältöä käyttäjän tarpeisiin sopivaksi. Internetissä toimiva hajautettu World Wide Web on paitsi tunnetuin hypermediajärjestelmä, myös globaali, kaikille avoin julkaisukanava, joka yhdistää valtavaa informaatiomäärää ja satoja miljoonia käyttäjiä, joiden tarpeet eroavat toisistaan. Tarve verkkopalveluiden tietosisältöjen mukauttamiselle on siis selkeä.

Mukautuvat verkkopalvelut perustuvat käyttäjästä luotavan mallin ja tarjolla olevan tietosisällön vertailuun. Tämä, kuten muukin informaation mekaaninen käsittely edellyttää, että sekä käyttäjämalli että verkkopalvelun tietosisältö ovat rakenteisessa muodossa. Mukautuvan verkkopalvelun tietomallia suunniteltaessa tuleekin huomioida rakenteisuuden aiheuttamat vaatimukset tietomallille.

Verkkopalvelun teknisestä toteuttamisesta ja tietosisällön hallinnasta vastaavat usein eri henkilöt. Koska tietosisällöstä vastaavat henkilöt eivät välttämättä tunne verkkopalveluiden toteutukseen liittyviä tekniikoita, syntyy tarve helppokäyttöisille sisällönhallintatyökaluille. Verkkopalvelun sisällöntuotantoon tarvittava työmäärä kasvaa entisestään, mikäli tietosisältö tulee kerätä mukautukseen soveltuvassa rakenteisessa muodossa. Sisällönhallintatyökalujen tulisi siis vähintään mahdollistaa, mutta mieluummin tukea rakenteisen tietosisällön tuottamista.

Tämä työ on kirjoitettu osana projektia, jossa toteuttiin verkkopalvelu erityisruokavaliota noudattaville henkilöille. Työ pyrkii kartoittamaan mukautuvien hypermediajärjestelmien, informaation mallintamisen ja sisällönhallinnan perusteita, sekä dokumentoimaan miten nämä on huomioitu verkkopalvelua toteutettaessa.

Abstract

TAMPERE UNIVERSITY OF TECHNOLOGY
Master's Degree Programme in Information Technology

MÄKI, ANTTI:
Content Management of an Adaptive Web Service

Master of Science Thesis, 94 pages, 1 appendix page
June 2008

Major: Hypermedia
Examiners: Senior Researcher Ossi Nykänen and Professor Seppo Pohjolainen
Keywords: adaptive web service, information modelling, content management,
 user interface

Hypermedia systems strive to better serve their users by adapting the presented content according to the users' needs. The most established hypermedia system, World Wide Web, is a global, open publishing venue which unites vast amounts of information with multitude of users with unique needs. Therefore adaptation is especially crucial for web services.

The adaptation process is based on selecting the most appropriate content for a model presenting the user. This — and other tasks involving electronic data processing — requires the information to be produced and stored in a structured format. This imposes specific requirements, which need to be considered while designing the information model of an adaptive web service.

Since it is a common practice to distribute the responsibilities of implementing the technical aspects of a web service and producing the content for it, it is customary for the content managers to be unacquainted with the underlying web technologies. As a result, user friendly tools that hide the implementation technologies are required for efficient content management. As producing the content in structured format makes the process more tedious, the given tools should not obstruct, but rather support producing structured content.

This thesis was written as a part of a project, in which we produced a web service aimed for people with dietary restrictions. In this thesis we study the basics of adaptive hypermedia systems, information modeling and content management. Furthermore, the thesis contains documentation on how these topics affected the implementation of the aforementioned web service.

Lyhenteet

Ajax	Asynchronous JavaScript And XML
CSS	Cascading Style Sheets
ECMA	European Computer Manufacturers Association
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
MTV	Model-Template-View
MVC	Model-View-Controller
SQL	Structured Query Language
USDA	United States Department of Agriculture
W3C	World Wide Web Consortium
WWW	World Wide Web
WYSIWYG	What You See Is What You Get
XML	Extensible Markup Language

Kuvat

4.1	BigOven-ohjelmiston näkymä ruokaohjeen muokkaamisesta	39
4.2	BigOven-ohjelmiston näkymä sopivan ruoka-aineen ehdottamisesta . .	40
4.3	Recipeml-elementti ja sen läheisimmät jälkeläiselementit.	42
4.4	Recipe-elementti ja sen läheisimmät jälkeläiselementit.	43
4.5	Ingredients-elementti ja sen läheisimmät jälkeläiselementit.	44
4.6	Ing-elementti ja sen läheisimmät jälkeläiselementit.	45
4.7	Esimerkki valmistusgraafista (mukailten [Wang & Li 2007])	47
5.1	Esimerkki muunnelman soveltumattomuuden korostamisesta.	62
5.2	Esimerkki vaiheistetun lomakkeen välilehtimetaforasta.	69
5.3	Esimerkki ehdottaja-järjestelmän toiminnasta.	73
5.4	Sallittujen tai kiellettyjen ruoka-aineiden valintaan käytettävä lomake	76

Luku 1

Johdanto

Tämä luku esittelee ja rajaa työn aihealueen. Lisäksi määritellään keskeisimmät käsitteet ja esitellään työn rakenne luvuittain.

Verkkopalvelut ovat arkipäiväistyneet informaation jakelukanavina. Globaalin luonteensa vuoksi verkkopalveluilla on ennen näkemättömän laaja lukijakunta. Eri käyttäjäryhmien tarpeet saattavat erota toisistaan merkittävästi, tai olla jopa keskenään ristiriitaisia. Tästä johtuen käyttäjäryhmiä voidaan palvella paremmin tarjoamalla juuri heidän tarpeisiinsa räätälöityjä toimintoja ja sisältöä. Tämä sisällön mukautuminen kuitenkin edellyttää, että verkkopalvelun tietosisältö on tuotettu riittävän rakenteisessa muodossa.

Verkkopalvelun tietosisällön tuottaminen muodostuu usein niin suureksi tehtäväksi, että yksi henkilö ei siitä selviydy. Harvoin pelkkä sisällön kertaluontoinen tuottaminen edes riittää, vaan lähes aina elävä verkkopalvelu vaatii tietosisällön jatkuvaa ylläpitoa. Nykyaikaisen verkkopalvelun menestyksekkääseen toteuttamiseen tarvitaan useiden konkreettisten tekniikoiden lisäksi erilaisten odotusten ja vaatimusten tuntemusta. Jos esimerkiksi yrityksen verkkopalvelun tietosisällön ylläpito tulisi suorittaa muiden työtehtävien ohessa, ei työntekijöiltä voida edellyttää tämän monitahoisen prosessin hienouksien opiskelua. Tästä johtuen tietosisällön ylläpitoon tulee kehittää helpokäyttöisiä työkaluja.

Tässä työssä selvitetään mukautuvien hypermediajärjestelmien taustalla vallitsevaa yleistä teoriaa, ja pohditaan miten World Wide Web (WWW) toimintaympäristönä vaikuttaa käytettäviin mukautustekniikoihin ja mahdollisiin sovelluskohteisiin. Työssä myös kartoitetaan millaisia ominaisuuksia erilaisista sisällönhallintaan kehitetyistä järjestelmistä löytyy. Lisäksi tutkitaan millaisia rajoitteita mukauttavaksi soveltuva tietosisältö aiheuttaa verkkopalvelun tietomallille sekä miten täl-

laisen rakenteisen tiedon keräämistä voidaan tukea web-pohjaisen käyttöliittymän kautta.

Tämä työ on toteutettu osana Raha-automaattiyhdistyksen (RAY) rahoittamaa, Tampereen teknillisen yliopiston (TTY) Hypermedialaboration, Pirkanmaan Allergia- ja Astmayhdistys ry:n ja Suomen Keliakialiitto ry:n yhdessä toteuttamaa Aatu-hanketta. Hankkeessa tuotettiin verkkopalvelu, joka tarjoaa ruokaohjeita ja elintarviketuotetietoja erityisesti erityisruokavalioita noudattaville ihmisille. Palvelun keskeisiä ominaisuuksia ovat ylläpitäjille tarjottu mahdollisuus hallinnoida palvelun tietosisältöä web-pohjaisen käyttöliittymän kautta sekä loppukäyttäjille näytettävän sisällön mukautuminen heidän ruokavalioihinsa sopivaksi. Tämän työn päämääränä voidaankin pitää tarvittavan esikartoituksen suorittamista, jotta edellä mainittu verkkopalvelu voitiin toteuttaa siten, että se tukee käyttäjiään heidän tavoitteidensa saavuttamisessa.

1.1 Määritelmät

Arkikielessä termillä ”tieto” voidaan tarkoittaa tilanteesta riippuen joko dataa, informaatiota tai tietämystä. *Data* on joukko käsittelemättömiä symboleja, jotka eivät itsessään liity mihinkään laajempaan asiayhteyteen. Tästä johtuen datalla ei ole omaa merkitystä tai suhteita mihinkään merkityksen omaavaan käsitteeseen. Tietokoneet tallentavat, siirtävät ja käsittelevät dataa binaarimuodossa. *Informaatio* puolestaan on dataa, joka on koostettu merkitykselliseksi rakenteeksi, joka välittää vastaanottajalleen vastauksia kysymyksiin kuten kuka, mitä, missä ja milloin. Ymmärtääkseen informaation tulee sen vastaanottajan ymmärtää yksittäisten dataalkioiden väliset suhteet. Vastaavasti *tietämys* on vastaanottajansa malleiksi ja toistettaviksi prosesseiksi organisoimaa informaatiota, joka vastaa kysymykseen miten. Tietämys on siis informaatiota korkeampitasoinen ja kattavampi, johdonmukainen ja pysyvä arkkityyppi, joka on aina riippuvainen vastaanottajansa subjektiivisesta tulkinnasta. [Nunamaker 2001]

Tässä työssä keskitytään käsittelemään datan, informaation ja tietämyksen sijasta pääasiassa verkkopalvelun kautta julkaistavaa *tietosisältöä*. Käytännössä tietosisältö on varastoitu datana palvelimella, se esitetään verkkoselaimen kautta informaationa käyttäjälle, joka voi oman tulkintatapansa perusteella muodostaa siitä tietämystä. Tietosisällöllä tarkoitetaan tässä yhteydessä sitä verkkopalvelun tarjoamaa keskeistä informaatiota, joka talletetaan erilliseen tietovarastoon (esimerkiksi tietokantaan tai XML-dokumentteihin) ja sivunlatauspyynnön yhteydessä upotetaan erilliseen sivupohjaan. Sivun vastaanottaja on oletettavasti kiinnostunut juuri

tästä tietosisällöstä, eikä esimerkiksi sivupohjaan kuuluvasta, dokumentin rakennetta kuvaavasta HTML-merkkauksesta (Hypertext Markup Language). Tässä työssä tietosisällön yhteydessä käytetään termiä ”tieto” tarkoitettaessa tietosisällön yksittäistä, rakenteisessa muodossa kerättyä informaatioalkiota.

Koska tietokoneet eivät kykene hahmottamaan käsittelemäänsä tietosisältöä samalla tavoin kuin ihmiset, ovat koneen mahdollisuudet käsitellä tätä tietosisältöä melko rajalliset. *Rakenteisella tiedolla* tarkoitetaan tietoa, joka on jäsennelty sellaisiin kokonaisuuksiin, että tietoalkioiden mekaaninen käsittely, kuten esimerkiksi vertailu, on mahdollista.

Pelkkä tietosisällön jäsentely ei kuitenkaan riitä tehokkaaseen tietojenkäsittelyyn, vaan koneelle pitää myös kertoa sen käsittelemän tiedon merkitys, eli muuntaa data informaatioksi. Tämä toteutetaan *metatiedolla*, joka on ”tietoa tiedosta”, eli tietosisältöä kuvailevaa informaatiota (vrt. esim. [Salminen 2005]). Tietosisällön mekaanisen käsittelyn lisäksi metatietoa voidaan hyödyntää tietojärjestelmän sisällön hallinnassa.

Koska työn loppuosa keskittyy ruokaohjeita sisältävän verkkopalvelun toteutustyön kuvailuun, ei ruokaohjeisiin liittyvältä termistöltä voida välttyä. Tiedon mallintamisen näkökulmasta termejä voidaan kuitenkin käyttää eri merkityksessä kuin arkikäytössä. Tämän vuoksi seuraavassa määritellään tässä työssä käytettävät ruokaohjeisiin liittyvät käsitteet.

Ruokaohje kertoo lukijalleen miten yksittäinen *ruokalaji* voidaan valmistaa. Esimerkki ruokaohjeesta on mustikkapiirakan valmistuohje. Ruokaohjeesta voidaan johdattaa erilaisia *ruokaohjemuunnelmia*, jotka ovat hyvin samankaltaisia alkuperäisen ruokaohjeen kanssa, mutta sisältävät jonkin muutoksen, jonka seurauksena valmistuva ruokalaji eroaa alkuperäisen ohjeen ruokalajista. Esimerkiksi korvaamalla mustikkapiirakan mustikat mansikoilla saadaan mansikkapiirakan ohje. On vaikeaa määrittellä koska ruokaohje muuttuu muunnelmaksi ja koska uudeksi ruokaohjeeksi. Ensinnäkin ruokaohjetta voi muuttaa niin vähän, että kyseessä on periaatteessa vielä sama ruokalaji: esimerkiksi sokerin korvaaminen muulla makeutusaineella ei muuta ohjeen perusteella valmistuvaa piirakkaa juuri lainkaan. Koska esimerkin kaltainen pienikin muutos valmistusaineissa saattaa ratkaista, soveltuuko ruokalaji erityisruokavaliota noudattavalle ihmiselle, toteuttamassamme verkkopalvelussa kyseessä olisi erillinen ruokaohjemuunnelma. Toisaalta pienikin muutos ruokaohjeessa voi vaikeuttaa valmistuvaan ruokalajiin niin paljon, että kyseessä ei enää ole muunnelma, vaan kokonaan uusi ruokaohje. Tämä työ ei pyri määrittelemään missä kulkee raja ruokaohjeiden ja ruokaohjemuunnelmien välillä, vaan tapauskohtainen päätösvalta

jätetään sisällöntuottajalle. Ruokaohjemuunnelmista käytetään tästä edespäin lyhyempää muotoa ”muunnelma”.

Valmistusaine on ruokaohjeessa mainittu, ruokalajin toteuttamiseen tarvittava ainesosa. Valmistusaine sisältää pelkän ainesosan lisäksi tiedon ainakin siitä, paljonko kyseistä ainesosaa tarvitaan. Lisäksi valmistusaineeseen voi liittyä muuta ruokalajin valmistamisen kannalta olennaista informaatiota. Valmistusaineita ovat esimerkiksi yksiselitteinen ”0,5 litraa maitoa” ja tulkinnanvaraisempi ”muutama iso banaani”.

Ruokaohjeeseen liittyy *valmistusvaiheita*, joissa kerrotaan miten valmistusaineita tulee käsitellä, jotta tavoitteena oleva ruokalaji saadaan valmistettua. Ruokaohjeen valmistusaineet voidaan jaotella *työvaiheisiin*, jotka ovat loogisia osakokonaisuuksia. Esimerkiksi mustikkapiirakan ruokaohjeessa työvaiheita voisivat olla taikinan ja täytteen valmistus.

Elintarviketuote on yhdestä tai useammasta valmistusaineesta tietyllä tavalla myyntiä varten valmistettu tuote, kuten esimerkiksi mustikkakeitto, jossa on mustikoiden lisäksi vettä, sokeria ja perunatärkkelystä. Elintarviketuotteesta käytetään tässä työssä lyhyempää termiä ”tuote”. Tuotteet voidaan luokitella *tuoteryhmiin* tuotteiden ominaisuuksien perusteella.

Tässä työssä käytetään käsitettä *tavallinen käyttäjä* kuvaamaan henkilöitä, jotka hallitsevat verkkopalvelun peruskäytön, mutta eivät tunne verkkopalveluihin liittyviä teknisiä yksityiskohtia, kuten merkkkaus- ja ohjelmointikieliä, tiedonsiirtoprotokollia tai asiakas-palvelin -arkkitehtuuria. Työn kannalta on kiinnostavaa miten tällaiset käyttäjät selviytyvät verkkopalvelun sisällönhallinnasta.

1.2 Työn rakenne

Toisessa luvussa määritellään työn kannalta olennaiset peruskäsitteet hypermedia ja hypermediajärjestelmä. Lisäksi luvussa esitellään prosessi, jolla hypermediajärjestelmän sisältöä voidaan mukauttaa ja havainnollistetaan miten mukautumisella voidaan tukea palvelun käyttäjiä. Lopuksi tutustutaan hypermediajärjestelmistä tunnetuimman WWW:n asettamiin reunaehtoihin mukautuvaa verkkopalvelua toteutettaessa.

Kolmannessa luvussa luodaan tiivis katsaus WWW:n lyhyen historian aikana verkkopalveluiden sisällönhallinnassa tapahtuneeseen kehitykseen. Tämän jälkeen tarkastellaan miten suuria käyttäjämääriä saavuttaneet wiki- ja blogi-järjestelmät

lähestyvät sisällönhallintaa sekä pohditaan millaisen alustan WWW tarjoaa sisällönhallintaan tarkoitettujen käyttöliittymien toteuttamiseen.

Neljäs luku esittelee tiedon mallintamisen käsitteen, jonka jälkeen tutkitaan miten työhön liittyvän hankkeen kannalta tärkeitä ruokaohjeita on mallinnettu eri tahoilla. Lisäksi tutkitaan millaisia reunaehtoja sisällön mukautuminen aiheuttaa tietomallille ja miten tietomalli puolestaan vaikuttaa sisällöntuotantoon.

Työn yhteydessä toteutettu verkkopalvelu ja sen kohdeyleisö esitellään *viidennessä luvussa*. Myös käytetyt toteutustekniikat ja niiden vaikutukset toteutustyöhön käydään läpi. Lopuksi pohditaan miten tässä työssä kartoitetut kolme teoriakokonaisuutta, eli mukautuminen, tiedon mallintaminen ja sisällönhallinta heijastuivat verkkopalvelun toteutukseen.

Kuudennessa luvussa arvioidaan verkkopalvelun toteutuksen onnistumista sekä toteutustyön aikana ilmenneitä jatkokehitysideoita. Lopulta työn tulokset tiivistetään yhteenvedoksi *seitsemännessä luvussa*.

Luku 2

Mukautuvat verkkopalvelut

Tämä luku esittelee tarkemmin käsitteet hypermedia, hypermediajärjestelmä ja mukautuva hypermediajärjestelmä. Lisäksi tutkitaan mukautuvuuden alla piilevää prosessia. Lopuksi pohditaan World Wide Webin ominaisuuksia, jotka vaikuttavat mukautuvien verkkopalveluiden tuottamiseen.

2.1 Hypermedia ja hypermediajärjestelmät

Vuonna 1945 Vannevar Bush kuvasi artikkelissaan ”As We May Think” [Bush 1945] kuinka ihmiskunnan tuottama valtava informaatiomäärä jää hyödyntämättä kattavan ja nopean hakumahdollisuuden puuttuessa. Ratkaisuksi ongelmaan ehdotettiin teoreettista Memex-järjestelmää (memory extender), joka olisi mahdollista tuottaa taloudellisesti ja luotettavasti tekniikoilla, joiden Bush arveli tulevan saataville jo silloisessa lähitulevaisuudessa. Järjestelmä pystyisi mikrofilmien avulla tallentamaan informaation pienen tilaan. Ennen kaikkea se tarjoisi mahdollisuuden liittää asioita toisiinsa assosiaation perusteella. Toisiinsa liittyvät asiat muodostaisivat polkuja informaatioavaruuteen, ja polut puolestaan muodostaisivat käsittekartan tyyppisiä verkkoja. Tämä ihmismielelle tyypillinen tapa luokitella asioita mahdollistaisi paljon perinteistä aakkosellista hakemistoa tehokkaamman tavan hakea ja selata informaatiota. Käytännössä Memex toimisi siis ihmismielen jatkeena.

Memex-konetta ei sellaisenaan kuitenkaan koskaan rakennettu. Vuonna 1965 Theodor Nelson [Nelson 1965] jalosti Bushin ideaa. Hän totesi, että vaikka silloiset tietokoneet olivat laitteistonsa puolesta riittävän kehittyneitä, olivat liiketalouden tai tieteen käyttöön suunnitellut ohjelmistot sopimattomia yksittäisen henkilön kirjoitusten ja muistiinpanojen käsittelyyn. Niinpä hän esitteli konseptin ohjelmistos-

ta, joka toimisi ”moni-ilmeisenä, monimuotoisena, moniulotteisena ja äärettömänä liitutauluna”, ja mahdollistaisi henkilökohtaisen informaation varastoinnin ja haun käyttäjäystävällisellä tavalla.

Nelsonin ideana oli, että järjestelmässä ei olisi hierarkkista tiedostojärjestelmää, vaan käyttäjä itse muodostaisi rakenteen, jonka mukaan informaatio järjestelmään taltioitaisiin. Pienin rakennuspala olisi merkintä (engl. entry), joka voisi olla esimerkiksi pätkä tekstiä tai kuva. Listat (engl. lists) koostuisivat käyttäjän tiettyyn järjestykseen asettamista merkinnöistä, ja sama merkintä voisi olla osana useammassa listassa. Eri listoissa olevia merkintöjä voisi yhdistää käyttäjän määrittämällä kaksisuuntaisilla linkeillä (engl. links).

Artikkelissaan Nelson lanseerasi käsitteen hyperteksti tarkoittamaan ”teksti- ja kuvamateriaalia, joka sisältäisi monimutkaisia sisäisiä ristiviittauksia tavalla, jota ei voisi käyttökelpoisesti esittää paperilla”. Hyper-etuliitteellä Nelson halusi siis kuvata hypertekstiä tavallisen lineaarisen tekstin jatkeena. Tämä hypertekstin sisältö voisi kasvaa rajattomasti ja koostua aina suuremmasta osuudesta kaikesta maailman kirjoitetusta informaatiosta.

Nykyään hyperteksti tyypillisesti mielletään joukkona solmuja (engl. nodes, vrt. Nelsonin listat) ja joukkona näitä yhdistäviä linkkejä (vrt. esim. [Cicconi 1999]). Solmut ovat yksinkertaisimmillaan atomaarisia informaatioalkiota, joita ei voida pilkkoa pienempiin osiin menettämättä informaation hyödyllisyyttä. Tällöin yksittäinen solmu voi olla esimerkiksi yksittäinen tekstikappale. [Lowe & Hall 1999] Linkit kuvaavat solmujen välistä yhteyttä. Käyttäjä voi linkin kautta siirtyä solmusta toiseen. Nelsonin linkeistä poiketen linkit voivat yleensä olla myös yksisuuntaisia. Linkkien muodostaessa verkon useiden solmujen välillä, pääsee tietystä solmusta toiseen solmuun mahdollisesti useita eri reittejä pitkin, jolloin käyttäjä voi edetä hypertekstissä parhaaksi katsomassaan järjestyksessä. Kaikkia hypertekstin sisältämiä solmuja ja niitä yhdistäviä linkkejä kutsutaan myös hyperavaruudeksi (engl. hyperspace).

Hyperteksti on siis potentiaalisesti kooltaan rajoittamaton verkko, joka koostuu solmuista ja linkeistä, jotka mahdollistavat epälineaarisen etenemisen. Hypertekstin rakenne on selkeästi määritelty, mutta muokattavissa ja laajennettavissa. Verkon rajoittamattomuudella tarkoitetaan sitä, että jokainen solmu voidaan liittää linkillä toiseen saman hypertekstin tai muun yhteensopivan hypertekstin solmuun. Epälineaarinen eteneminen antaa käyttäjälle mahdollisuuden vaivattomasti seurata mielenkiintoiseksi kokemiaan linkkejä, ja näin soveltaa omaa ajattelumalliaan tekstiä lukiessaan, sen sijaan että seuraisi kirjoittajan päättämää suoraviivaista reittiä. [Cicconi 1999]

Hypermedia voidaan määritellä hypertekstin laajenuksena, joka hyödyntää tekstin lisäksi muita mediaelementtejä, kuten kuvia, ääntä ja videota [Lowe & Hall 1999]. (Tämä määritelmä eroaa merkittävästi Nelsonin määrittelemästä hypermediasta, jolla hän tarkoitti epälineaarisesti käsiteltävää (engl. browsable or vari-sequenced) video- tai äänitallennetta, johon ei olisi yhdistetty muita mediamuotoja.) Ääntä voidaan käyttää esimerkiksi puheena (esimerkiksi sanan lausumisohje), musiikkina (tunnelman luonti) tai varoitus-, huomio- ja tehosteääninä. Tavallisten kuvien lisäksi voidaan käyttää animaatioita ja videokuvaa. (Vrt. [Tervakari & Silius 2007].)

Hypermediajärjestelmä on mikä tahansa järjestelmä, joka toteuttaa edellä mainitut hypermedian perustoiminnot. Kobsa, Koenemann & Pohl (2001) määrittelevät hypermediajärjestelmän interaktiivisena järjestelmänä, jonka avulla käyttäjät voivat navigoida hypermediaobjektien muodostamaa verkkoa. Lowe ja Hall (1999) tulkitsevat hypermediajärjestelmän informaation hallintajärjestelmänä, joka mahdollistaa informaatioavaruudessa liikkumisen miellelyhtymiin perustuvien linkkien (engl. associative links) avulla.

2.2 Mukautuvat hypermediajärjestelmät

Käyttäjäkunnan ja tietosisällön kasvaessa perinteisen staattisen hypermediajärjestelmän käyttö saattaa vaikeutua, sillä ne tarjoavat kaikille käyttäjille saman sisällön ja hypermedialinkkivalikoiman. Käyttäjäkunnan laajentuessa käyttäjien tavoitteet, tietotaidot ja mieltymykset monipuolistuvat, jolloin sama sisältö ei enää palvele kaikkia käyttäjiä yhtä hyvin. Esimerkiksi staattinen hypermediajärjestelmä saattaa hukuttaa aloittelevan käyttäjän sisällön ja toimintojen paljouteen tai vastaavasti tarjota kokeneelle käyttäjälle jatkuvasti tälle itsestään selvää informaatiota ja tarpeettomia avusteita. Hyperavaruuden kasvaessa mahdolliset navigointireitit saattavat kasvaa eksponentaalisesti, jolloin käyttäjä saattaa eksyä tai kokea kognitiivisen ylikuormituksen pyrkiessään selvittämään mikä tarjotuista navigointireiteistä on tälle paras mahdollinen. (Vrt. [Brusilovsky 1996].)

Näitä ongelmia on eri hypermediajärjestelmissä pyritty ratkaisemaan luomalla hypermediajärjestelmistä mukautuvia. Mukautuva hypermediajärjestelmä siis huomioi erilaiset käyttäjät ja käyttökontekstit muokkaamalla käyttäjälle näyttämäänsä sisältöä tilanteeseen sopivaksi. Näin järjestelmän käyttö (ideaalisesti) helpottuu ja tehostuu. Kuten mille tahansa palvelulle, myös verkkopalvelulle on tärkeää tuottaa käyttäjälleen lisäarvoa. Lisäarvon tuottamisessa puolestaan on tärkeää tuntea käyttäjät ja palvella näitä yksilöllisesti [Kobsa, Koenemann & Pohl 2001].

Brusilovsky (1996) määrittelee mukautuvan hypermediajärjestelmän seuraavien kolmen kriteerin perusteella:

- Järjestelmän tulee olla hyperteksti- tai hypermediajärjestelmä.
- Järjestelmällä tulee olla malli käyttäjästä.
- Järjestelmän tulee muokata hypermediasisältöään käyttäjämallin perusteella, eli tarjota eri käyttäjämallin omaaville käyttäjille erilaiset näkymät sisältöön.

Mukautuvat hypermediajärjestelmät siis luovat jokaiselle käyttäjälle oman käyttäjämallin, joka koostuu käyttäjän tavoitteista, mieltymyksistä ja tietämyksestä. Huomioimalla tämän mallin kaikessa käyttäjän kanssa tapahtuvassa vuorovaikutuksessa järjestelmä mukautuu käyttäjänsä tarpeisiin. [Brusilovsky 2001] Ideaalisesti tämä tarkoittaisi, että jokaiselle käyttäjälle tarjottaisiin juuri hänelle oikeaa sisältöä ja toimintoja oikeaan aikaan (vrt. esim. [Chiu 2001]). Vaikka tähän ideaalitulanteeseen pääseminen olisikin vain teoreettinen tavoite, voidaan mukautumisella usein parantaa hypermediajärjestelmän käyttökokemusta myös käytännön sovelluksissa.

Esimerkiksi ruokaohjeita sisältävä mukautuva hypermediajärjestelmä voi havaita käyttäjän katselevan pelkästään erilaisten kasvisruokien ohjeita. Jos käyttäjä nyt pyytää nähdä lasagnen ohjeen, voidaan sisältöä mukauttaa ja tarjota ohjeesta ensin muunnelmaa, jossa liha on korvattu kasviksilla. Vastaavasti jos käyttäjä haluaa nähdä kaikki suolaisten leivonnaisten ohjeet, voidaan sisällön esitystapaa muuttaa sijoittamalla tarjottavassa listassa sopivat kasvisruoat ennen liharuokia.

On olennaista huomioida, että hypermediajärjestelmä, joka muuttaa näyttämäänsä sisältöä esimerkiksi satunnaisesti tai ajan funktiona, ei ole määritelmän mukaan mukautuva hypermediajärjestelmä. On syytä myös erottaa toisistaan mukautuvat ja mukautettavat hypermediajärjestelmät: ensin mainitussa mukautus tapahtuu (käyttäjämallin luonnin jälkeen) automaattisesti, kun taas jälkimmäisessä käyttäjä aktiivisesti muokkaa järjestelmän tarjoamia näkymiä mieleisikseen.

Myös mukautuvan ja mukautettavan hypermediajärjestelmän välimuoto on mahdollinen. Tällaisia ovat esimerkiksi käyttäjän hallitsemat ja käyttäjälähtöiset mukautukset. Käyttäjän hallitsemassa mukautumisessa (engl. user-controlled adaptivity) järjestelmä ehdottaa käyttäjälle mukautuksia, jotka käyttäjä voi halutesaan ottaa käyttöön. Käyttäjälähtöisessä mukautumisessa järjestelmä suorittaa parhaaksi katsomansa mukautukset, mutta vain jos käyttäjä tätä erikseen pyytää. [Kobsa, Koenemann & Pohl 2001]

Sopivan mukautumisen ja mukautettavuuden tasot on syytä arvioida tapauskohtaisesti. Käyttäjän kannalta on olennaista, että prosessi ei ole liian vaativa tai

ärsyttävä. On tärkeää myös vertailla mukautuvuudesta saatavia hyötyjä toteutus-työn vaativuuteen ja mahdollisten virheellisten mukautusten aiheuttamiin ongelmiin. (Vrt. [Kobsa, Koenemann & Pohl 2001].) Chiu (2001) huomauttaa myös, että mukauttamisen suorittaminen kuluttaa palvelinjärjestelmän resursseja, joten tämä täytyy huomioida varsinkin silloin, kun järjestelmällä tulee olemaan suuri käyttäjäkunta.

Mukautuminen ei automaattisesti ratkaise kaikkia hypermediajärjestelmiin liittyviä ongelmia. Huonosti toteutettu mukautus saattaa päinvastoin hämmentää käyttäjää ja tehdä järjestelmästä vaikeamman käyttää. Mukautuvuutta suunniteltaessa tulisikin pitää mielessä, että tarkoituksena on tukea käyttäjää saavuttamaan tavoitteensa, eikä toteuttaa tarpeettomia mukautuksia vain koska se on mahdollista.

2.3 Mukautusprosessi

Hypermediajärjestelmän toteuttama mukautusprosessi sisältää kolme päävaihetta. Ensimmäinen vaihe on tiedonkeruu (engl. acquisition) jonka aikana tunnistetaan saatavilla oleva tieto käyttäjästä (engl. user data), tämän toiminnasta (engl. usage data) ja käyttökontekstista (engl. environment data). Toisen vaiheen muodostavat ensimmäisessä vaiheessa kerätyn tiedon muokkaaminen sopivaan muotoon (engl. representation) ja sopivien oletusten tekemistä käyttäjästä tämän tiedon perusteella (engl. secondary inference). Ensimmäinen ja toinen vaihe muodostavat päättymättömän prosessin, jossa käyttäjämallia päivitetään jatkuvasti käyttäjästä saatavan uuden tiedon perusteella. Viimeisessä vaiheessa suoritetaan itse mukautus kerätyn tiedon ja siitä tehtyjen päätelmien perusteella. (Vrt. esim. [Brusilovsky 1996], [Kobsa, Koenemann & Pohl 2001].)

2.3.1 Tiedonkeruu

Tiedonkeruun tarkoituksena on luoda käyttäjästä malli, joka kuvaa käyttäjän kiinnostuksen kohteita, aiempaa käyttäytymistä sekä muita ominaisuuksia, joita hypermediajärjestelmää ylläpitävä taho pitää oleellisina. Nämä tiedot voidaan kerätä käyttäjältä joko eksplisiittisesti tai implisiittisesti. Myös aiemmin kerätyn tiedon hyödyntäminen on mahdollista. [Chiu 2001] Koska käyttäjästä tarvitsee kerätä tietty määrä tietoa ennen kuin mukauttaminen voidaan suorittaa, saadaan mukauttamisesta todellista hyötyä jos järjestelmän olla sellainen, että sitä käytetään joko pitkiä aikoja kerralla tai toistuvina käyttökertoina [Kobsa, Koenemann & Pohl 2001].

Kun tieto kerätään suoraan käyttäjältä kysymällä, esimerkiksi rekisteröinnin yhteydessä, puhutaan eksplisiittisestä tiedonkeruusta. Eksplisiittisen tiedon etu on siinä, että käyttäjä voi suoraan kertoa millaista sisältöä tämä haluaa nähdä [Chiu 2001]. Koska monimutkaista päättelyprosessia ei tarvita, virhepäätelmiä ei tapahdu. Tyypillisesti esimerkiksi demografisen tiedon tapauksessa tämä on myös ainoa käytettävissä oleva keino. Käyttäjältä kysymiseen liittyy kuitenkin joitakin ongelmia. Käyttäjät voivat kokea kyselyprosessin epämiellyttäväksi, varsinkin jos he eivät usko kerätyn tiedon auttavan heitä välittömien tavoitteidensa saavuttamisessa. Prosessi on myös virhealtis, sillä käyttäjät voivat helposti arvioida oman tietotaitotasonsa väärin. [Kobsa, Koenemann & Pohl 2001]

Implisiittisen tiedonkeruun tapauksessa tietoa kerätään seuraamalla käyttäjän toimia. Tämä tapahtuu käyttösuoituksen taustalla, eikä käyttäjän aktiivista osallistumista tarvita. [Chiu 2001] Edut ja haitat ovat eksplisiittisen tiedonkeruun vastakohtia: tiedonkeruuprosessi ei häiritse käyttäjää, mutta saatu tieto saattaa sisältää virhepäätelmiä.

Aiemmin kerätyn tiedon hyödyntäminen tarjoaa tyypillisesti rikkaimman tietolähteen käyttäjämallin rakentamiseen [Chiu 2001]. Jotta aiemmin kerättyä tietoa voidaan hyödyntää, täytyy käyttäjä joka käyttökerralla tunnistaa. On syytä myös huomioida, että aiemmin kerätyt tiedot ovat saattaneet vanhentua, eivätkä enää palvele tarkoitustaan mukauttamisprosessissa.

Käyttäjätieto

Käyttäjätieto (user data) koostuu käyttäjän ominaispiirteistä, kuten demografisista tekijöistä, käyttäjän tietotaidoista, mieltymyksistä ja tavoitteista. Demografisiin tekijöihin kuuluvat esimerkiksi käyttäjän henkilötiedot (nimi ja osoite), geografinen tieto (sijainti vaikkapa maan tai postinumeron tasolla), henkilöluokitukset (ikä, sukupuoli, koulutus ja ammatti) sekä psykografiset tiedot (luonne, arvot, asenteet ja elämäntyyli). [Kobsa, Koenemann & Pohl 2001]

Käyttäjätiedon implisiittinen kerääminen perustuu usein monimutkaisiin päätelyketjuihin, jolloin saatuihin tuloksiin ei voida luottaa sokeasti. Tästä johtuen käyttäjätiedon yhteydessä puhutaan usein oletuksista. [Kobsa, Koenemann & Pohl 2001] Jotta mukautus voidaan suorittaa oletuksen perusteella, täytyy sen olla riittävän hyvin perusteltu. Tästä johtuen käyttäjätieto kerätäänkin usein eksplisiittisesti.

Käyttötieto

Käyttötiedon (usage data) muodostavat ne käyttäjän toimet, joiden ei voida katsoa kuuluvan käyttäjätietoon, mutta joita voidaan kuitenkin käyttää hyödyksi mukauttamisessa [Brusilovsky 2001]. Tunnistamalla miten käyttäjä toimi tietyssä tilanteessa voidaan myöhemmin vastaavissa tilanteissa tukea käyttäjää tarjoamalla hänelle sopivia toimintoja tai suorittamalla ne automaattisesti hänen puolestaan [Kobsa, Koenemann & Pohl 2001].

Kobsa, Koenemann & Pohl (2001) luokittelevat käyttötiedon kahden ominaisuuden perusteella. Kerätyn tiedon voidaan siis kuvitella sijoittuvan kaksiakseliselle tasolle. Ensimmäisen akselin muodostaa indikaattorin vahvuus, joka kuvaa indikaattorin luotettavuutta. Heikot indikaattorit sisältävät merkittäviä epävarmuustekijöitä, eikä niiden perusteella voida tehdä tärkeitä päätelmiä, kun taas vahvat indikaattorit ovat selkeitä todisteita joiden perusteella voidaan päätellä jotain käyttäjästä. Toisen akseli kuvaa sitä, osoittaako toiminta käyttäjän positiivista vai negatiivista reaktiota, ja onko tämä reaktio vahva, kuten todella epämiellyttävä kokemus, vaiko heikko, kuten lievästi miellyttävä tapahtuma.

Käyttötieto kerätään tarkkailemalla käyttäjän toimia, kuten millaista sisältöä tämä valitsee nähtäväkseen. Tietoa voidaan kerätä käyttäjän tekemien valintojen perusteella, esimerkiksi käyttäjän valitessa linkin pitkästä listasta. Käyttäjän suorittamat valinnat ovat heikkoja positiivisia indikaattoreita, koska käyttäjä oletettavasti valitsee itseään kiinnostavimman vaihtoehdon (selvästi positiivinen indikaattori), mutta ei voida olla varmoja siitä, että käyttäjä on selkeästi hahmottanut kaikki vaihtoehdot (epävarmuutta, heikko indikaattori). Valitsematta jääneistä linkeistä ei voida päätellä juuri mitään. [Kobsa, Koenemann & Pohl 2001]

Käyttäjän tietyssä solmussa viettämä aika on tyypillinen kiinnostusindikaattori. Jos käyttäjä poistuu solmusta nopeasti, voidaan tätä pitää vahvasti negatiivisena indikaattorina. Toisaalta pitkä vierailuaika ei välttämättä tarkoita, että käyttäjä olisi käyttänyt kaiken tämän ajan solmun sisällön tutkimiseen, joten kyseessä on vain heikko positiivinen indikaattori. [Kobsa, Koenemann & Pohl 2001]

Verkkokaupassa tehty ostos on luonnollisesti vahva positiivinen indikaattori ostettua tuotetta kohtaan. Pelkästään tuotteen siirtämistä ostoskoriin voidaan pitää positiivisena indikaattorina, vaikkei tuotetta lopulta ostettaisikaan. Vastavasti tietyn solmun tallentaminen, tulostaminen, lisääminen kirjanmerkkeihin tai välittäminen eteenpäin muille käyttäjille ovat kaikki positiivisia indikaattoreita. [Kobsa, Koenemann & Pohl 2001]

Oman käyttötiedon ryhmänsä muodostavat käytön säännönmukaisuudet. Käyttötaajuus, esimerkiksi kuinka usein käyttäjä vierailee tiettyssä solmussa, kertoo siitä kuinka tärkeä toiminto on käyttäjälle. Säännöllisesti käytetyt komennot voidaan tuoda etualalle, esimerkiksi listan kärkeen, kun taas harvoin käytetyt toiminnot voidaan tiputtaa listan loppupäähän. Vastaavasti tiettyjen toimintojen käyttöä voidaan analysoida käyttötilanteen mukaan (engl. situation-action correlation), jolloin tietyn toiminnon suorittaminen voidaan tehdä helpoksi niissä tilanteissa, joissa käyttäjä sitä usein käyttää, mutta ei vaikuteta sen suorittamiseen muissa tilanteissa. Kolmannen mukauttamismahdollisuuksia tarjoavan käytön säännönmukaisuusryhmän muodostavat toimintoketjut (engl. action sequences), joissa tietyt toiminnot suoritetaan usein peräkkäin samassa järjestyksessä. Näiden toimintoketjujen suorittamista voidaan tällöin tukea tai automatisoida. [Kobsa, Koenemann & Pohl 2001]

Käyttökonteksti

Mikäli hypermediajärjestelmää voidaan käyttää eri paikoista erilaisilla laiteilla, voidaan mukautuksessa huomioida myös käyttökonteksti. Käyttökonteksti sisältää siis fyysisen, teknisen ja sosiaalisen ympäristön, jossa järjestelmää käytetään. Fyysisellä ympäristöllä tarkoitetaan käyttäjän sijaintia ympäristöönsä nähden sekä käyttöolosuhteita, kun taas teknisellä ympäristöllä tarkoitetaan käytettäviä laitteita ja ohjelmistoja. Sosiaalinen ympäristö kuvaa niitä tiedostettuja ja alitajuisia vaikutuksia, jotka käyttäjä kokee järjestelmän käytön yhteydessä muiden ihmisten ja näiden välisten sosiaalisten suhteiden vuoksi. Kuten käyttäjätiedon, myös käyttökontekstintiedon implisiittinen kerääminen on haastavaa.

Mikäli käyttäjän sijainti tiedetään, voidaan näytettävää tietoa mukauttaa siten, että käyttäjää lähellä olevat kohteet saavat muita suuremman prioriteetin. Mitä tarkemmin käyttäjän sijainti ja tätä lähellä olevat kohteet tiedetään, sitä yksityiskohtaisempia mukautuksia hypermediasisällölle voidaan toteuttaa. Käyttäjälle voidaan suositella tätä lähellä olevia kohteita, kuten ravintoloita käyttäjän kävellessä kaupungilla tai taideteoksia tämän liikkuesssa taidenäyttelyssä. Ympäristön ominaisuuksista esimerkiksi valaistus voidaan huomioida säätämällä näytön kontrastia valoisuuden mukaan ja taustamelu korvaamalla äänimerkit graafisilla vastineilla. [Kobsa, Koenemann & Pohl 2001]

Tekninen ympäristö voi koostua laajasta joukosta erilaisten laitteiden ja ohjelmistojen yhdistelmiä. Yleisesti katsoen eri laite- ja ohjelmistoversiot asettavat rajoituksia informaation esitystapaan. Tyypillinen tekniseen ympäristöön perustuva mukautus onkin valita informaation esittämiseen parhaiten sopivan media, kuten kuva tai video ([Joerding 1999], [Brusilovsky 2001]:n mukaan).

2.3.2 Käyttäjämallin rakentaminen

Käyttäjämallin luominen on mukauttamisen kannalta keskeinen tehtävä. Vaikka on mahdotonta aukottomasti ennustaa mitä käyttäjä tulee tekemään, voidaan siitä esittää valistuneita arvauksia käyttäjän aiemman toiminnan perusteella [Hirsh, Basu & Davison 2000]. Jotta tietokone voi suorittaa tämän tehtävän, täytyy käyttäjästä ja tämän toimista luoda malli, jota tietokone osaa tulkita. Tyypillisesti malli muodostuu joukoista ominaisuus-arvo-pareja (engl. "concept - value" pair). Tällöin kullekin ominaisuudelle talletetaan arvo, joka saattaa olla binaarinen (käyttäjä joko tuntee käsiteltävän aiheen tai ei tunne aiheetta), kvalitatiivinen (käyttäjän tietämys aiheesta on huono, kohtalainen tai hyvä) tai kvantitatiivinen (todennäköisyys sille, että käyttäjä tuntee aiheen). [Brusilovsky 1996]

Yksinkertaisempi ja karkeampi malli saadaan stereotyyppioivan päättelyn (stereotype reasoning) avulla. Tässä menetelmässä käyttäjät luokitellaan ominaisuuksien perusteella eri kategorioihin. Ennustukset tehdään näihin kategorioihin liittyvien stereotyyppisten olettamusten perusteella ja tiettyyn kategoriaan lokeroituille käyttäjille näytetään samalla tavalla mukautettu näkymä sisältöön. Tämä menetelmä on kätevä ensimmäisen arvauksen tekemiseen, kun käyttäjästä ei ole vielä paljon informaatiota saatavilla. [Kobsa, Koenemann & Pohl 2001] Kun käyttäjästä on myöhemmin kerääntynyt riittävästi informaatiota, voidaan mukauttaminen suorittaa henkilökohtaisen käyttäjämallin pohjalta.

Brusilovsky (1996, 2001) mainitsee seitsemän eri ominaisuutta, joita käyttäjästä voidaan koittaa mallintaa mukauttamisen onnistumiseksi.

Käyttäjän tavoitteet (engl. goals) Käyttäjän tavoitteet kertovat miksi järjestelmää käytetään. Syy voi olla esimerkiksi tietyn tehtävän suorittaminen, informaation löytäminen, ongelmanratkaisu tai oppiminen. Käyttäjällä voi olla useita eri tärkeysasteen tavoitteita, ja tavoitteet voivat koostua pienemmistä osatavoitteista. [Brusilovsky 1996]

Käyttäjän tietämys (engl. knowledge) Järjestelmä pitää kirjaa käyttäjän tietämyksestä käsiteltävästä aihealueesta esimerkiksi sen perusteella, mitä sisältöä käyttäjälle on jo näytetty. [Brusilovsky 1996]

Käyttäjän tausta (engl. background) Mukautus voidaan perustaa myös käyttäjän taustaan, eli siihen käsiteltävää aihealuetta koskevaan tietämykseen, jonka käyttäjä on hankkinut järjestelmän ulkopuolelta, esimerkiksi työnsä kautta. [Brusilovsky 1996]

Käyttäjän hyperavaruuden tuntemus (engl. hyperspace experience)

Hyperavaruuden tuntemus kuvaa kuinka hyvin käyttäjä tuntee hyperavaruuden rakenteen ja siinä navigoinnin. [Brusilovsky 1996]

Käyttäjän mieltymykset (engl. preferences) Käyttäjillä saattaa olla erilaisia mieltymyksiä sisällön suhteen. Nämä mieltymykset voivat olla ehdottomia, eli voimassa aina, tai suhteellisia, jolloin mieltymys riippuu kontekstista. [Brusilovsky 1996]

Käyttäjän kiinnostuksen kohteet (engl. interests) Mukauttamisen kannalta olennaisia kiinnostuksen kohteita ovat sellaiset, jotka pysyvät samoina käyttökerrasta toiseen. Lyhyempiaikaiset kiinnostuksen kohteet liittyvät yleensä suoraan käyttäjän tavoitteisiin, eivätkä kiinnosta käyttäjää enää tavoitteen saavuttamisen jälkeen. [Brusilovsky 2001]

Käyttäjän ominaispiirteet (engl. traits) Ominaispiirteitä ovat esimerkiksi luonteenpiirteet, kognitiiviset tekijät ja oppimistyyli. Ominaispiirteet ovat joko muuttumattomia tai muuttuvat erittäin hitaasti. Tämän tyyppisen tiedon kerääminen on hankalaa, ja siihen tarvitaan yleensä tätä varten suunniteltuja psykologisia testejä. [Brusilovsky 2001]

Käyttäjämallin ei tarvitse sisältää tietoa kaikista edellä mainituista ominaisuuksista, jotta mukauttaminen olisi mahdollista. Käytännössä mallin tulisi koostua vain niistä ominaisuuksista, joita todella voidaan hyödyntää mukauttamisessa. Tarpeettoman monimutkainen malli lisää työmäärää järjestelmää suunniteltaessa ja toteutettaessa sekä kuluttaa laskentatehoa ja varaa muistikapasiteettia järjestelmää käytettäessä.

2.3.3 Mukautus käyttäjämallin perusteella

Kuten kohdassa 2.1 todettiin, hyperavaruus koostuu solmuista ja niiden välisistä linkeistä. Tästä johtuen Brusilovsky (1996) jakaa mukautustekniikat kahteen osaan. Ensimmäisen osan muodostaa sisällön mukauttaminen (engl. adaptive presentation), jolla tarkoitetaan sivun tietosisällön esitystavan mukauttamista. Toinen osa on navigoinnin mukauttaminen (engl. adaptive navigation support), jossa muokataan sivulla esiintyviä muihin dokumentteihin johtavia linkkejä. Empiiristen tutkimusten mukaan navigoinnin mukauttaminen voi nopeuttaa navigointia ja oppimista, kun taas sisällön esitystavan mukauttaminen voi auttaa sisällön ymmärtämisessä ([Boyle & Encarnacion 1994]; [Brusilovsky & Pesin 1998]; [Kaplan, Fenwick & Chen 1993], [Brusilovsky & Maybury 2002]:n mukaan).

Sisällön mukauttaminen

Hypermediasolmun sisältö koostuu tekstistä, kuvista, äänestä ja videosta. Näiden mediaelementtien mukauttaminen perustuu piilottamiseen, valitsemiseen ja järjestämiseen [Brusilovsky 1996]. Sisältöä piilotettaessa solmu tai sen osa joko näytetään tai ei näytetä tietylle käyttäjälle. Sisältöä valittaessa taas valitaan eri vaihtoehdoista parhaiten käyttäjämalliin sopiva versio. Sisällön järjestäminen ei vaikuta siihen, mitä sisältöä käyttäjä näkee, vaan ainoastaa siihen, missä järjestyksessä asiat tälle esitetään. Luonnollisesti näitä kolmea menetelmää voidaan käyttää myös yhdessä.

Tietyn informaation näyttäminen tai piilottaminen voidaan nähdä saman kolikon eri puolina. Brusilovsky (1996) esittää, että käyttäjältä piilotetaan sisältö, joka on tälle tarpeetonta tai saattaa muuten häiritä käyttäjää. Tällöin esimerkiksi kokeelle käyttäjälle ei tarvitse tarjota samoja perustoimintojen suorittamista koskevia avusteita kuin aloittelevalla käyttäjällä. Aloittelevaa käyttäjää on turha hämmentää liian monimutkaisilla toiminnoilla ja informaatiolla, joten ne voidaan aluksi piilottaa, ja tuoda taas esiin, kun käyttäjä on perehtynyt järjestelmään paremmin.

Kobsa, Koenemann & Pohl (2001) taas katsovat asiaa toisesta suunnasta, ja ehdottavat, että käyttäjällä näytetään ylimääräistä sisältöä, jos se katsotaan tarpeelliseksi. Aloittelevalla käyttäjällä voidaan tarjota ylimääräistä informaatiota tukemaan sivun muun sisällön ymmärtämistä. Sivun sisällön sisäistäneelle käyttäjälle voidaan näyttää yksityiskohtaisia lisäinformaatiota, joka syventää sivulla esitettyä asiaa.

Ylimääräisen informaation näyttämisen erikoistapauksena voidaan nähdä verkkosivut, joihin liitetään dynaamisesti sisältöä tai linkkejä sivun pääsisällön tuottamisen jälkeen. Tyypillisesti tätä hyödynnetään suosittelemalle käyttäjälle ylimääräistä sisältöä. Esimerkiksi käyttäjän ostettua tietyn tuotteen verkkokaupasta, voidaan tälle tarjota informaatiota muista samankaltaisista tuotteista. Sen sijaan, että tätä verkkosivua tuotettaessa siihen sisällytettäisiin kaikki verkkokaupan tuotteet, joista käyttäjälle näytettäisiin sopivat, haetaan niiden tiedot dynaamisesti esimerkiksi tietokannasta. Vastaava toiminnallisuus voidaan toteuttaa uutissivustolla tarjoamalla tiettyyn aihepiiriin liittyviä uutisia lukeneelle käyttäjälle muita samoja avainsanoja sisältäviä viimeaikaisia uutisia. Olennaista on, että näitä muita uutisia ei välttämättä ollut edes julkistettu alkuperäisen uutisen luontihetkellä, vaan uutiset linkitettiin toisiinsa vasta sivun luonnin yhteydessä.

Hirsh, Basu ja Davison (2000) esittävät kaksi tapaa käyttäjää kiinnostavan sisällön tunnistamiseen. Sisältöön perustuva tunnistaminen (engl. content-based prediction) perustuu siihen, että järjestelmä tunnistaa millaisia asioita kohtaan käyttäjä on aiemmin osoittanut kiinnostusta, ja etsii samankaltaista sisältöä. Yhteisöllisissä

menetelmissä (engl. collaborative methods) käyttäjää kiinnostanutta sisältöä verrataan muita käyttäjiä kiinnostaneisiin sisältöihin. Jos kaksi käyttäjää ovat kiinnostuneet samoista asioista, voidaan käyttäjälle suositella sellaista materiaalia, joka on kiinnostanut toista käyttäjää, mutta johon käyttäjä itse ei ole vielä tutustunut.

Kobsa, Koenemann & Pohl (2001) nostavat sisällön mukauttamisessa omaan luokkaansa sopivan esitysmuodon valinnan (engl. adaptation of presentation and modality). Tällöin esitettävä asiasisältö pysyy ennallaan, mutta sitä esittävä mediaelementti korvataan toisella. Esimerkiksi hitaan verkkoyhteyden kautta sisältöä selaavalle käyttäjälle videoleike voidaan korvata kuvilla. Vastaavasti pieneltä näyttölaitteelta sisältöä tutkivalle voidaan tarjota suuren kuvan sijasta sama informaatio tekstimuodossa.

Sisällön valitseminen voidaan toteuttaa eri tavoilla. Koko solmusta (engl. page variant) tai sen osasta (engl. fragment variant) voidaan tuottaa vaihtoehtoisia muunnelmia, joista valitaan esitettäväksi käyttäjämalliin parhaiten sopiva. Kokonaisten vaihtoehtoisten solmujen tuottaminen ja ylläpitäminen on työlästä, mutta sopivan vaihtoehdon valitseminen on kevyt prosessi. Vastaavasti sivupohjaan upotettavien vaihtoehtoisten osien tuottaminen on helpompaa, mutta sivua näytettäessä täytyy jokainen osa valita erikseen. [Brusilovsky 1996], [Kobsa, Koenemann & Pohl 2001] Koottaessa sivua osista täytyy varmistaa, että osat muodostavat järkevä kokonaisuuden, mutta tällaista tehtävää suorittavan ohjelmiston toteuttaminen on vaikeaa.

Navigoinnin mukauttaminen

Navigoinnin mukauttamisella pyritään tukemaan käyttäjää tämän liikkeessa hyperavaruudessa. Käyttäjälle pyritään tarjoamaan mahdollisimman lyhyt reitti tämän etsimään informaatioon. Lisäksi on tärkeää, että käyttäjä tietää mihin nykyisestä solmusta voidaan siirtyä ja ymmärtää hyperavaruuden rakenteen. Näiden tavoitteiden saavuttamiseksi linkkejä voidaan poistaa, piilottaa, poistaa käytöstä, järjestää tai liittää niihin lisäinformaatiota. [Brusilovsky 1996]

Kobsa, Koenemann & Pohl (2001) huomauttavat, että navigoinnin mukauttamista suoritettaessa täytyy erotella toisistaan asiayhteyteen sidotut ja muut linkit. Asiayhteyteen sidotuilla linkeillä (engl. contextual links) tarkoitetaan linkkejä, joissa linkkinä toimii esimerkiksi tietty sana lauseessa. Tällaisen linkkin mukauttaminen saattaa vaikuttaa myös lauseen loppuosaan.

Poistamalla linkit käyttäjälle epäolennaisiin solmuihin voidaan hyperavaruuden osia piilottaa, ja näin vähentää käyttäjän kokemaa kognitiivista ylikuormitusta. Asiayhteyteen sidotuilla linkeillä ei poistamista kuitenkaan voida suorittaa, koska

tällöin ympäröivästä lauseesta jäisi osa puuttumaan. Nämä linkit voidaan sen sijaan piilottaa tai poistaa käytöstä (engl. disable). Linkki piilotetaan poistamalla se visuaalinen (tai muu) tehoste, jolla linkki erotetaan ympäröivästä tekstistä. Tällöin linkkiä voidaan vielä käyttää, mutta käyttäjän huomio ei kiinnity siihen tarpeettomasti. Linkin käytöstä poistaminen taas jättää linkin tehosteen ennalleen, mutta poistaa linkistä toiminnallisuuden. Tällöin käyttäjä ymmärtää, että hyperavaruus sisältää osia, joihin järjestelmä toistaiseksi evää käyttäjältä pääsyn. Koska tällainen toiminnallisuus (tai sen puute) on ristiriidassa yhtenäisen käytännön käytettävyyssperiaatteen (engl. consistency and standards, ks. [Nielsen 1994]) kanssa, tulisi linkin piilottamista ja käytöstä poistamista käyttää ainoastaan yhdessä. [Kobsa, Koenemann & Pohl 2001]

Linkkien järjestäminen voidaan suorittaa käyttäjän kiinnostuksen kohteiden tai tietyn linkin käyttötiheyden perusteella. Käyttäjälle parhaaksi seuraavaksi kohteeksi katsotun yksittäisen linkin korostamista visuaalisesti tai muuten etualalle tuomista kutsutaan suoraksi neuvonannoksi (engl. direct guidance). Asiayhteyteen sirottujen linkkien järjestäminen ei ole tarkoituksenmukaista. Lisäksi automaattisesti suoritettu järjestäminen saattaa hämmentää varsinkin kokemattomia käyttäjiä. [Brusilovsky 1996]

Tyypillisesti linkeihin liitettäviä lisäinformaatiota ovat kommentit ja erilaiset tehosteet. Kommentit tarjoavat lisäinformaatiota linkin toisessa päässä sijatsevasta solmusta. Tehosteita ovat erilaiset tekstitehosteet sekä kuvakkeet. Tekstitehosteilla voidaan esimerkiksi käyttäjälle suositeltavat linkit lihavoida ja ei-suositeltavat linkit näyttää himmeämmällä värillä. Kuvakkeita voidaan käyttää esimerkiksi verkkosivustolta muille verkkosivustoille johtavien linkkien yhteydessä, jolloin käyttäjä tietää poistuvansa alkuperäiseltä sivustolta. [Kobsa, Koenemann & Pohl 2001], [Brusilovsky 1996]

2.4 Web hypermediajärjestelmänä

Tunnetuin hypermediajärjestelmä on internetin päällä toimiva World Wide Web, jossa hypermediasolmuina toimivat verkkosivut. Webissä linkit voivat kulkea tietyn sivun sisällä tai eri sivujen välillä. Verkkosivut sisältävät erilaisia mediaelementtejä, kuten tekstiä, kuvia, video- ja äänitiedostoja, pieniä sovelluksia ja interaktion mahdollistavia elementtejä, kuten painikkeita ja lomakkeita. WWW tarjoaa globaalin, käyttäjän sijainnista riippumattoman, ympärivuorokautisen palvelualueen, joka mahdollistaa sisällön dynaamisen räätälöinnin pienillekin käyttäjryhmille [Kobsa, Koenemann & Pohl 2001].

Käyttäjämallin rakentamisen ja mukauttamisen kannalta webin keskeisin ominaisuus lienee järjestelmän hajautuneisuus. Tiedonkeruulle tämä hajautuneisuus näkyy siten, että yksittäisen sivuston mukautusjärjestelmä ei voi luotettavasti arvioida käyttäjän toimia tai tälle näytettyä sisältöä muilla sivustoilla. Käyttäjämallin rakentamista saattaa hankaloittaa myös se, että käyttäjät ovat kyllästyneet jatkuvaan käyttäjätiedon eksplisiittiseen keräämiseen. Käyttäjäkunnan laajuudesta johtuen oletusten tekeminen kerätyn tiedon perusteella on vaikeaa.

Mukauttamiselle, ja informaation esitystavalle yleisesti, webin hajautettu luonne aiheuttaa ongelmia laajan käyttäjäkunnan ja käyttöympäristöjen kirjon kautta. On haastavaa vastata kaikkien tarpeisiin, kun globaalin käyttäjäkunnan sisällä on merkittäviä eroja ja jopa ristiriitoja tavoitteissa, tiedoissa, taidoissa, kulttuuritaustoissa ja lukuisissa muissa ominaisuuksissa. Käytännössä tämä pakottaa suunnittelijan arvioimaan todennäköisimmät käyttäjäryhmät ja luomaan näiden tarpeiden perusteella sopivan kompromissiratkaisun.

Koska WWW:ssä käyttäjät voivat käyttää palvelua lähes mistä tahansa maailmassa, ja käyttö voi tapahtua hyvin erityyppisillä päätelaitteilla, parhaan mahdollisen palvelun tarjoamiseksi on otettava huomioon muutakin kuin käyttäjä henkilönä [Nieminen 2002]. Huomioimalla esimerkiksi fyysinen, tekninen ja sosiaalinen käyttöympäristö voidaan verkkosivun käyttökokemusta parantaa. Yleisesti fyysisestä ympäristöstä huomioitavia asioita ovat ympäristöolosuhteet, mutta webin tapauksessa tällaisen tiedon kerääminen on haasteellista. Kiinnostuksen kohteeksi ovatkin nousseet paikkatietoa hyödyntävät mukautukset ja sovellukset. Webin teknisen ympäristön muodostavat lukuisat laite- ja ohjelmistokokoonpanot. Erilaisia sosiaalisia ympäristöjä ovat esimerkiksi verkkopalvelun käyttö työn apuvälineenä työpaikalla, tiedonhankintaan koulussa yhdessä ystävien kanssa tai virkistäytymiseen kotona perheen kanssa.

2.4.1 Fyysinen ympäristö

Langattomat verkkoyhteydet ja mobiilit päätelaitteet ovat monipuolistaneet niitä fyysisiä ympäristöjä ja ympäristöolosuhteita, joissa webiä on mahdollista käyttää. Erot esimerkiksi ympäristön valoisuudessa, meluisuudessa, värinäissä ja lämpötilassa luovat uusia reunaehtoja verkkopalvelun käytölle. Olemassa olevat tekniset menetelmät fyysisistä ympäristöä koskevan tiedon keräämiseksi ovat nykyisellään kuitenkin melko rajoittuneet. Tästä johtuen automaattinen mukautuminen ympäristöolosuhteisiin on haastavaa: mahdollisuus käyttäjän aloitteesta tapahtuvaan mukautettavuuteen onkin käytännöllisempi ratkaisu.

Kun ymmärretään miten ympäristöolosuhteiden muutos vaikuttaa järjestelmän käyttämiseen, voidaan käyttöä tukea tarjoamalla sopivat mukauttamisvaihtoehdot. Näkemistä heikentäviä ympäristöolosuhteita varten voidaan kontrastia ja tekstikokoa suurentaa. Meluisissa tai hiljaisuutta vaativissa tiloissa on hyödyllistä korvata äänen välityksellä tarjottu informaatio visuaalisilla vastineilla. Tärisevässä tai kylmässä ympäristössä, jossa laitetta käytetään käsineet kädessä, voidaan välttää tarkkuutta vaativien syötteiden käyttöä esimerkiksi navigaatioissa. Edellä mainitut esimerkit voivat tietenkin palvella tiettyjä käyttäjäryhmiä myös normaaleissa ympäristöolosuhteissa.

Mikäli päätelaite pystyy ilmoittamaan sijaintinsa, voidaan tämä huomioida näytettävän sisällön valinnassa. Esimerkiksi käyttäjän etsiessä informaatiota ravintolasta, voidaan lista tuloksista painottaa käyttäjää lähellä olevien ravintoloiden mukaan. Lisäksi liikkeistään ilmoittava laite avaa mahdollisuuksia täysin uudenlaisille sovelluksille ja käyttötarkoituksille. Burrell, Treadwell ja Gay (2000) pohtivat esimerkiksi mahdollisuutta tiedottaa käyttäjille läheisistä tapahtumista, näyttää muiden käyttäjien tiettyyn sijaintiin liittämiä muistiinpanoja, ja tiedottaa käyttäjää samassa työtilassa saman aiheen parissa työskentelevistä muista käyttäjistä yhteistyön aikaansaamiseksi.

2.4.2 Tekninen ympäristö

Tekniikan kehittyessä web-selailun mahdollistavien laitteiden ja ohjelmistojen määrä on kasvanut räjähdysmäisesti. Laitteet eroavat toisistaan internet-yhteyden nopeudessa, näytön ominaisuuksissa, syötelaitteissa ja suorituskyvyssä. Ohjelmistot, eli selaimet, tukevat eri ominaisuuksia, toteuttavat standardeja vaihtelevasti ja niille on tarjolla joukko erilaisia liitännäisiä (engl. plug-in). Eri selainten lisäksi eroavaisuuksia on myös saman ohjelmiston eri versioiden välillä.

Koska WWW:n selailun mahdollistavat päätelaitteet ja ohjelmistot eroavat toisistaan teknisesti merkittävästi, on vaikeaa tuottaa yksi kaikille sopiva versio palvelusta. Yleisesti eritasoisia laitteiden ja ohjelmistojen yhdistelmiä voidaan palvella niin sanotulla asteittaisen lisäarvon-käytännöllä (engl. progressive enhancement). Ideana on tarjota palvelun sisältö ja perustoiminnot muodossa, jota yksinkertaisimmatkin tekniset ympäristöt osaavat hyödyntää. Tämän perustason päälle aletaan kerroksittain lisäämään lisäarvoa tuottavia ominaisuuksia käyttämällä tekniikoita, joita kehittyneemmät ympäristöt tukevat.

Vaikka tietyn tekniikan käytöstä saataisiinkin merkittävä hyöty, ei sitä aina voida hyödyntää, koska tekniikan käyttö saattaa hankaloittaa tai estää palvelun käy-

tön kokonaan niiltä käyttäjiltä, joilla kyseistä tekniikkaa ei ole käytössä. Perusteettomien oletusten tekemistä käytettävissä olevista tekniikoista tulisi välttää. Ilmaiseksi tarjolla olevien tekniikoiden hyödyntäminenkin ei ole itsestäänselvää. Esimerkiksi erilaiset liitännäiset voivat puuttua koska käyttäjä ei osaa, halua tai pysty, esimerkiksi puuttellisten käyttöoikeuksien takia, asentamaan niitä. Verkkosivujen dynaamista toimintaa voidaan merkittävästi parantaa ECMAScript-komentosarjakielillä (engl. scripting language). Vaikka yleisimmät selaimet tukevatkin ECMAScriptiä, ei sen saatavuuteen voida luottaa, sillä lukuiset ohjelmistot eivät sitä tue, käyttäjä on voinut kytkeä sen pois päältä (esimerkiksi tietoturvasyistä) tai komentosarja on voinut jäädä esimerkiksi yrityksen palomuriin jo ennen käyttäjän selaimen asti pääsemistä.

Selailuun käytettävä laitteisto luo erilaisia rajoitteita verkkopalvelun toteutukseen. Hidas yhteysnopeus saattaa merkittävästi vaikuttaa raskaan verkkopalvelun käyttökokemukseen. Näytöt eroavat toisistaan fyysiseltä kooltaan, erottelukyvyltään ja värien määrältä. Suurelle näytölle suunnitellun verkkopalvelun käyttö pieneltä näytöltä voi olla epämukavaa, ja vastaavasti pienille näytöille suunniteltu sivusto jättää hyödyntämättä suuren näytön tuoman lisätilan. Syötelaitteet vaihtelevat Qwerty-näppäimistön ja hiiren yhdistelmästä kosketusnäyttöjen kautta pienikokoiseen matkapuhelimen näppäimistöön.

Webin tapauksessa käyttöympäristön helpoiten ja luotettavimmin tunnistettava osa on tekninen ympäristö. Tästä johtuen se tarjoaa myös parhaat mahdollisuudet sisällön mukauttamiselle. Selaimen palvelimelle lähettämä HTTP-pyyntö (Hypertext Transfer Protocol) sisältää tunnistustietoa käytettävästä ohjelmistosta, ja tämän kautta voidaan päätellä myös käytössä olevaa laitteistoa (vrt. [Kobsa, Koenemann & Pohl 2001]). Ongelmatonta tekniseen ympäristöön mukautuminen ei kuitenkaan ole, sillä HTTP-pyyntön mukana lähetettävät tiedot eivät ole täysin luotettavia ja päätelmiin liittyy aina virheen mahdollisuus. Lisäksi tunnistuksen onnistuessaakin saadaan informaatiota vain laitteesta, eikä sen käyttäjästä, joita voi yhdellä laitteella olla useita.

2.4.3 Sosiaalinen ympäristö

Sosiaalisella ympäristöllä tarkoitetaan käyttötilanteeseen liittyviä tekijöitä, kuten yhteiskunnan, ryhmän ja sen yksittäisten jäsenien välisiä sosiaalisia suhteita, kulttuuria ja asenteita, tilaisuuden luonnetta ja niin edelleen. Esimerkiksi suositukset, joita käyttäjä haluaa vastaanottaa, riippuvat suuresti siitä, onko hän yksin kotonaan vai tärkeässä työtapaamisessa, jossa epäonnistuminen saattaa johtaa kasvojen menettämiseen.

Sen enempää aika kuin paikkakaan eivät rajoita webin käyttöä, joten se voi tapahtua toisistaan huomattavasti eroavissa sosiaalisissa ympäristöissä. Käytännössä verkkopalvelun tuottajan on haastavaa saada luotettavaa informaatiota sosiaalisesta ympäristöstä (ilman eksplisiittistä tiedon keruuta) ja reagoida siihen mielekkäällä tavalla.

2.4.4 Mukautuminen käytännössä

Tim Bernes-Lee (1996) suunnitteli webistä jaettua informaatioavaruutta, jonka kautta ihmiset ja koneet voisivat kommunikoida. Suunnittelussa painotettiin erityisesti seuraavia ominaisuuksia: Tietojärjestelmään tulisi voida tallentaa assosiaatioita objektien välille ja käyttäjien luomien assosiaatoiden tulisi olla muiden käyttäjien hyödynnettävissä. Informaation oikeellisuuden varmistamiseksi tiedon syöttämisen ja korjausten tekemisen tulisi olla helppoa. Käyttäjiä ei haluttu pakottaa käsittelemään informaatiota tietokoneiden tavoin. Lisäksi vältettiin käytetyn kielen tai käyttöjärjestelmän rajoittamista. Mukauttamiseen ei siis suunnitteluvaiheessa otettu kantaa, joten webin perustekniikat eivät juurikaan tue mukautuvien verkkopalveluiden toteuttamista. Käytännössä verkkopalveluiden mukautukset ovatkin verraten yksinkertaisia ja toteutettu webin perustekniikoiden päälle liitettyjen lisäteknikoiden avulla.

Kuten kohdassa 2.3 mainittiin, mukautusprosessi koostuu tiedon keruusta, käyttäjämallin rakentamisesta ja itse mukautuksen toteutamisesta. Luotettavimmin ja helpoiten tiedonkeruu webin kautta onnistuu eksplisiittisesti. Koska tämä rasittaa käyttäjää, tulisi kerättävä tieto rajoittaa sellaiseen aineistoon, jota todella hyödynnetään joko mukautuksessa tai muutoin.

Implisiittinen tiedonkeruu on webin perustekniikoilla hankalaa: käytännössä käyttäjän toimista tiedetään vain mitä tietosisältöä käyttäjälle on näytetty ja mitä linkkejä käyttäjä on päättänyt seurata. Erilaisia lisäteknikoita hyödyntämällä kattavampi implisiittinen tiedonkeruu on mahdollista, mutta kuten edellä mainittiin, näiden tekniikoiden käytössäolon kaikilla käyttäjillä ei voida luottaa. Webin tilat- tomasta luonteesta johtuen rajoittunutkin tiedonkeruu edellyttää käyttäjien tunnistamista ja kerätyn tiedon pysyvää tallentamista. Käytännössä tämä tapahtuu rekisteröitymis- ja sisäänkirjautumistoimenpiteillä. Samalla käyttöön saadaan myös edellisten käyttökertojen aikana kerätty, jo olemassa oleva informaatio.

Palvelinpäessä tarjolla olevat ohjelmointitekniikat mahdollistavat käyttäjämallin rakentamisen monimutkaisillakin päättelyketjuilla. Käytännössä mallin rakentamista siis rajoittaa tarjolla olevan käyttäjä- ja käyttötiedon määrä.

Koska jokaisen sivunlatauspyynnön yhteydessä voidaan tarjottava sivu tuottaa dynaamisesti palvelinpään tekniikoilla, ovat kaikki aiemmin mainitut mukautusmenetelmät hyödynnettävissä. Sisällön kannalta tämä tarkoittaa sitä, että pienemmistä sisältökokonaisuuksista voidaan piilottamalla, näyttämällä, valitsemalla ja järjestämällä koota toivotunlainen sisältökokonaisuus. Myös mediaelementtien korvaaminen paremmin käyttökontekstiin soveltuvaksi on mahdollista, mikäli kyseinen konteksti vain pystytään tunnistamaan. Vastaavasti linkkejä voidaan poistaa, piilottaa, poistaa käytöstä, järjestää tai liittää niihin lisäinformaatiota.

Käytännössä hyödynnettävät mukautukset ovat useimmiten melko yksinkertaisia. Yleisimmin sivu kootaan tietyn sivupohjasäännön mukaisesti. Sivupohjiin voidaan sisällyttää osioita, joihin valitaan tietty sisältö nykyisen käyttäjän käyttäjämallin perusteella. Koska tyypillinen käyttäjämalli koostuu ominaisuus-arvo -pareista, perustuu sisällön valinta yhden tai useamman ominaisuuden arvoihin yksinkertaisten jos-niin -sääntöjen kautta.

Luku 3

Verkkopalvelun sisällönhallinta

Tämä luku luo lyhyen katsauksen verkkopalveluiden sisällönhallinnassa tapahtuneeseen kehitykseen. Luku myös kartoittaa olemassa olevien sisällönhallintajärjestelmien ominaisuuksia ja käsittelee web-pohjaisten käyttöliittymien mahdollisuuksia ja rajoitteita sisällönhallintajärjestelmää luotaessa.

3.1 Verkkopalvelun sisällönhallinnan kehitys

McKeever (2003) määrittelee verkkopalvelun sisällönhallinnan webin kautta tarjotavan digitaalisen sisällön tuottamiseen ja jakeluun liittyvänä toimintana. Verkkopalvelun sisällönhallintajärjestelmä puolestaan koostuu tämän toiminnan automatisointiin ja tukemiseen käytettävistä ohjelmistoista. Goodwin ja Vidgen (2002) vastaavasti määrittelevät verkkopalvelun sisällönhallinnan ohjelmistoilla avustetuksi organisaation prosessiksi, jolla hallitaan webiin tarkoitettua monimuotoista sisältöä koko sen elinkaaren ajan luonnista tuhoamiseen. Molemmat määrittelyt puhuvat sisällönhallinnasta, eli pelkkä sisällön kertaluontoinen tuottaminen ei siis riitä, vaan kyseessä on jatkuva ylläpitoprosessi. Webin nuoresta iästä huolimatta tämän prosessin hallinnassa on otettu jo merkittäviä edistysaskeleita.

Webin alkuaikoina lähinnä tekstistä, hyperlinkeistä ja kuvista koostuvien verkkosivujen HTML-merkkäus kirjoitettiin käsin sopivalla tekstieditorilla. HTML-tiedostoja muokattiin joko suoraan palvelimella tai muokatut tiedostot siirrettiin palvelimelle FTP-ohjelman avulla. Yhteisön verkkosivun tuotannosta ja hallinnasta vastasi yleensä yksittäinen ”webmaster”, jonka tietotaito heijastui suoraan sivujen tekniseen laatuun. (Vrt. esim. [McKeever 2003].)

1990-luvun lopulla verkkosivujen hallinta tällaisella menettelyllä muuttui riittämättömäksi, kun sisällön ja käyttäjien määrä kasvoi, sisältö muuttui dynaamisemmaksi ja käytetyt tekniikat monipuolistuivat. Nykyään sivujen sisältöön kuuluu hypertekstin ja kuvien lisäksi muita mediatyyppejä. Lisäksi sivuihin liitetään ulkoasua määritteleviä tyyli-tiedostoja ja selainten toiminnallisuutta täydentäviä komentosarjatiedostoja. Sivujen sisältöä voidaan säilyttää staattisten tiedostojen sijaan esimerkiksi tietokannoissa ja itse sivut luoda palvelimella dynaamisesti jokaisen sivun latauspyynnön erityispiirteiden mukaan. (Vrt. esim. [McKeever 2003].)

Muutokset eivät kuitenkaan rajoitu pelkästään tekniseen kehitykseen, vaan myös tavassa tuottaa ja hallita sisältöä on tapahtunut merkittävä muutos: vastuu sisällön tuottamisesta on hajautettu webmasterilta sisällön aihealueen paremmin tunteville henkilöille. Näin ollen tekniikasta vastaava webmaster ei enää muodostu pullonkaulaksi, jonka kautta kaikki sisältö siirtyy sivustolle. Toisaalta sisältöä tuottavat käyttäjät eivät välttämättä tunne verkkosivujen toteutuksen teknistä puolta, joten tarvitaan helpokäyttöisiä työkaluja verkkopalveluiden sisällön hallintaan. Tähän tarkoitukseen onkin kehitetty suuri määrä erilaisia sisällönhallintajärjestelmiä. (Vrt. esim. [McKeever 2003].)

Nykyaikaisen verkkopalvelun tuottaminen on siis monimutkainen ja teknisesti hienostunut prosessi, joka vaatii useiden työkalujen, erilaisten merkkauk- ja ohjelmointikielten, protokollien, odotusten ja vaatimusten tuntemusta. Tästä johtuen WWW:n julkaisumalli on ollut perinteisesti epäsymmetrinen: sisällöntuottajat ovat olleet selkeästi erotettavissa lukijoista. [Di Iorio & Vitali 2005] Viime vuosina tässä on kuitenkin ollut havaittavissa selkeä muutos.

”Web 2.0” on noussut suosituksi termiksi kuvattaessa uuden tyyppisiä verkkopalveluja. Pääsääntöisesti termillä viitataan kahteen asiaan: ensinnäkin käyttäjien tuottamaan sisältöön ja sen ympärille muodostuviin sosiaalisiin verkostoihin, toisaalta kokoelmaan teknologioita, jotka mahdollistavat perinteisten työpöytäsovellusten tyylisten sovellusten tuottamisen verkkopalveluina. [Wilde 2007] Termillä ei ole vakiintunutta tarkkaa määritelmää, mutta Wilde mainitsee siihen yleisesti mielletävän muun muassa tietokeskeisyyden (engl. being data-centered) ja yhteisön tietämyksen (engl. collective intelligence) hyödyntämisen.

Kuten edellä mainittiin, kehittyneessä ”Web 1.0”-julkaisumallissa vastuu verkkosivuston teknisestä ylläpidosta ja sisällöntuotannosta on jaettu eri henkilöille. Vastaavasti sisällön julkaiseva organisaatio ja sisältöä lukeva käyttäjä ovat selkeästi erotettavissa toisistaan. Web 2.0:n tyypillinen ominaisuus on kaikkien käyttäjien valjastaminen sisällöntuottajiksi. Tämä sisällöntuottajien määrän räjähdysmäinen kasvu aiheuttaa tarpeen entistä helpokäyttöisemmille sisällönhallintatyökaluille.

Blogit ja wikit ovat kaksi eri käyttötarkoitukseen erikoistunutta sisällönhallintajärjestelmätyyppiä. Mielenkiintoisia blogeista ja wikeistä tekee se, että molemmissa sisällönhallinta tapahtuu yksinkertaisesti tavallisen verkkoselaimen kautta. Ehkä juuri tämän yksinkertaisuuden ja helppokäyttöisyytensä ansiosta ne ovat vakiinnuttaneet asemansa tavallisten käyttäjien keskuudessa. Seuraavassa tutkitaan näiden järjestelmien tyypillisiä ominaisuuksia ja pohditaan miten nämä ominaisuudet vaikuttavat verkkopalvelun sisällönhallintaan.

3.2 Blogit

Blogi (engl. weblog, blog) on jatkuvasti päivittyvä, yhden tai useamman kirjoittajan päivättyjä merkintöjä sisältävä verkkosivu. Tyypillisesti muut käyttäjät voivat tuottaa lisäsisältöä kommentoimalla näitä merkintöjä. [Lindahl & Blount 2003] Luonteenomaista blogeille on, että vanhat merkinnät säilyvät muuttumattomina ja luettavissa [Kilpi 2006]. Tavanomaisesti blogimerkinnöissä hyödynnetään hypermedialinkkejä viittaamalla runsaasti muihin verkkosivuihin ja blogimerkintöihin.

Elokuussa 1999 pieni amerikkalaisyhtiö Pyra Labs julkisti Blogger-nimisen verkkopalvelun. Blogger tarjosi sivuston, jossa kuka tahansa pystyi julkaisemaan oman bloginsa. [Kilpi 2006] Nykyään valinnanvaraa erilaisissa blogipalveluissa on runsaasti. Tekniikkaan perehtyneet käyttäjät voivat asentaa haluamalleen palvelimelle yhden monista blogeja pyörittävistä blogijulkaisujärjestelmistä. Riippumatta siitä pyöriikö blogijulkaisujärjestelmät palveluntarjoajan tai käyttäjän omalla palvelimella, tarjoavat ne erilaisia helppokäyttöisiä työkaluja sisällönhallintaan.

Useimmissa blogialustoissa sisältö voidaan syöttää joko niin sanotun WYSIWYG-editorin (engl. What You See Is What You Get) tai raakatekstieditorin kautta. WYSIWYG-editori on yksi keskeisimpiä käyttöä helpottavista ominaisuuksista niille käyttäjille, jotka eivät halua vaivata itseään teknisillä yksityiskohdilla. WYSIWYG-editorin vastaa toiminnaltaan esimerkiksi toimistosovelluksia, joissa käyttäjä voi muotoilla tekstiä graafisen käyttöliittymän avulla. Näin käyttäjä voi esimerkiksi lihavoida tai värjätä tietyn kohdan tekstistä napin painalluksella, ja editori tuottaa tämän perusteella tarvittavan HTML-merkkauksen. Nimensä mukaisesti editori siis näyttää sisällön jo muokkausvaiheessa samankaltaisena, kuin miltä se tulee julkaisun jälkeen näyttämään. Käytön helppoudella on kuitenkin oma hintansa, sillä usein WYSIWYG-editorilla tuotetussa merkkauksessa tekstielementtien semanttinen merkitys katoaa muotoilevan merkkauksen alle. Semanttinen, dokumentin rakennetta kuvaava merkkaukset on usein muotoilevaa merkkaukseta kevyempää ja helpommin tulkittavissa ei-graafisten selainten käyttäjille, hakukoneille, ja mukautuk-

sesta vastaavalle järjestelmälle. Raaketekstieditorin avulla merkkauksen hallitsevat käyttäjät voivat halutessaan tuottaa merkkauksen suoraan käsityönä tai muokata WYSIWYG-editorin tuottamaa merkkausta mieleisekseen.

Yleinen esikatselu-toiminto tarjoaa käyttäjälle mahdollisuuden nähdä, miltä kirjoitettu teksti tulee näyttämään sivupohjaan upotettuna. Näin tarvittavat muutoksen ja korjaukset voidaan tehdä välittömästi ennen tekstin julkaisua, jolloin loppukäyttäjät eivät vahingossa pääse näkemään virheellistä tekstiä.

Tyypillisesti sisällönhallintaprosessi toteutetaan blogeissa liittämällä jokaiseen artikkeliin tieto sen julkaisutilasta. Yleisesti käytettyjä tiloja ovat luonnos, hyväksyntää odottava ja julkaistu artikkeli, mutta ymmärrettävästi blogialustojen välillä on eroavaisuuksia. Luonnos-tilassa tallennettu artikkeli on ylläpitäjien nähtävillä blogin erillisessä ylläpito-osiossa, mutta ei näy muille käyttäjille. Näin artikkelia voidaan työstää vaiheittain, esimerkiksi useamman kirjoittajan toimesta. Usean käyttäjän blogeissa käyttöoikeukset voidaan hajauttaa siten, että vain osalla blogin ylläpitäjistä on oikeus julkaista artikkeleita. Muut ylläpitäjät voivat kirjoittaa artikkeleita ja asettaa ne odottamaan hyväksyntää. Blogista kokonaisuutena vastaavat ylläpitäjät voivat harkintansa mukaan julkaista nämä artikkelit. Vasta julkaistu artikkeli näkyy ylläpitäjien lisäksi muille käyttäjille. Blogialustat tarjoavat myös työkalut käyttäjätunnusten ja näihin liittyvien eri tasoisten käyttöoikeuksien hallintaan.

Artikkelien hallintaan liittyy tietysti kirjoittamisen ja julkaisun lisäksi muokkaus- ja poistotoiminnot. Artikkelien hallinnan lisäksi blogijärjestelmät tarjoavat välineet artikkeleihin liittyvien kommenttien hallintaan. Ylläpitäjä voi haluta hyväksyä kommentit ennen niiden julkaisua, jotta asiattomat viestit, kuten mainokset saadaan suodatettua pois asiallisten kommenttien joukosta.

Blogiartikkeleihin voidaan yleensä liittää avainsanoja tai ne voidaan luokitella kuuluvaksi tiettyihin aihepiireihin. Blogialusta tarjoaa käyttäjille mahdollisuuden hakea kaikki tietyn luokituksen saaneet artikkelit. Tämä selkeyttää blogin rakennetta ja parantaa informaation löytyvyyttä, kun käyttäjän ei tarvitse selata artikkeleita pelkästään julkaisuajan perusteella. Kirjoittajan työtä helpotetaan näyttämällä mitä avainsanoja hän ja muut kirjoittajat ovat aiemmissa artikkeleissa käyttäneet. Tämä auttaa yhtenäisen merkintätavan saavuttamisessa, jonka seurauksena käyttäjien ei tarvitse etsiä eri kirjoittajien samaa aihetta käsitteleviä, mutta eri avainsanoilla leimattuja artikkeleita.

Blogimoottorit erottavat toisistaan sisällön ja esitystavan. Tällöin sisältöä voidaan muokata ilman, että sen esitystapa muuttuu, ja päinvastoin. (vrt. [Lindahl & Blount 2003]) Käytännössä tämä tapahtuu hyödyntämällä sivupohjia (engl. templates), joihin jokaisen blogiartikkelin sisältö upotetaan. Koska esitysta-

pa on erotettu sisällöstä, ulkonäköön tehdyt muutokset vaikuttavat kaikkiin blogin artikkeleihin, myös niihin jotka on julkaistu ennen muutosta. Tyypillisesti blogipalvelut tarjoavat käyttäjille vaihtoehtoisia ulkoasuteemoja, joista ylläpitäjä voi valita mieleisensä. Näin käyttäjät voivat muuttaa bloginsa ulkoasua tietämättä mitään pohjalla toimivista CSS-tyyleistä. Cascading Style Sheets (CSS) on tyylimäärittelykieli, jonka avulla rakenteisiin dokumentteihin, kuten HTML-dokumentteihin ja XML-sovelluksiin, voidaan liittää ulkoasua koskevaa tietoa säilyttäen sisältö selvästi erillään esitystavasta (vrt. esim. [W3C 2007]).

Kilpi (2006) kuvaa blogeja merkittävänä teknologisenä, sosiaalisena ja taloudellisenä muutosvoimana, joka tekee lukijoista julkaisijoita ja kuluttajista tuottajia. Blogipalvelut ja -alustat tekevät webistä julkaisukanavan niille, joilla ei riitä aikaa, kiinnostusta tai muita resursseja laajan tekniikkaryppään haltuunottoon. Blogit tarjoavat valmiit perustoiminnot näille tavallisille käyttäjille, mutta samalla tarjoavat mahdollisuuden kattavaan kustomointiin niille, joilla taitoa ja kiinnostusta löytyy.

3.3 Wikit

Wikit ovat verkkosivustoja, joiden sisältö on kaikkien käyttäjien vapaasti muokattavissa. Nimi wiki on johdettu Hawaijin kielisestä termistä, joka tarkoittaa ”erittäin nopeaa”. Juuri nopeus on wikien keskeinen ominaisuus, sillä ne tarjoavat WWW-palvelun käyttäjille nopean ja helpon tavan tuottaa sisältöä. Wikejä käytetään esimerkiksi tietosana- ja sanakirjojen sekä erilaisten oppaiden luomiseen. Wikejä voidaan käyttää myös keskustelupalstojen ja postituslistojen korvaajana sekä erilaisten hankkeiden organisointiin ja dokumentointiin. [Ebersbach & Glaser 2004] Erilaisten blogijärjestelmien tapaan tarjolla on eri tekniikoilla toteutettuja ja erilaisilla ominaisuuksilla varustettuja wikijärjestelmiä. Myös oman wikin perustamisen mahdollistavia palveluita on olemassa.

Kaikkien käyttäjien vapaasti muokattavissa oleva sisältö luo luonnollisesti omat ongelmansa. Informaation luotettavuus on jatkuvasti kyseenalaistettavissa. Korkealaatuinen artikkeli saattaa muuttua useiden käyttäjien tekemien muutosten myötä epämääräiseksi rönsyileväksi sekasotkuksi. Käyttäjien eriävät mielipiteet voivat johdattaa artikkelin edestakaiseen muokkaamiseen. Ilkivaltatapauksissa käyttäjä voi poistaa sivuja tai tuottaa tahallisesti virheellistä sisältöä. Tällaiset ei-toivotut muutokset voidaan kumota palauttamalla sivun edellinen versio, mutta ilkivallan havaitseminen vaatii aina muiden käyttäjien työpanosta. (Vrt. [Ebersbach & Glaser 2004])

Ratkaisuna edellämainittuihin ongelmiin on rajata muokkaamiseen liittyvät käyttöoikeudet tietyille käyttäjryhmälle. Käyttöoikeuksien perus-

teella wikit voidaan jakaa kuuteen ryhmään ([Leuf & Cunningham 2001], [Chawner & Lewis 2006]:n mukaan). Täysin avoin wiki on kaikkien käytettävissä. Lukittava wiki on kaikkien luettavissa, mutta tiettyjen sivujen muokkaaminen on estetty. Aidattu (engl. gated) wiki puolestaan tarjoaa osan sivuistaan (mahdollisesti lukittuina) kaikkien nähtävillä, mutta loput sivut ovat vain rajatun käyttäjäryhmän nähtävillä. Astetta suljetumpi wiki tarjoaa kaiken materiaalin vain rekisteröidyille käyttäjilleen. Kaksi suljetuinta wikimuotoa rajaavat käyttäjäryhmiä entisestään, joko tiettyyn IP-osoitteiden ryhmään tai yksittäiselle tietokoneelle. Tarpeettoman rajoittavia käyttöoikeuskäytäntöjä on ymmärrettävästi syytä välttää, sillä wikien tyyppisten yhteisöllisten työkalujen vahvuudet nousevat esille vasta kun tekniset ja sosiaaliset rajoitteet niiden käyttöön on poistettu.

Alkuperäinen määritelmä wikille oli ”yksinkertaisin mahdollinen verkkotietokanta, joka voi toimia” [Ebert 2006]. Tämä näkyy wikien toiminnallisuudessa: toimintojen määrä on yleensä melko rajoittunut, minkä ansiosta wikien käyttö on yksinkertaista. Vaikka toteutettujen toimintojen kirjo vaihtelee wiki-järjestelmien välillä, löytyvät seuraavassa käsiteltävät yleisimmät toiminnallisuudet useimmista wiki-järjestelmistä.

Wiki on joukko dynaamisia hypermediasivuja, jossa käyttäjä voi muokata kaikkea näkemäänsä sisältöä [Ebersbach & Glaser 2004]. Käytännössä jokaisen sivun yhteydessä on linkki, jonka kautta kyseisen sivun sisältöä pääsee muokkaamaan. Myös uusien sivujen luonti on tehty helpoksi. Tyypillisesti käyttäjän etsiessä tai pyrkiessä siirtymään sivulle, jota ei ole olemassa, tarjotaan käyttäjälle mahdollisuutta uuden sivun luontiin. Lisäys- ja muokkaustoiminnot on yleensä toteutettu yksinkertaisesti hyperlinkkien ja HTML-lomakkeiden avulla, jolloin näiden käytön sisäistäneiden käyttäjien ei tarvitse opetella erillisten työkalujen käyttöä.

Hyperlinkit wikeissä voidaan jakaa karkeasti kahteen osaan: wiki-sivulta toiseen johtaviin sisäisiin linkkeihin, sekä muille verkkosivustoille johtaviin ulkoisiin linkkeihin. Sisäisten linkkien muodostamisesta on tehty yksinkertaista: korostamalla toisen wiki-sivun otsikko esimerkiksi kirjoittamalla se hakasulkeisiin tai isoilla kirjaimilla, se toimii automaattisesti linkkinä kyseiselle sivulle [Ebersbach & Glaser 2004]. Lisäksi tekstimuodossa kirjoitetut URL-muotoiset osoitteet tunnustetaan ja muutetaan automaattisesti toimiviksi linkiksi niin sisäisten kuin ulkoisten linkkienkin tapauksessa.

Kaikki sivuihin tehdyt muutokset tallentuvat järjestelmän tietokantaan. Tällöin jokainen koskaan luotu versio jokaisesta sivusta on tallessa ja tarvittaessa käyttäjien nähtävillä: käytännössä kaikki sivut ovat siis versionhallinnan alaisia. Versionhallinnan kautta käyttäjät voivat tarkastella tietyn sivun kehittymistä ajan myötä

ja tarvittaessa vanha versio sivusta voidaan palauttaa esimerkiksi ilkeävaltatapauksissa. Yksinkertaiset mahdollisuudet sivujen ja linkkien luontiin ja muokkaamiseen sekä versionhallintatoiminnallisuus muistuttavat läheisesti Bushin ja Nelson esittämiä ajatuksia hypertextistä [Ebersbach & Glaser 2004].

Perinteisesti käyttäjän on täytynyt opetella joko HTML-merkkausta tai kyseistä merkkausta tuottavan ohjelmiston käyttöä, jotta WWW-julkaisu on ollut mahdollista. Wikit ovat tyypillisesti WWW-pohjaisia, mutta sivujen muokkaaminen tapahtuu HTML:ää yksinkertaisemmalla merkkauskielellä [Ebersbach & Glaser 2004]. Piilottamalla teknisenä pohjana toimiva HTML-merkkaus käyttäjien tekninen tietotaito ei rajoita verkkopalvelun sisällöntuotantoa. Toisaalta esimerkiksi toimistotyökalujen WYSIWYG-käytäntöihin tottuneilta käyttäjiltä saattaa yksinkertaistetun merkkauksenkin sisäistäminen vaatia aikaa [Chawner & Lewis 2006].

Edellä mainittujen ydintoimintojen lisäksi useimmista wikeistä löytyy toimintoja, jotka helpottavat informaation löytämistä ja rohkaisevat yhteisöä tietosisällön tuottamiseen. Perinteisten hakujen lisäksi tyypillisiä informaation löytämisen tukitoimintoja ovat viimeksi muuttuneiden ja eniten katseltujen sivujen listaaminen sekä satunnaisen sivun näyttäminen. Käyttäjät voivat nopeuttaa tiettyä aiheutta käsittelevän sivun syntymistä lisäämällä aiheen niin sanotulle toivelistalle, jonka kautta muut käyttäjät voivat innostua luomaan kyseisen sivun. Niin kutsutut hiekkalaatikkoympäristöt ovat testisivuja, joissa käyttäjät voivat turvallisesti harjoitella wikin käyttöä muuttamatta varsinaisten sivujen sisältöä. Tyypillisiä ovat myös sivukohtaiset keskustelupalstat, joissa käyttäjät voivat suunnitella ja neuvotella kyseisen sivun sisällön kehittämisestä. (Vrt. esim. [Chawner & Lewis 2006], [Ebersbach & Glaser 2004].)

Wikien vahvuus tietosisällön tuotannossa on ehdottomasti käytön helppous, joka madaltaa kynnystä tuottaa sisältöä ja tätä kautta kasvattaa potentiaalisten sisällöntuottajien joukkoa. Mikäli tietosisältöä tuottavaa käyttäjäryhmää halutaan rajata — esimerkiksi informaation luotettavuuden varmistamiseksi — ei kaikkia wikien menettelytapoja voida hyödyntää. Seuraavat toiminnot kuitenkin tukevat tietosisällön syöttämistä web-käyttöliittymän kautta myös muunlaisissa ympäristöissä.

Wikeissä käytetyt, muokattavan tiedon välittömään läheisyyteen sijoitetut muokkausnäkympään johtavat linkit ovat intuitiivisia: käyttäjä tietää välittömästi mitä tietosisältöä tämä on muokkaamassa, ja missä muokkauksen vaikutukset tulevat näkympään. Käyttäjän ei siis tarvitse etsiä tiettyä tiedostoa tai tietokantataulua palvelimelta, vaan ainoastaan informaatio, jota haluaa muokata. Tehdyt muutokset näkyvät palvelussa välittömästi, joten mahdolliset virheet voidaan korjata nopeasti.

Wikien tavalla tallentaa jokainen versio tietosisällöstä on hyvät ja huonot puolensa. Vanhan version palauttaminen voi säästää käyttäjää ylimääräiseltä työltä inhimillisen virheen, eri käyttäjien samanaikaisten muokkauksen tai ilkivallan tapauksessa. Toisaalta versionhallinnan toteuttaminen on työlästä ja kaiken tiedon tallentaminen kasvattaa tietokannan kokoa. Suurimmat hyödyt versionhallinnasta saadaan käsiteltäessä informaatiota joka muuttuu usein: melko pysyvän informaation tapauksessa hyödyt jäävät vastaavasti pienemmiksi.

Yksinkertaistettu merkkkaus soveltuu oivallisesti tilanteisiin, joissa sisällöntuotannosta vastaavat omien alojensa ammattilaiset, jotka eivät osaa tai ole kiinnostuneita oppimaan HTML:ää tai muuta vastaavaa merkkaukieltä. Näin käyttäjät voivat keskittyä itse sisältöön, ja antaa järjestelmän vastata merkkauksen tuottamisesta. Vaihtoehtoisesti erillinen henkilö voisi vastata asiantuntijoiden tuottaman raakatekstin muokkaamisesta WWW-julkaisuun soveltuvaan muotoon, mutta tämän välikäden poisjättäminen yksinkertaistaa ja myös nopeuttanee julkaisuprosessia. Vaikka yksinkertaistetulla merkkauksella voidaan tuottaa kevyesti rakenteista tietoa, mielekkäisiin mukautustoimintoihin sillä tuotettu materiaali ei välttämättä kelpaa. Mukauttamiseen soveltuvan rakenteisen tiedon tuottaminen siis edellyttää sisällöntuotajalta ylimääräistä työtä, jota ei voida kevytmerkkauksen avulla vähentää.

3.4 Web-pohjaisen käyttöliittymän erityispiirteet sisällönhallinnassa

Eloisan verkkopalvelun edellytyksenä on uuden sisällön lisääminen ja vanhan ylläpitäminen [Goodwin & Vidgen 2002]. Sisällönhallinnan tapahtuessa blogien ja wikien tapaan web-pohjaisen käyttöliittymän kautta, heijastuvat käytettävissä olevien web-tekniikoiden vahvuudet ja heikkoudet sekä sisällönhallintajärjestelmään että sisällönhallintaprosessiin. Seuraavassa tutkitaan millaisia reunaehdoja web-tekniikat aiheuttavat sisällönhallintajärjestelmän toteuttamiselle sekä millaisia mahdollisuuksia ja uhkia näistä reunaehdoista seuraa.

3.4.1 Tekniset vahvuudet ja heikkoudet

Koska erilaiset verkkopalvelut ovat muodostuneet osaksi ihmisten arkipäivää, voidaan olettaa verkkoselaimen ja verkkosivujen peruselementtien kuten lomakkeiden toiminnallisuuden olevan tuttuja tavalliselle käyttäjälle. Verrattuna muihin käyttäjien tyypillisesti tuntemiin ohjelmistoihin kuten toimistosovelluksiin, soveltuu tarkoitusta varten suunniteltu verkkopalvelu paremmin sisällönhallintaan. Esimerkiksi

Microsoft Wordin tyyppisellä toimistosovelluksella on rakenteisen tiedon tuottaminen hankalaa. Taulukkolaskentaohjelmistolla tuotettu rakenteinen tieto taas täytyy erikseen muuntaa muotoon, joka sopii paremmin palvelimelle varastoitavaksi ja verkkopalvelussa julkaistavaksi.

Tuotettaessa sisältöä web-käyttöliittymän kautta voidaan sisältö tallentaa suoraan palvelun tietokantaan, jolloin erillisiä työkaluja tietosisällön vientiin ja tuontiin ei tarvita. Tästä hyödytään varsinkin sisällön ollessa jatkuvasti muuttuvaa, koska prosessi, jolla tietosisältö palveluun tuodaan voi olla verraten raskas. Suoraan palveluun tuotettu sisältö on myös välittömästi käyttäjien nähtävillä. Toisaalta jos muutokset tallentuvat suoraan tuotantopalvelimelle, täytyy huomioida tietosisällön säilyvyys virhetilanteissa.

HTML on alkujaan suunniteltu dokumenttien rakenteen kuvailuun ja HTTP tällaisten dokumenttien siirtämiseen. Tästä johtuen näiden tekniikoiden päälle rakennetut sovellukset eivät pärjää perinteisille työpöytäsovelluksille vasteajoissa, vuorovaikutuksen monipuolisuudessa ja graafisessa ulkoasussa. Esimerkiksi verkkopalvelun vuorovaikutus perustuu pääasiassa HTML-lomakkeiden käyttöön. Perustoinnallisuuden, kuten syötteiden virhetarkastelun, virheviestien ja esitäytettyjen lomakkeiden esittämiseksi täytyy HTML-lomakkeiden toiminnallisuutta täydentää palvelinpään ohjelmoinnilla. Puutteiksi voidaan katsoa myös jatkuva internetyhteyden tarve sekä rajoittuneet mahdollisuudet käsitellä käyttäjän koneella sijaitsevaa tietoa.

Verkkopalveluiden ja työpöytäsovellusten välistä eroa voidaan kaventaa erilaisien lisäteknikoiden avulla. Näitä tekniikoita voidaan kuitenkin hyödyntää vain lisäarvoa tuottavien lisätoimintojen toteuttamisessa, sillä sisällön ja perustoimintojen saatavuus on turvattava myös niille käyttäjille, joilla kyseiset tekniikat eivät syystä tai toisesta ole käytettävissä.

Heikon vuorovaikutuksen lisäksi useat web-teknologioiden aiheuttamat rajoitteet näkyvät mitä tahansa verkkopalvelua toteutettaessa. Esimerkiksi HTTP:n tilattomuus, tietoturvanäkökohdat ja merkistöongelmat aiheuttavat lisätyötä verkkopalvelua toteutettaessa. Näiden ongelmien nujertaminen syö luonnollisesti resursseja itse sisällönhallintaan liittyvien toimintojen toteuttamiselta. Tätä ongelmaa voidaan kiertää hyödyntämällä verkkopalveluiden yleisimmät toiminnallisuudet valmiiksi toteutettavaa verkkosovelluskehystä.

3.4.2 Mahdollisuudet ja uhat sisällönhallinnalle

Web-pohjaisen sisällönhallintajärjestelmän merkittävimmät mahdollisuudet saavutetaan järjestelmän hajautuneisuudesta. Useat käyttäjät voivat osallistua sisällöntuotantoon sijainnistaan riippumatta. Toisaalta käyttö vaatii aina verkkoyhteyden, jota ei välttämättä ole saatavilla. Koska sisällönhallinta tapahtuu ilmaiseksi saatavilla olevien verkkoselainten kautta, ei sisällönhallintaan tarvitse hankkia erillisiä, mahdollisesti kalliitakin ohjelmistoja.

Kuten edeltävä lyhyt katsaus blogeihin ja wikeihin osoitti, on verkkopalveluihin mahdollista toteuttaa ominaisuuksia, jotka eristävät käyttäjän teknisestä toteutuksesta. Kun käyttäjän ei tarvitse tuntea järjestelmän alla toimivia tekniikoita, kuten HTML:ää, tämä voi keskittyä itse sisällön tuottamiseen [Lindahl & Blount 2003].

Vaikka sisällönhallinnan hajauttaminen useammalla käyttäjälle ideaalisesti tehostaa sisällönhallintaprosessia, liittyy siihen omat vaaransa. Useiden tahojen lisäessä sisältöä verkkopalveluun ilman koordinoitua, sivuston informaatioarkkitehtuuri rämettyy helposti, jolloin informaation löytäminen ja sivuston rakenteen hahmottaminen vaikeutuu (vrt. esim. [Goodwin & Vidgen 2002]). Koordinoitua edellyttää myös laadultaan, tyylyltään, käytännöiltään ja ulkoasultaan yhtenäisen materiaalin tuottaminen. Ilman koordinoitua eri tahot voivat lisätä samaa sisältöä sivuston eri osiin, jolloin sisällön ylläpitäminen vaikeutuu. On myös tärkeää huomioida onko kaikilla sisällöntuottajilla oikeus julkaista tuottamansa sisältö, vai täytyykö sisältö katselmoida ennen sen julkaisemista.

Tietosisällön syöttömekanismeja suunniteltaessa sekä sisällöntuottajia ohjeistettaessa tulee huomioida sisällön tuleva käyttötarkoitus. Mikäli sisältöä aiotaan mukauttaa, täytyy sen olla riittävän rakenteista. Blogeissa ja wikeissa ei tyypillisesti ole kiinnitetty merkittävästi huomiota sisällön rakenteisuuteen, koska mukautuspäätelyn puuttuessa tälle ei ole tarvetta. Vaikka rakenteisen sisällön tuottaminen web-käyttöliittymän kautta on siis mahdollista, ei se synny itsestään, vaan edellyttää tarpeen tiedostamista ja ylimääräistä työtä sisällöntuottajilta.

Luku 4

Tietomallin vaikutus sisällöntuotanto- ja mukauttamisprosesseihin

Tässä luvussa käsitellään tiedon mallintamista lyhyesti yleisellä tasolla ja esimerkkitapausten kautta tarkemmin ruokaohjeiden osalta. Lisäksi pohditaan millaisia edellytyksiä mukautuminen aiheuttaa tietomallille sekä miten tietomalli puolestaan vaikuttaa sisällönhallintaan.

Tässä luvussa tiedon mallintamisella tarkoitetaan verkkopalvelun tietosisällön kuvaamista sellaisessa muodossa, että sen varastoiminen ja tehokas käsittely on mahdollista koneellisesti. Tietosisältöön voidaan katsoa kuuluvaksi itse välitettävän informaation lisäksi myös sellainen metatieto, joka vaikuttaa siihen, miten muuta tietosisältöä käytetään.

4.1 Tiedon mallintaminen

Tiedon mallintaminen luo pohjan tietosisällön merkityksen yksikäsitteiselle määrittelylle ja tulkinnalle, jotka puolestaan ovat edellytyksiä sille, että tietosisältöä käsittelevät tahot voivat kommunikoida tehokkaasti keskenään. Lisäksi tiedon mallintaminen mahdollistaa eri tietoalkioiden yhdistelyn ja vertailun koneellisesti. Verkkopalveluissa tällaista koneellista tiedonkäsittelyä hyödynnetään esimerkiksi sisällön mukauttamisessa ja sivupohjajärjestelmän toiminnassa.

Moodyn ja Shanksin (2003) mukaan sopivan tiedon esitystavan valitseminen on tärkeimpiä tehtäviä uutta järjestelmää kehitteässä, sillä vaikka tiedon mallinta-

minen on vain pieni osa koko kehitystyöstä, on sen vaikutus lopputuloksen kannalta todennäköisesti muita yksittäisiä vaiheita suurempi. Tietomallin valinnan onnistuminen tai epäonnistuminen heijastuu järjestelmän kehityskuluihin, joustavuuteen, yhteensopivuuteen sekä järjestelmän kykyyn vastata käyttäjiensä tarpeita. [Moody & Shanks 2003]

Moody ja Shanks (2003) määrittelevät laadukkaalle tietomallille seuraavat kahdeksan ominaisuutta:

Tekninen virheettömyys (engl. correctness) Tietomalli noudattaa käytetyn mallinnustekniikan sääntöjä.

Kattavuus (engl. completeness) Tietomallin avulla pystytään esittämään kaikki informaatio, jota järjestelmän toiminnallisuuden toteuttaminen vaatii.

Eheys (engl. integrity) Tietomalli on eheä jos kaikki tietosisältöön liittyvät liiketoimintasäännöt (engl. business rules) on määritelty.

Yksinkertaisuus (engl. simplicity) Tietomalli ei sisällä tarpeettomia entiteettejä tai näiden välisiä suhteita.

Joustavuus (engl. flexibility) Joustavuus määrittelee sen, miten helposti tietomalli selviytyy muiden järjestelmän osien muuttumisesta johtuvista muutoksista.

Yhteneväisyys (engl. integration) Hyvä tietomalli on yhteneväinen organisaation muiden tietojärjestelmien kanssa.

Ymmärrettävyys (engl. understandability) Ymmärrettävyydellä tarkoitetaan sitä, miten helposti tietomallin rakenteita ja käsitteiden merkitystä pystytään tulkitsemaan, esimerkiksi toteutusryhmän ulkopuolisten henkilöiden toimesta.

Toteutettavuus (engl. implementability) Lopulta hyvä tietomalli yhdistää edellämainitut ominaisuudet käytettävissä olevan ajan, teknologian ja muiden resurssien puitteissa toteutuskelpoiseksi kokonaisuudeksi.

Teknistä virheettömyyttä ja toteutettavuutta voitaneen pitää lähes itsestäänselvyytenä laadukkaasta järjestelmästä puhuttaessa. Ominaisuuksista ehkäpä tärkein on kuitenkin kattavuus, sillä liian rajoittava tietomalli vaikuttaa suoraan siihen mitä järjestelmällä voidaan tehdä ja miten se vastaa käyttäjänsä tarpeita. Tietomallin joustavuus on erityisen tärkeää kun järjestelmää toteutetaan ketterillä ohjelmistokehitysmenetelmillä, joille jatkuvat muutokset järjestelmän kehitysvaiheessa ovat tyyppillisiä. Yhteneväisyys on huomioitava erityisesti rakennettaessa uutta järjestelmää

organisaation olemassa olevien järjestelmien rinnalta. Toisaalta tällaisessa tilanteessa täytyy arvioida onko yhteensopivuudesta saatava hyöty sen toteuttamisesta aiheutuvan vaivan arvoista.

Edellä listatut ominaisuudet tarkastelevat tietomallia yleisellä tasolla ottamatta kantaa yksityiskohtiin. Tällaisen tarkistuslistan käyttö puutteiden löytämiseksi on suositeltavaa tietomallin suunnittelun loppuvaiheessa, mutta se ei tarjoa apua tietomallin yksityiskohtien suunnitteluun. Yksityiskohtien huomioiminen ja tarpeiden ennakointi on tietomallin suunnittelun vaikein osuus, sillä juuri yksityiskohdat lopulta määrittelevät, onko tietomalli käyttötarkoitukseensa sopiva. Yksityiskohtien suunnitteluun on vaikeaa antaa yleispäteviä ohjeita, koska ne riippuvat täysin suunniteltavasta järjestelmästä. Apua suunnitteluprosessiin voidaan saada esimerkiksi tutkimalla millaisia tietomalleja muissa vastaavissa järjestelmissä on käytetty. Koska toteuttamassamme verkkopalvelussa ruokaohjeet ovat keskeisessä roolissa, tutkitaan seuraavassa esimerkinomaisesti miten niitä on mallinnettu erilaisissa tietojärjestelmissä.

4.2 Olemassa olevat ruokaohjemallit

Perinteisesti ruokaohjeet on kirjoitettu käsin tai painettu paperille. Tällöin kirjoittajalla on valta valita erikseen jokaisen ohjeen tarpeisiin sopiva rakenne ja kirjaamisen tarkkuustaso, jotka saattavat vaihdella suuresti eri ohjeiden välillä. Kirjoittaminen on yleensä nopeaa ja helppoa, mutta tuotettu teksti on heikosti silmäiltävää. Lisäksi teksti heijastaa kirjoittajansa tyyliä, ja samaa ruokalajia kuvaavat eri kirjoittajien tuottamat ohjeet voivat vaihdella lyhyestä yleisluontoisesta kuvauksesta pitkään, yksityiskohtaiseen rautalankaohjeeseen.

Paperille tuotetut ruokaohjeet kirjoitetaan tyypillisesti osittain rakenteiseen muotoon: ruokaohjeesta eritellään ainakin valmistusaine- ja valmistusvaihelistat. Esimerkiksi valmistusaineen tapauksessa yksittäinen valmistusaine voidaan kuitenkin puolirakenteisessa muodossa kuvata lukuisilla tavoilla. Valmistusaine voi koostua useammasta ruoka-aineesta, joilla on useita synonyymeja, tai valmistusaine voidaan ilmoittaa yleisemmällä kategoriatasolla. Esimerkiksi valmistusaineen ollessa makkara on vaikeaa ilmaista tarkasti mitä kaikkia ruoka-aineita ohjeeseen tulee, koska eri valmistajien makkarat sisältävät eri ruoka-aineita. Valmistusaineiden merkkauksellisuuden monimuotoisuutta lisäävät myös valmistusaineen määrä ja muoto. Määrä voidaan ilmaista yksiselitteisesti (100g makkaraa) tai epämääräisemmin (paketti makkaraa, yksi iso makkara). Lisäksi valmistusaineisiin liittyy joukko olomuotoja (pil-

kottu, raastettu, murskattu) ja käyttötarkoituksia (koristeluun, mausteeksi). Kaikki olomuodot ja käyttötarkoitukset eivät ole mielekkäitä kaikille valmistusaineille.

Vaikka ruokaohje olisikin helppoa ja nopeaa kirjoittaa tietokoneella vapaamuotoisena tekstinä, ei tämä ole optimaalinen ratkaisu verkkojulkaisemisen kannalta. Koska kone ei pysty ymmärtämään vapaamuotoisen tekstin merkitystä, on sillä rajalliset mahdollisuudet käsitellä tätä tietosisältöä. Esimerkiksi jos ruokaohjeen valmistusaineet on kirjattu vapaatekstinä, on valmistusaineiden perusteella ohjeet luokittelevan ohjelman toteuttaminen haastavaa, koska eri kirjoittajat saattavat käyttää valmistusaineesta eri termiä tai kirjoittaa sen eri taivutusmuodossa. Tällöin kattavakin informaatiovarasto jää Bushin ajatusten mukaisesti hyödyntämättä puutteellisten haku- ja selausmahdollisuuksien takia.

Tietosisällön rakenteinen mallintaminen luo pohjan myös sen mukauttamiselle. Esimerkiksi verkkopalvelu, joka haluaa suodattaa käyttäjälle soveltumattomat ruokaohjeet pois näkyvistä, tarvitsee riittävän tarkat mallit käyttäjän ruokavaliosta ja ruokaohjeista. Mikäli ohjelmisto ei sovellu valmistusaineiden ja käyttäjän ruokavaliossa määriteltyjen allergisoivien aineiden välisen yhteyden käsittelyyn, ei tällainen mukauttaminen ole mahdollista. Käytännössä kaikki luvussa kaksi esitetyt tekstiä koskevat mukauttamismenetelmät vaativat toimiakseen tiedon siitä, mitä teksti itseasiassa tarkoittaa. Vaikka tekstin merkitys olisi ruoanlaittommikolle itsestään selvä, täytyy asioiden merkitys määritellä koneelle yksikäsitteisesti.

Kuten edellä mainittiin, voivat eri käyttäjien samasta ruokalajista tekemät ruokaohjeet erota toisistaan suuresti. Tarkka mallinnus luo hyvät lähtökohdat myös yhtenäiselle merkkaukselle, mutta viime kädessä vastuu on kuitenkin käyttäjällä.

Yleisesti ottaen tiedon tarkka mallinnus on hyödyllistä, mutta on kuitenkin huomioitava, että liian tarkka mallinnus tekee sisällöntuotannosta tarpeettoman työlästä. Tärkeää onkin löytää sopiva tasapaino tiedon rakenteisuudesta saatavan hyödyn ja sen tuottamisesta aiheutuvan vaivan välillä. Lisäksi liian pikkutarkka mallinnus saattaa asettaa tarpeettomia rajoitteita sisältöä tuottaessa. Esimerkiksi valmiiksi annetut vastausvaihtoehdot aiheuttavat ongelmia tiedon syöttäjän kokiessa kaikki vaihtoehdot täysin sopimattomiksi.

Loppuosa tästä kohdasta käsittelee kolmea eri tahoilla suunniteltua lähestymistapaa ruokaohjeen mallintamiseen. BigOven, jota käsitellään ensimmäisenä, vaikeuttaa tietomallinsa osalta vähimmäisvaivalla luodulta ja uusien tarpeiden ilmaantua laajennetulta ratkaisulta. RecipeML näyttäisi olevan alusta asti suunniteltu kattavaksi, yksikäsitteiseksi ja käytännössä toimivaksi ruokaohjemalliksi, kun taas Dish_Masterissa on keskitytty toiminnallisuudella teoretisointiin ja jätetty konkreettiset tekniset yksityiskohdat taka-alalle.

4.2.1 BigOven

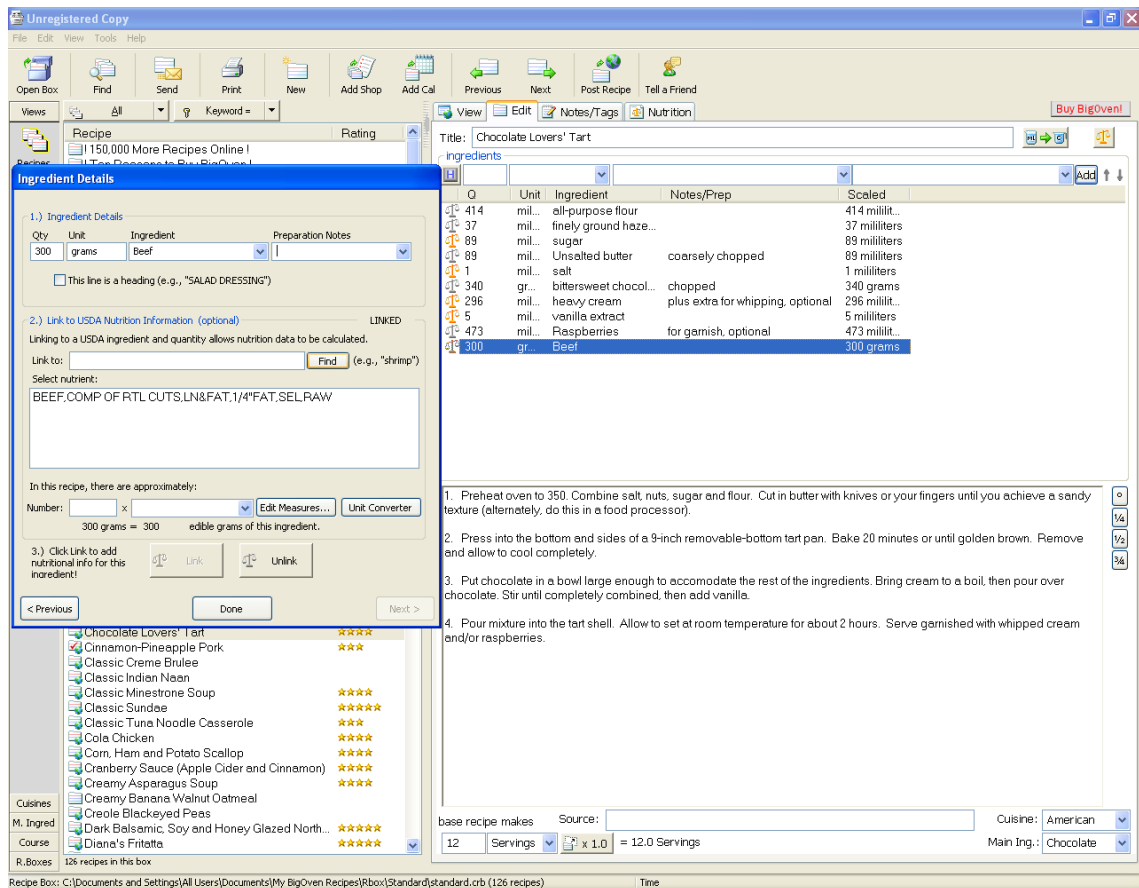
BigOven (ks. <http://www.bigoven.com/>) on yhdysvaltalaisen Lakefront Software -yhtiön ruokaohjeiden hallintaan tarkoitettu kaupallinen ohjelmisto. Ohjelmistosta on tarjolla ilmainen kokeiluversio, johon tämän kohdan huomiot perustuvat.

BigOvenin keskeisimmät toiminnot ovat ruokaohjeiden hallinta (tuonti, luominen, lajittelu, vienti), ruokalistojen suunnittelu ja ostoslistojen luominen ohjeiden perusteella. Lisäksi ohjelmisto sisältää lukuisia aputoimintoja, kuten mittayksiköiden muuntaminen metrijärjestelmän ja brittiläisen mittajärjestelmän välillä, ruokaohjeiden skaalaaminen eri annosmäärille, ruokaohjeen ravintoarvojen laskeminen sekä sopivien ruokaohjeiden etsiminen käytettävissä olevien valmistusaineiden perusteella. Ohjelmistoon liittyy läheisesti verkkosivusto, jossa käyttäjät voivat julkaista omia ruokaohjeitaan ja ruoanlaittovideoitaan, sekä keskustella ruoanlaittoon ja ohjelmistoon liittyvistä aiheista.

BigOvenissa uuden ruokaohjeen luominen aloitetaan antamalla ruokaohjeelle nimi, kertomalla ohjeesta saatavien annosten määrä ja liittämällä ohjeeseen vapaamuotoinen kuvausteksti. Minkäänlaisia ohjeita tai avusteita ei tässä vaiheessa käyttäjälle tarjota, joten yhtenäisen merkkaustavan saavuttaminen on käyttäjien vastuulla.

Valmistusaineiden syöttäminen tapahtuu neljän syöttökentän kautta. Ruudunkaappauskuvassa 4.1 syöttökentät näkyvät oikeassa reunassa valmistusainelistan yläpuolella. Syöttökenttien tarkoitusta ei kerrota, vaan käyttäjän pitää päätellä tämä alla olevan taulukon sarakkeiden otsikoista ja kentissä tarjolla olevista vaihtoehtoista. Ensimmäiseen kenttään tulee syöttää valmistusaineen määrä kokonaislukuna. Desimaaliluvut järjestelmä pyöristää lähimpään kokonaislukuun. Kenttä hyväksyy myös vapaamuotoisen tekstin, mutta lopputuloksena valmistusainelistaan merkitään valmistusaineen määräksi pelkkä nolla. Loppukäyttäjälle tämä näkyy kuin valmistusaineen määrää ei olisi määritetty. Arvoväliä (esimerkiksi "1-2") syötettäessä järjestelmä ei huomioi väliviivaa ja sitä seuraavan lukua mitenkään. Kaikki yllämainitut valmistusmäärään automaattisesti tehtävät muutokset suoritetaan käyttäjää asiasta huomauttamatta, jolloin mahdolliset virheet jäävät helposti havaitsematta.

Toista kenttää käytetään määrän yksikön määrittämiseen. Käyttäjä voi valita yksikön pitkästä, 65 yksikköä sisältävästä pudotusvalikosta. Vaihtoehtoisesti käyttäjä voi kirjoittaa yksikön vapaamuotoisena tekstinä. Kenttään voi kirjoittaa pitkänkin yksikön, mutta käyttäjää asiasta huomauttamatta järjestelmä automaattisesti katkaisee syötteen 20 merkin kohdalta. Käyttäjän kirjoittamia yksiköitä ei myöskään talleteta yksikkölistaan, vaan käyttäjä joutuu kirjoittamaan nämä yksiköt kerta toisensa jälkeen käsin.

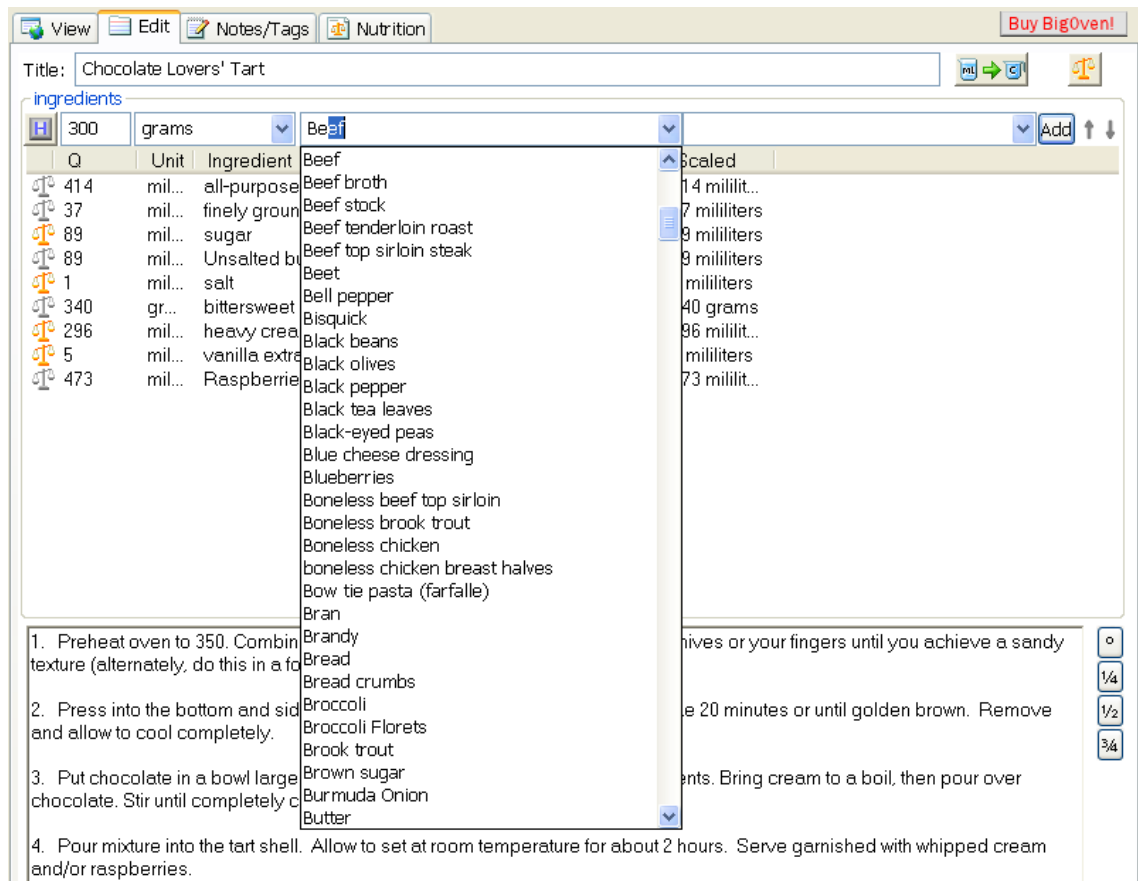


Kuva 4.1: BigOven-ohjelmiston näkymä ruokaohjeen muokkaamisesta

Itse ruoka-aine syötetään kolmanteen kenttään. Käyttäjä voi valita ruoka-aineen runsaan 400 aineen pudotusvalikosta. Vaihtoehtoisesti käyttäjä voi kirjoittaa kenttään muutaman ensimmäisen kirjaimen, jolloin kenttä ehdottaa sopivia ruoka-aineita, kuten kuvassa 4.2 esitetään. Mikäli käyttäjän syöttämää ainetta ei löydy listalta, ehdottaa järjestelmä sen tallentamista, jolloin aine on seuraavalla kerralla käytettävissä. Jos valmistusainelistaan lisätään valmistusaineita, joiden ruoka-ainetta ei ole määriteltä, huomautetaan tästä käyttäjälle.

Viimeiseen kenttään voidaan syöttää valmistusaineen olomuoto (pilkottu, raastettu tms.) tai muu vapaamuotoinen seliteteksti. Valmiita olomuotoja on tarjottu 26 kappaletta, ja kuten mittayksiköiden tapauksessa, käyttäjän omia syötteitä ei talleteta listaan myöhempää käyttöä varten.

Valmistusaineen lisäämisen jälkeen käyttäjä voi halutessaan luoda linkin valmistusaineen ja sen ravintoarvojen välillä. Linkin luominen tapahtuu kuvassa 4.1 vasemmassa näkyvän ikkunan kautta. Lähteenä ravintoarvoille on käytetty Yhdysvaltain maatalousministeriön (USDA) tarjoamaa tietokantaa. Jos kaikille valmistusaineil-



Kuva 4.2: BigOven-ohjelmiston näkymä sopivan ruoka-aineen ehdottamisesta

le määrittellään tällainen ravintoarvolinkitys, voi ohjelma laskea koko ruokaohjeen sisältämän energian ja ravintoaineiden määrän.

Valmistusainelistaan voidaan lisätä myös otsikoita, ja näin jakaa ruokaohje eri työvaiheisiin, kuten leivonnaisen pohjaan, täytteeseen ja kuorrutukseen. Ruokaohjetta katseltaessa näitä otsikoita ei korosteta mitenkään, vaan ne hukkuvat helposti muiden valmistusaineiden sekaan. Käyttäjä voi korostaa otsikkoa käyttämällä esimerkiksi isoja kirjaimia tai erikoismerkkejä, mutta tällöin yhtenäisen merkintätavan sopiminen on jälleen kerran käyttäjien vastuulla.

Valmistusvaiheiden kirjaamiseen tarjotaan yksi suuri tekstikenttä, joka näkyy kuvassa 4.1 oikeassa alareunassa. Käyttäjällä on vapaus ja vastuu kirjoittaa valmistusvaiheet parhaalla katsomallaan tavalla. Käytännössä koko ohjeen voi kirjoittaa yhteen pitkään tekstikappaleeseen, tai sen voi jakaa numeroiduille riveille työvaiheittain.

Ruokaohjeeseen voi liittää joukon lisätietoa, kuten ohjeen lähteen, vapaamuotoisen huomautuksen ja ohjeeseen liittyvän valokuvan. Ruokaohje voidaan kategorioida pääraaka-aineen sekä ruokakulttuurin perusteella. Kategoriointiin voidaan käyttää

myös avainsanoja. Ohjelma tarjoaa neljä valmista avainsana-kategoriaa: ruokalaji (jälkiruoka), valmistustapa (grillaus), ajankohtaiset (joulu) ja erityishuomiot (vähärasvainen). Käyttäjä voi myös lisätä järjestelmään omia avainsanojaan.

Käyttäjä voi vapaasti muokata olemassa olevia ruokaohjeita, mutta tällöin vanha ohje korvautuu uudella. Minkäänlaista versionhallintaa ei siis ole toteutettu. Uuden ruokaohjeen luominen vanhaa muokkaamalla ei onnistu, vaan erilaisten muunnelmien tuottamiseksi täytyy käyttäjän kirjoittaa kaikki ruokaohjeen tiedot uudestaan.

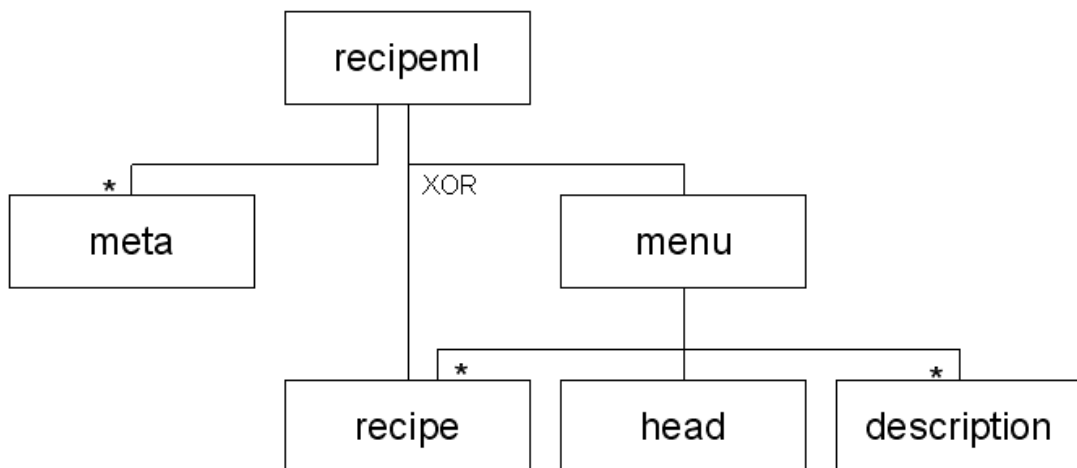
Ehkä mielenkiintoisinta BigOvenissa on sen tapa valmistusaineiden merkitsemiseen. Toisaalta käyttäjälle tarjotaan valmiita vaihtoehtoja, mikä tukee yhtenäistä merkintätapaa, mutta toisaalta käyttäjälle annetaan mahdollisuus omien merkintäratkaisujensa tekemiseen. Toisaalta käyttäjä kohtaa pitkiä, kankeita pudotusvalikoita, mutta toisaalta tälle tarjotaan myös toiminto vaihtoehtojen karsimiseen kirjoittamalla etsittävän tiedon ensimmäiset kirjaimet. Omien valmistusaineiden tallentaminen vapauttaa käyttäjän saman ruoka-aineen toistuvalla kirjoittamiselta, mutta samalla vaikeuttaa yhtenäisen merkkauksen löytämistä. Lisäksi herää kysymys, miksei yksikkö- ja olomuotokenttien arvoja voi tallentaa järjestelmään? Neljästä ruoka-aineen kentästä voidaan luoda mahdollisia yhdistelmiä ("kolme suurta maitoa raastettuna"), mutta ymmärrettävästi vastuu tästä jätetään käyttäjälle, koska järkevien yhdistelmien ymmärtäminen olisi tietokoneelle vaativa tehtävä. Ehkä suurin ongelma BigOvenissa on kuitenkin käyttäjälle ilmoittamatta tehtävät muutokset virheellisiin syötteisiin: huomaamatta jääneet virheet saattavat olennaisesti muuttaa ruokaohjetta ja johtaa epäonnistuneeseen valmistusyritykseen.

Kaiken kaikkiaan BigOven tarjoaa käyttäjälle monipuolisen työkalun ruokaohjeiden hallintaan. Käyttäjille annetaan vapaus ruokaohjeen kirjaamiseen haluamallaan tavalla, mutta samalla heille säilytetään vastuu ruokaohjeiden muotoilusta lukijaystävällisellä ja säännönmukaisella tavalla.

4.2.2 RecipeML

RecipeML on yhdysvaltalaisen FormatData-yhtiön tuottama XML-pohjainen ruokaohjemalli. XML (Extensible Markup Language) on W3C:n (World Wide Web Consortium) suositus merkintäkieleksi rakenteisen tiedon kuvailuun (ks. [W3C 2006]). RecipeML-määrittelyssä ruokaohjeiden tietosisältö on kuvattu yksinkertaisella DTD-tyyppimäärittelyllä (Document Type Definition). DTD-tyyppimäärittely kuvaa validin dokumentin rakenteen määrittelemällä listan sallituista elementeistä ja näiden attribuuteista.

RecipeML-tyyppimäärittelyn ([FormatData 2000]) uusin 0.5-versio on julkaistu vuonna 2000, ja spesifikaatiodokumentin uusin, ”working draft”-versio ([FormatData 2002]) 2002. Määrittely vaikuttaa siis kehitystyön osalta hylätyltä. Määrittelyn tuottaja, FormatData, edellyttää, että RecipeML:ää käytetään tarkasti määrittelyn mukaisesti: muutosten tekeminen on kiellettyä. RecipeML:n käytöstä täytyy myös mainita sitä käyttävässä verkkosivussa tai ohjelmassa. Näistä puutteista huolimatta esimerkiksi kaupallinen Connoisseur ja harrastelijapohjainen Largo-Recipes hyödyntävät RecipeML-dokumentteja tiedon tallentamisessa.



Kuva 4.3: Recipeml-elementti ja sen läheisimmät jälkeläiselementit.

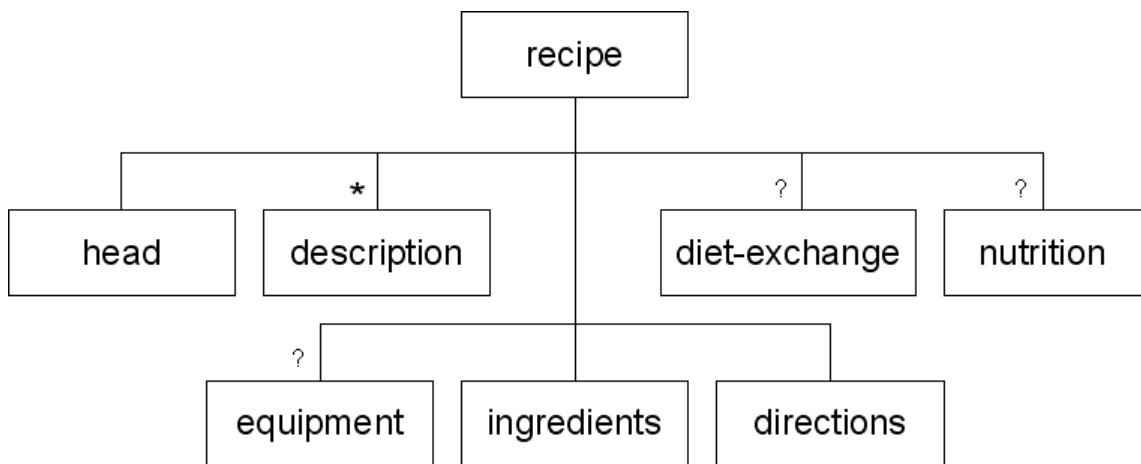
DTD-konvention mukaisesti RecipeML-dokumentti kuvaa yhden recipeml-elementin rakenteen, jota havainnollistetaan kuvassa 4.3. Tämä elementti määrittelee joko yhden ruokaohje- tai ruokalistakokonaisuuden (recipe- ja menu-elementit). Lisäksi siihen on mahdollista liittää metatietoa (meta-elementti), joilla voidaan kuvata dokumentin sisältöä. RecipeML hyödyntää Dublin Core-metatietosanastostandardia, joka pyrkii parantamaan tiedon löydettävyyttä tarjoamalla yhtenäisen tavan metatiedon määrittelyyn (ks. [Dublin Core 2007]).

Ruokalista koostuu vapaavalintaisesta määrästä ruokaohjeita. Jokseenkin mielenkiintoisesti ruokalistan ei siis tarvitse sisältää yhtään ruokaohjetta. Ruokaohjeeseen voidaan liittää rajoittamaton määrä selitetekstikappaleita (description-elementtejä), joilla kuvataan kokonaisuuden osia, eli ruokalistan tapauksessa ruokaohjeita. Näitä selitetekstikappaleita voidaan liittää myös useisiin muihin RecipeML-dokumentin elementteihin.

Ruokalistan ainoa pakollinen tieto on otsake (head-elementti). Samanlainen otsake on pakollinen osa jokaista ruokaohjetta. Otsake sisältää aina otsikon, sekä mahdollisesti alaotsikon, versiotiedon, lähteen, kategoriatietoa, valmistuksen keston ja

ruokaohjeen saannon. Vaikka DTD yleisesti ei ota kantaa tiedon esittämistapaan, ehdottaa kommentti, että lähdetieto esitetään alaviitteenä ohjeen yhteydessä. Kategorointi koostuu vapaamuotoisista avainsanoista: valmiiksi annettuja arvoja ei ole. Valmistuksen kestolle voidaan määrittää useita arvoja, jolloin ruokaohjeen eri työvaiheiden kestot voidaan eritellä.

Ruokaohjeeseen, jonka rakenne on kuvattu kuvassa 4.4, kuuluu edellä mainittujen selitetekstien ja otsaketietojen lisäksi omat elementtinsä käytettävien välineiden, valmistusaineiden, valmistusohjeiden ja ravintotietojen määrittelyyn. Lisäksi ruokaohjeeseen voidaan liittää korvaustietoja (diet-exchange-elementti), joiden käyttötarkoitus ei yksiselitteisesti selviä tyyppimäärittelystä. Mahdollisesti tarkoitus on ilmoittaa, millaisella ruoalla esimerkiksi diabeetikko voi korvata kyseisen ruokalajin ja kuitenkin saada saman määrän eri ravintoaineita.

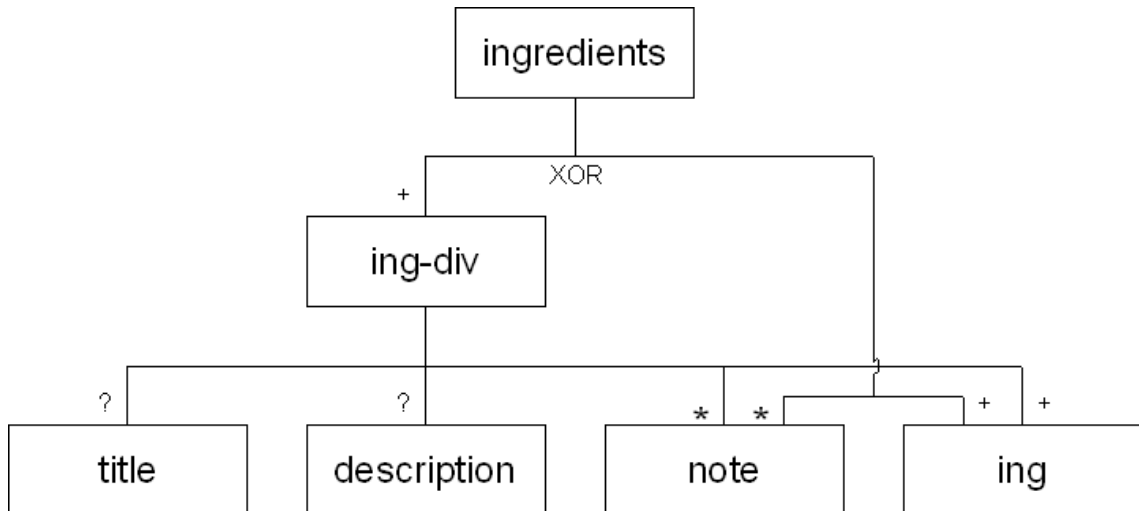


Kuva 4.4: Recipe-elementti ja sen läheisimmät jälkeläiselementit.

Valmistusvälineiden, -aineiden ja -ohjeiden määrittelyissä käytetään samantyyliisiä rakenteita. Tiedot voidaan esittää joko joukkona yksittäisiä elementtejä ja niihin liittyviä vapaamuotoisia huomautustekstejä, tai vaihtoehtoisesti jaettuna erillisiin osioihin (engl. division). Esimerkiksi valmistusväline-elementti voi sisältää vapaavalintaisen määrän tool- ja note-elementtejä, tai yhden tai useamman equip-div-elementin määrittelemän osion. Nämä osiot sisältävät mahdollisen otsikon, edellä esitetyn description-elementin, sekä vaihtelevan määrän tool- ja note-elementtejä.

Työvälineet määritellään siis tool-elementillä, jossa työväline kuvataan vapaamuotoisella tekstillä, sekä mahdollisilla tiedoilla välineen valmistajasta ja tarvittavien välineiden määrästä. Lisäksi voidaan määrittää, onko kyseinen työväline valmistusprosessin kannalta välttämätön vai vapaavalintainen.

Valmistusaineita määriteltäessä käytetään tool-elementtiä vastaavaa ing-elementtejä. Kuten työvälineet, myös valmistusaineet voidaan luetella vapaavalintai-

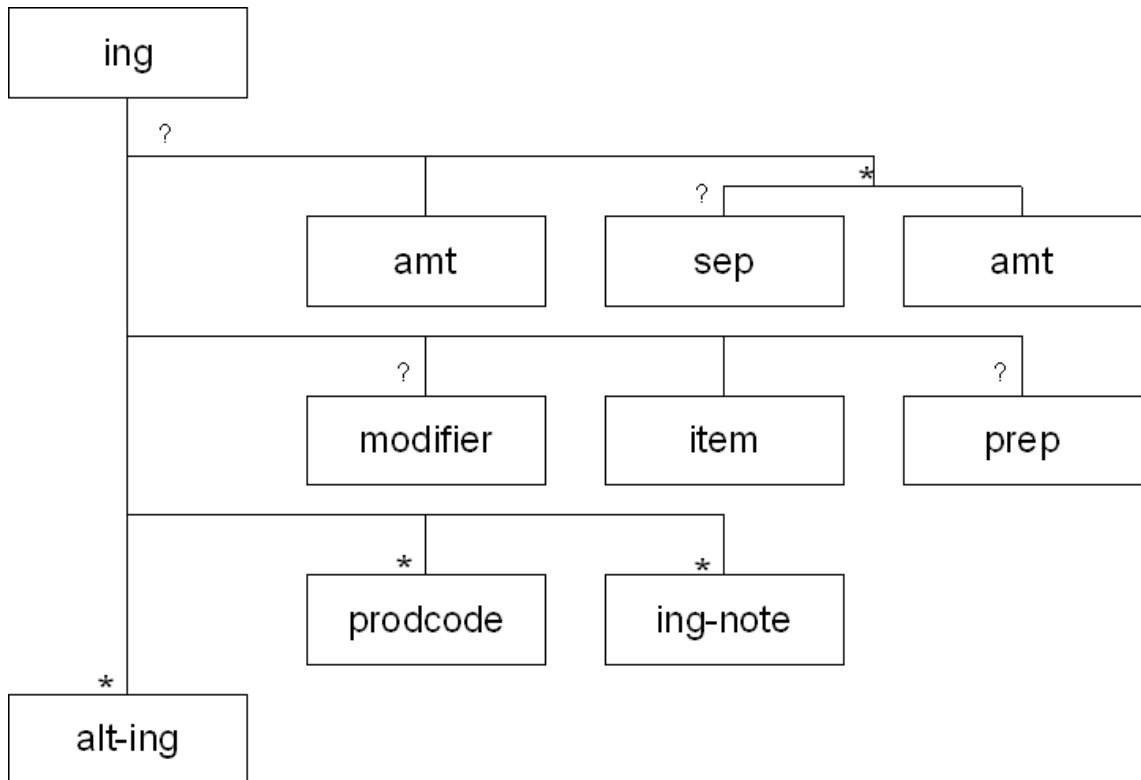


Kuva 4.5: Ingredients-elementti ja sen läheisimmät jälkeläiselementit.

nessä järjestyksessä, tai ne voidaan ryhmitellä ruokaohjeen työvaiheiden mukaisiin osioihin. Tätä rakennetta havainnollistetaan kuvassa 4.5. Vaihtoehtoisesti valmistusaineet voidaan luetella valmistusvaiheisiin sidottuina directions-elementin sisällä. Yksittäiselle valmistusaineelle, joka on kuvattu kuvassa 4.6, määritellään tarvittava määrä, ruoka-aine ja sen mahdollinen tuotekoodi, mahdollinen korvaava valmistusaine sekä tieto siitä, voidaanko valmistusaine tarvittaessa jättää pois ruokaohjeesta. Ing-elementille voidaan antaa korkeintaan yhdet modifier- ja prep-elementit, mutta näiden elementtien käyttötarkoitusta ei kuvata itse tyyppimäärittelyssä tai siihen liittyvässä spesifikaatiossa.

Valmistusaineen määrän määrittelyyn käytetään tietorakennetta, joka mahdollistaa sekä yksinkertaisten että monipuolisempien merkintöjen tallentamisen. Määrä voidaan ilmoittaa yhdellä arvolla tai kahden arvon vaihteluvälillä (esimerkiksi 3 tai 3-4). Tähän voidaan liittää koko (pieni, suuri), yksikkö (purkkia, desilitraa) sekä yksikön koko (iso paketti). Mallin avulla voidaan määrätieto määrittellä hyvin vapaasti, kuten ”yksi kappale” tai ” $1\frac{1}{2}$ -2 kappaletta 2 desilitran purkkeja”. Tämän monipuolisuuden hintana on mahdollisuus luoda järjettömiä yhdistelmiä käytettävissä olevista paloista.

Valmistusvaiheet voidaan määrittellä tähän tarkoitettuun step-elementin tai niitä sisältävien osioiden avulla. Valmistusvaihe sisältää tekstimuotoiset kuvaukset vaiheen tehtävistä, kestosta, siihen liittyvistä ehdoista (condition-elementti) ja asetuksista. Tyyppimäärittely tai spesifikaatio ei määrittele condition-elementin tarkoitusta, mutta kyseessä voisi olla esimerkiksi ehtoon (”paista kunnes kypsä”) liittyvä tieto. Asetuksia (setting-element) voidaan käyttää esimerkiksi valmistusvaiheessa käytettävän uunin lämpötilan ilmoittamiseen. Valmistusvaiheessa voidaan kertoa



Kuva 4.6: Ing-elementti ja sen läheisimmät jälkeläiselementit.

sihen liittyvät työvälineet ja valmistusaineet toolref- ja ingref-elementtien IDREF-attribuuttien avulla.

Nutrient- ja diet-exchanges-elementtien käyttötarkoituksen voi päätellä niiden nimistä, mutta myös näiden elementtien osalta on tyyppimäärityksen luoja tarjoama ohjeistus olematon.

RecipeML mahdollistaa varsin monipuolisten ja yksityiskohtaisten ruokaohjeiden tallentamisen. Käytännössä tämä monipuolisuus voi johtaa työlääseen tiedon-syöttöprosessiin ja järjettömiin yhdistelmiin. Tyyppimääritys ei kuitenkaan ota kantaa siihen, millä tavalla talletettava tieto syötetään. Sopivilla syöttömenetelmillä voidaan helpottaa tiedon syöttämistä ja samalla rajata tapauskohtaisesti millaisia ohjeita järjestelmään voidaan tallentaa. Esimerkiksi ruokaohjeen kategoriatiedot tallennetaan RecipeML-dokumenttiin vapaamuotoisena tekstinä. Jos ruokaohjeiden kirjoittajat voivat syöttää kategoriatiedon vapaamuotoisena tekstinä, on vaarana, että nämä käyttävät eri merkintätapoja saman asian ilmaisuun, jolloin kategoriatietoa ei saada hyödynnettyä tehokkaimmalla mahdollisella tavalla. Ongelma voidaan kiertää rajaamalla tarjolla olevat kategoriavaihtoehdot tietoa syötettäessä.

4.2.3 Dish_Master

Wang ja Li (2007) esittävät kolme ruokaohjetiedolle tyypillistä ominaisuutta, jotka heidän mukaansa tulee huomioida ruokaohjetta mallinnettaessa. Ensinnäkin ruokaohjetieto on tyypillisesti vain löyhästi rakenteista (engl. loosely structured). Sama ohje voidaan kuvata usealla eri tavalla ja tarkkuustasolla, ilman että kuvattava ruokalaji muuttuu. Toiseksi, koska merkittävä osa ruokaohjeen sisällöstä kuvaa miten tietty ruokalaji valmistetaan, ei ruokaohjemalli ole pelkästään tietopainotteinen (engl. data-intensive), vaan myös toimintapainotteinen (engl. behaviour-oriented). Kolmanneksi ruokaohjeiden valmistusvaiheilla on usein erilaisia rajoitteita, jotka liittyvät joko yksittäiseen valmistusvaiheeseen tai useamman valmistusvaiheen ketjuun. [Wang & Li 2007]

Wang ja Li toteavat myös, että nykyiset verkossa sijaitsevat ruokaohjetietokannat eivät käytännössä tue käyttäjiä ruokaohjeiden analysoinnissa, vertailussa ja käyttäjälle sopivien muunnelmien luomisessa. Nämä ongelmat huomioiden suunniteltiin Dish_Master-järjestelmä. Dish_Masterin tietomalli huomioi myös löyhästi rakenteisen tiedon, toimintapainotteisen tiedon sekä ruokaohjeisiin liittyvien rajoitteiden aiheuttamat erityistarpeet. [Wang & Li 2007]

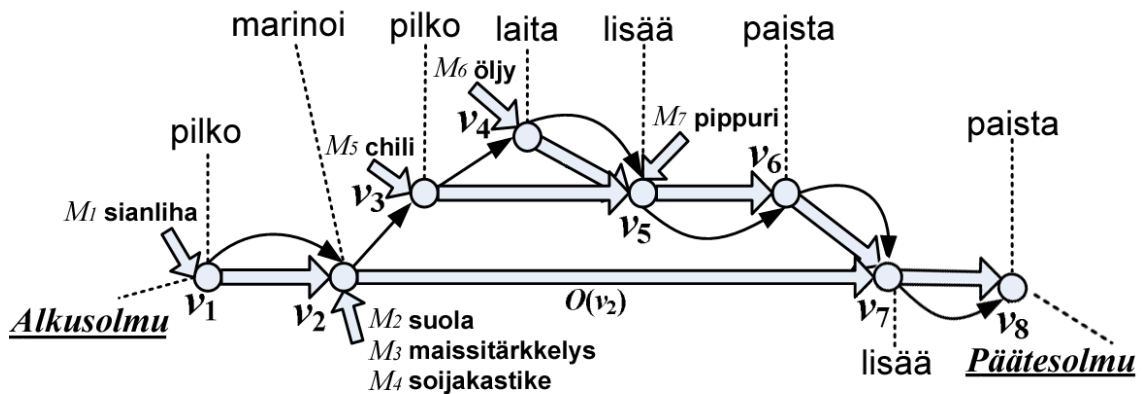
Dish_Masterin tietomallissa ruokaohje koostuu ruokaohjetta kuvaavista ominaisuuksista, valmistusaineista ja valmistusgraafista (engl. cooking graph). Ruokaohjeeseen liittyviä ominaisuustietoja ovat esimerkiksi valmistustekniikka, ravintoarvot, ruokakulttuuri josta ohje on lähtöisin sekä kuva valmiista ruoka-annoksesta. [Wang & Li 2007] Ruokaohjeet voidaan kategorioida maun, ulkonäön, ravintoarvojen, ruokakulttuurin, valmistustavan, valmistusaineiden ja juhlateeman mukaan [Wang et al. 2006].

Jokainen ruokaohjeen valmistusaine voi olla joko yksittäinen ruoka-aine tai useamman ruoka-aineen kooste. Valmistusgraafissa valmistusaineena voi toimia myös aiemman valmistusvaiheen vaihetuote. Valmistusaineeseen liittyy yksilöllinen tunniste sekä joukko ominaisuuksia, kuten nimi, määrä ja kuva [Wang & Li 2007]. Lisäksi valmistusaineeseen liittyy tieto sen mausta, ulkonäöstä ja ravintoarvoista [Wang et al. 2006]. Valmistusaineet luokitellaan roolinsa mukaan pääraaka-aineiksi, sivuraaka-aineiksi ja mausteiksi ([Wang et al. 2006], [Wang & Li 2007]).

Valmistusaineisiin liittyvät läheisesti annoksen (engl. selection) ja vaikutuksen (engl. effect) käsitteet. Annos liittyy valmistusaineeseen tiedon tarvittavasta määrästä, määrän yksiköstä sekä valmistusaineen rakenteesta, kuten ”pilkottu” tai ”raastettu”. Vaikutus kertoo millä tavalla kuvattu valmistusaine tulee valmistaa, ja miten tämä vaikuttaa ruokalajin makuun, ulkonäköön ja ravintoarvoihin. [Wang et al. 2006]

Dish_Masterissa ruokaohjeen valmistusohje muodostaa kolmitasoisien hierarkian. Korkeimmalla, ruokaohjetasolla (engl. Recipe Level) kerrotaan ruokaohjeen nimi ja erilaisia luokitteluun käytettäviä tietoja. Yhdistelmätasolla (engl. Composite Level) kuvataan ruokaohje valmistusvaiheiden tarkkuudella, ja perustasolla (engl. Primitive Level) valmistustoimintojen tarkkuudella. Jokainen ruokaohje koostuu siis vaihtelevasta määrästä valmistusvaiheita (engl. cooking step), jotka voidaan pilkkoa toimenpiteisiin (engl. cooking action). Jokainen näistä toimenpiteistä kohdistuu yhteen tai useampaan valmistusaineeseen. [Wang et al. 2006]

Perustuen siihen, kuinka suuri vaikutus yksittäisellä valmistusvaiheella tai valmistustoiminnolla on koko ruokalajin makuun, ulkonäköön ja ravintoarvoihin, määritellään valmistusvaiheille ja -toiminnoille painoarvot välillä [0..1]. Tutkittaessa esimerkiksi ruokaohjeen makua, voidaan painoarvojen avulla päätellä, mitkä maust ovat niin heikkoja muihin makuihin verrattuna, että ne voidaan tulkita peittyviksi ja jättää huomioimatta lopputuloksen makua arvioitaessa. Suuren arvon saavia makuja voidaan puolestaan käyttää apuna ruokalajien luokittelussa. Edellä mainitut heikkojen ja vahvojen arvojen ominaisuudet pätevät vastaavalla tavalla myös ruokalajien ulkonäköä ja ravintoarvoja arvioitaessa. [Wang et al. 2006]



Kuva 4.7: Esimerkki valmistusgraafista (mukaillen [Wang & Li 2007])

Valmistusgraafi on nimetty suunnattu graafi, joka kuvaa koko ruoanvalmistusprosessin. Esimerkinomainen yksinkertainen valmistusgraafi on esitetty kuvassa 4.7. Valmistusgraafi koostuu solmuista, joissa kerrotaan mikä toimenpide kokon tulee suorittaa. Lisäksi solmuihin liittyy joukko rajoitteita, siirtymiä (engl. edge) ja valmistusaineita. Rajoitteita on kahdenlaisia. Toimintorajoitteiden (engl. cooking action constraints) tulisi olla voimassa tiettyä toimenpidettä suoritettaessa. Tällainen rajoite on esimerkiksi uunin riittävä lämpötila. Siirtymiin liittyy omat rajoitteensa (engl. cooking transition constraints), jotka määrittelevät toimenpiteiden keskinäisen suoritusjärjestyksen. Näin voidaan siis määritellä, että tietty toimenpide tulee

olla suoritettu loppuun, ennen kuin toinen toimenpide voidaan aloittaa. Esimerkki tällaisesta rajoitteesta on tilanne, jossa edellisen toimenpiteen vaihetuotteen tulee olla kypsä, jotta seuraavaan toimenpiteeseen voidaan siirtyä. [Wang & Li 2007]

Kuvassa 4.7 harmaat ympyrät esittävät valmistustoimenpiteitä $v_1 - v_8$. Mustat ohuet nuolet kuvaavat toimenpiteiden suorittamisjärjestystä. Paksummat harmaat nuolet puolestaan kuvaavat valmistusaineiden siirtymiä solmujen välillä. Valmistusaineita ovat ruoka-aineet $M_1 - M_7$ sekä solmun v_2 vaihetuote, $O(v_2)$. Valmistusaineena käytetään oletusarvoisesti edellisen toimenpiteen vaihetuotetta, mikäli toisin ei mainita. Valmistusgraafi siis ohjeistaa ensin pilkkomaan sianlihan ja marinoimaan sen suolan, maissitärkkelyksen ja soijakastikkeen kanssa. Tämän jälkeen tulee pilkkoa chili, laittaa se öljyyn, lisätä pippuria ja paistaa sekoitus. Lopuksi tähän lisätään marinoitu porsaanliha ja paistetaan ruoka valmiiksi.

Wang & Li huomauttavat, että graafit saattavat sisältää hahmoja (engl. patterns), eli pienempiä graafeja, jotka saattavat esiintyä samanlaisina useissa ruokaohjeissa [Wang & Li 2007]. Esimerkiksi keittäminen on hahmo, joka koostuu veden lisäämisestä kattilaan, sen lämmittämisestä kiehuvaaksi ja valmistusaineiden kypsennyksestä kiehuvaan vedessä. Tällaisten hahmojen tunnistaminen on hyödyllistä, sillä hahmoja voidaan hyödyntää ruokaohjeiden mukauttamisessa, mikäli niiden reaktiot eri valmistusaineiden kanssa tunnetaan.

Dish_Masterin tapauksessa ruokaohjeen mukauttaminen perustuu vahvasti juuri valmistusgraafin hyödyntämiseen. Ideana on, että käyttäjä ilmoittaa missä graafin solmussa (eli valmistusvaiheessa) hän haluaa suorittaa tietyn muutoksen, ja järjestelmä päättää tämän perusteella, voidaanko ruokaohjeen loppuosa valmistaa siten, että lopputulos on järkevä. Käytännössä järjestelmä siis ennustaa, miten tietyn valmistusaineen tai valmistusvaiheen muuttaminen vaikuttaa lopputulokseen [Wang et al. 2006]. Mikäli sopivia mukautuksia on olemassa, järjestelmä esittää käyttäjälle sopivat vaihtoehdot, ja antaa tämän valita niistä mieleisensä. Käyttäjä siis ehdottaa alkuperäistä muokkausta joko mieltymystensä tai käytännön tarpeiden, kuten puuttuvan valmistusaineen, takia. Järjestelmä ei siis automaattisesti suosittele käyttäjälle ruokaohjeen mukauttamista. [Wang & Li 2007] Lähestymistavalla on hyvät ja huonot puolensa. Järjestelmän ei tarvitse suorittaa ylimääräisiä arvauksia käyttäjän tarpeista, koska käyttäjä kertoo tarpeistaan niiden ilmaantuessa. Toisaalta käyttäjä voi alkaa turhaan käydä läpi ruokaohjetta, jota ei voida toteuttaa tälle sopivalla tavalla. Mikäli tällaiset tapaukset tuhlavat runsaasti käyttäjän aikaa, saattaa tämä kyllästyä sopivien ohjeiden etsimiseen, jolloin potentiaalisesti sopivat muunnokset saattavat jäädä löytämättä.

Dish_Masteria käsittelevissä artikkeleissa ([Wang et al. 2006], [Wang & Li 2007]) keskityttiin lähinnä erilaisten toimintojen edellyttämän tietomallin esittämiseen yleisellä tasolla. Yksityiskohtaisia mallinnusratkaisuja ei esitetty eikä toteutustekniikkaan otettu kantaa. Dish_Master sisälsi muutamia mielenkiintoisia ominaisuuksia, joita BigOvenissa ja RecipeML:ssä ei oltu huomioitu. Vaikka apua ruokaohjeen mallintamiseen ei tarjottu, herättävät valmistusainekoosteet ja valmistusgraafi mallinustyön kannalta mielenkiintoisia ajatuksia.

Mukauttamisen tarpeita silmälläpitäen valmistusaineet on Dish_Masterissa määriteltä melko tarkasti. Esimerkiksi valmistusaineen koostuminen useammasta ruoka-aineesta on tärkeää, kun järjestelmä arvioi valmistusaineen vaikutusta ruokaohjeen kokonaisuuteen. Myös tämä mahdollisuus arvioida ja painottaa yksittäisen valmistusvaiheen tai ruoka-aineen vaikutusta on ainutkertainen. Tällainen ominaisuus olisi varmasti hyödyllinen henkilölle, joka luo erilaisia muunnelmia ruokaohjeista. Toiminnon toteuttamisen vaatima työmäärä ei kuitenkaan liene perusteltua, mikäli toimintoa hyödyntää kaikkien käyttäjien sijasta vain pieni sisällöntuottajien käyttäjärühmä.

Myös valmistusgraafi on ajatuksena mielenkiintoinen, mutta siitä saatavat hyödyt jäänevät pieniksi verkkopalvelussa, jossa ruokaohjeiden dynaamista mukautusta ei toteuteta. Mahdollisesti valmistusgraafista lähtöisin oleva huomio siitä, että valmistusaineena voi toimia tavallisen valmistusaineen lisäksi edellisen vaiheen vaihe-tuote on ruokaohjeen mallintamisen kannalta hyödyllinen, mutta käytännössä yksinkertaisempi ratkaisu tähän on perinteisesti ollut asian kertominen vapaamuotoisesti valmistusvaiheiden kuvauksen yhteydessä.

4.3 Mallintamisen vaikutus sisällöntuotantoon ja mukautusprosessiin

Tietomallin yksityiskohtaisuus heijastuu sisällönhallintaan ylimääräisenä työnä. Sisällön syöttäjän täytyy nähdä ylimääräistä vaivaa, jotta syötetty sisältö saadaan tietomallin edellyttämässä rakenteisessa muodossa. Esimerkiksi ruokaohjeen valmistusvaiheiden kirjoittaminen yhteen tekstikenttään olisi nopeaa, mutta tällöin työvaiheet toisistaan erottavan ohjelmiston luominen olisi vaikeaa. Sisällöntuottajan täytyy eritellä valmistusvaiheet joko erillisen syntaksin avulla tai käyttämällä useampaa syöttökenttää.

Rakenteisen tiedon työlästä syöttämistä voidaan pyrkiä helpottamaan erilaisilla, tapauskohtaisesti sovellettavilla mallintamisen ratkaisuilla. Esimerkiksi valmiiden

arvojen tarjoaminen tilanteissa, jossa mahdolliset vaihtoehdot tiedetään ennalta, nopeuttaa tiedon syöttöä, vähentää mahdollisia käyttäjän tekemiä virheitä sekä yhtenäistää sisällöntuottajien käyttämää merkintätapaa. Tällöin on kuitenkin tärkeää varmistua, että käyttäjän vaihtoehtoja ei tarpeettomasti rajoiteta.

Luonnollisesti tulisi välttää keräämästä tietoa, jota ei hyödynnetä mitenkään. Vastaavasti tietomallin suunnittelijan tulee miettiä ovatko kaikki vaadittavat tiedot aina sisällöntuottajan saatavilla. Jos käyttäjää vaaditaan antamaan tietoja joita hän ei osaa tai halua kertoa, saattaa hän syöttää virheellistä tietoa tai täyttää roskaa. Pahimmassa tapauksessa käyttäjä keskeyttää tiedon syöttämisen kokonaan. Tiettyjen tietojen pakollisuus on kuitenkin perusteltua esimerkiksi tilanteissa, joissa tietoa käytetään mukautukseen tai muuhun järjestelmän kannalta kriittiseen toimintoon.

On tärkeää huomioida, että tietomallin suunnittelijan käsitelmä mallinnettavasta asiasta saattaa erota merkittävästi sisällöntuottajan käsitelmästä. Lisäksi käytettävän teknologian tapa mallintaa asioita heijastuu helposti lopulliseen tietomalliin. Käytettävän tietomallin ei kuitenkaan tarvitse näkyä sisällöntuottajalle. Sisällönhallintaan käytettävästä käyttöliittymästä voidaan luoda sisällöntuottajan käsitelmään sopiva, jolloin hänen ei tarvitse välittää tiedon varastointiin käytettävästä tietomallista. Käyttöliittymässä tiedot tulisi kerätä juuri sisällöntuottajan käsitelmän mukaisissa loogisissa kokonaisuuksissa ja järjestyksessä.

Sisällön mukautuminen ja muu koneen suorittama päättely edellyttää, että itse tietosisällön lisäksi kuvaillaan myös kyseisen sisällön merkitys. Esimerkiksi valmistusainetta käsittelevän koneen tulee voida erotella mikä osa tekstistä kuvaa valmistusaineen määrää, ainetta, rakennetta ja niin edelleen. Rakenteisen tiedon lisäksi tietosisällön koneellinen käsittely saattaa edellyttää metatiedon hyödyntämistä.

Toimivassa järjestelmässä sekä käytettävä tietomalli että tiedon syöttämiseen käytettävä käyttöliittymä ovat kunnossa. Molemmat osat ovat kokonaisuuden kannalta yhtä tärkeitä, sillä hienoinakin tietomalli jää hyödyntämättä mikäli tietosisällön syöttö ei onnistu ja päinvastoin. Yleisesti katsottuna tietomallin suunnittelijan tulee etsiä sopiva kompromissi monipuolisen, sisällöntuottajalle runsaasti vapauksia tarjoavan ja yksikäsitteisen, yhtenäistä merkintätapaa tukevan, konepäättelyyn soveltuvan tietomallin väliltä. Huomioimalla oikeiden loppukäyttäjien käsitelmät ja toimintatavat voidaan luoda tehokkaan sisällöntuotannon mahdollistava, käytettävä käyttöliittymä.

Luku 5

Aatu — Apua erikoisruokavalion hallintaan

Tämä luku esittelee hankkeen aikana toteutetun verkkopalvelun niin tekniseltä kuin toiminnalliseltakin kannalta. Lisäksi kerrotaan, miten mukautuminen, tiedon mallintaminen ja sisällönhallinta huomioitiin verkkopalvelun toteutuksessa.

5.1 Aatu-verkkopalvelun esittely

Terveydellisistä syistä erityisruokavalioita noudattaville epäsopivasta ravinnosta saattaa aiheutua vakavia seurauksia. Tästä johtuen verkkosivuilta saatavan informaation kriittinen medialukutaito on erityisen tärkeää näille henkilöille. Aaturuokaohjepalvelun (jäljempänä ”palvelu”) tavoitteena on olla julkinen, avoin ja maksuton verkkosivusto, joka tarjoaa ajantasaista ja luotettavaa täsmätietoa erityisruokavalioihin sopivista elintarviketuotteista ja ruokaohjeista. Haasteena on luotettavan informaation tarjoamisen lisäksi pitää mukauttamisprosessit niin läpinäkyvinä, että käyttäjien ei tarvitse luottaa niihin sokeasti, vaan he voivat halutessaan itse varmistaa suoritettujen mukautusten oikeellisuuden.

Palvelu tarjoaa ylläpitäjilleen web-pohjaisen käyttöliittymän palvelun keskeisen tietosisällön hallintaan. Erityisruokavalioihin soveltuvia elintarvikkeita valmistavat tuottajat voivat web-käyttöliittymän kautta lisätä ja hallita tuotteitaan koskevaa, palvelussa julkaistavaa tietosisältöä. Loppukäyttäjät puolestaan voivat luoda itselleen ruokavalioprofileja, joiden perusteella sivuston muu sisältö mukautetaan käyttäjän ruokavalioon sopivaksi. Loppukäyttäjät voivat myös lisätä palvelusta löytyviin ruokaohjeisiin omia suosikki- ja avainsanamerkintöjä.

Aatu-hanke on Tampereen teknillisen yliopiston Hypermedialaboratorion yhdessä Pirkanmaan Allergia- ja Astmayhdistys ry:n ja Suomen Keliakialiiton kanssa toteuttama hanke. Vuoden 2006 alussa alkaneen kolmivuotisen hankkeen rahoittaa Raha-automaattiyhdistys.

5.2 Käyttäjryhmät ja palvelun käytön tavoitteet

Tässä esiteltyt käyttäjryhmät perustuvat hankkeen alkuvaiheessa luotuun toiminnalliseen määrittelydokumenttiin. Dokumentti on tarkoitettu hankeryhmän sisäiseen käyttöön eikä ole julkisesti saatavilla. Varsinaisen toteutustyön aikana keskityttiin erityisesti kuluttajien, elintarviketuottajien ja ylläpitäjien tarpeiden täyttämiseen, koska nämä katsottiin käyttäjryhmistä keskeisimmiksi. Yleisellä tasolla palvelun käyttäjät voidaan jakaa ylläpitäjiin, sisällöntuottajiin ja loppukäyttäjiin. Loppukäyttäjryhmän sisällä käyttäjät voidaan luokitella edelleen erilaisiin käyttäjrytöyrypeihin.

5.2.1 Ylläpitäjät

Ylläpitäjäksi luokitellaan verkkopalvelun tarjoajaa edustavat käyttäjät. Ylläpitäjien tavoitteena on tarjota laadukas verkkopalvelu loppukäyttäjille ja he vastaavat lopulta palvelussa julkaistun informaation ajantasaisuudesta ja paikkansapitävydestä.

Ylläpitäjät hallitsevat palveluun liittyviä käyttöoikeuksia ja tietosisältöä. Käyttöoikeuksien hallintaan kuuluu esimerkiksi elintarviketuottajiksi hakevien yritysten hyväksyminen ja tietosisällön tuottamiseen liittyvien käyttöoikeuksien jakaminen. Tietosisällön ylläpito koostuu julkaistavaksi ehdotettujen ruokaohjeiden, elintarviketuottajien ja heidän tuotteidensa julkaisutilojen hallinnasta sekä palvelun tuntemien ruoka-aineiden listan päivittämisestä. Esimerkiksi sisällöntuottajan palveluun tallentama ruokaohje ei näy loppukäyttäjille, ennen kuin ylläpitäjä on hyväksynyt ruokaohjeen julkaisun. Julkaisutiloja kuvataan tarkemmin alikohdassa 5.6.4.

Ylläpitäjien ryhmä on kooltaan pieni ja käyttö on säännöllistä. Lisäksi ylläpito on osa käyttäjän työtehtäviä, joten lyhyen käyttöopetuksen järjestäminen ei ole ylivoimaista. Ryhmän jäsenet myös kommunikoivat keskenään, jolloin yhteisen toimintatöyryjen sopiminen on mahdollista.

5.2.2 Sisällöntuottajat

Sisällöntuottajia ovat ruokaohjeiden tuottajat ja elintarviketuottajat, joiden tuotteita palvelussa esitellään. Merkittävin ero elintarviketuottajien ja muiden sisällöntuottajien välillä on rekisteröitymiskäytännössä. Palveluun rekisteröityessään elintarviketuottajat ilmoittavat olevansa elintarviketuottajia, ja ylläpitäjän hyväksynnän jälkeen he voivat alkaa lisätä tuotteitaan palveluun. Valituille elintarviketuottajille voidaan antaa oikeudet julkaista tuotetietoja suoraan ilman ylläpitäjän suorittamaa tarkistusta. Edellytyksenä tälle kuitenkin on, että tuotteen valmistusaineluottelo on annettu täydellisenä sellaisessa muodossa, että sen koneellinen käsittely on mahdollista. Epäselvyyksiä sisältävät tuotteet tulee aina hyväksyttää ylläpitäjällä ennen julkaisua.

Elintarviketuottajien ryhmään kuuluvat yritykset, jotka tuottavat tai maahan tuovat erityisruokavalioihin soveltuvia elintarviketuotteita. Tälle käyttäjäryhmälle verkkopalvelu toimii markkinointiväylänä kohdennetulle asiakaskunnalle, maksuttomana tuoteluettelonä internetissä sekä mahdollisena informaatiolähteenä tuotekehitystä varten.

Koska verkkopalvelun käyttäjäkunnan voidaan olettaa olevan kiinnostunut erityisruokavaliotuotteista, on se oiva markkinointiväylä elintarviketuottajille. Kuluttajat saavat informaatiota uusista ja muuttuneista tuotteista vahtipalvelun kautta ja selaamalla palvelua. Myöhemmin palvelua voidaan kehittää tukemaan entistä tehokkaampaa kommunikointia, kuten esimerkiksi lähettämällä käyttäjille — heidän suostumuksellaan — ruokavalion perusteella personoituja viikottaisia uutiskirjeitä.

Elintarviketuottajat voivat käyttää verkkopalvelua myös informaation lähteenä. Informaatiota voidaan kerätä kuluttajilta ja ammattikeittiöiltä aktiivisesti esimerkiksi kyselytutkimusten avulla. Huomaamattomammin informaatiota voidaan kerätä tarkkailemalla loppukäyttäjien toimintaa tilastojen kautta. Esimerkiksi tilastoja palveluun luoduista ruokavalioprofiileista voidaan hyödyntää uusien tuotteiden ideoinnissa. Muuta hyödynnettävää informaatiota ovat esimerkiksi tuotteisiin tutustuneiden käyttäjien määrä sekä merkinnät, joita käyttäjät ovat tuotteisiin liittäneet.

Käyttäjärühmän kannalta tärkein toiminto palvelussa on oman yrityksen sekä yrityksen tuotteiden tietojen ylläpito. Tuotetietojen ajantasaisuuden varmistamiseksi yrityksen täytyy kolmesti vuodessa vahvistaa tietojen ajantasaisuus. Mikäli vahvistusta ei anneta, voidaan tuotteet siirtää piilotettuun tilaan, eikä niitä esitetä käyttäjille ennen kuin vahvistus tietojen ajantasaisuudesta on saatu (ks. alikohta 5.6.4).

Elintarvikeyrityksen työntekijöillä saattaa olla erittäin tarkka tietämys oman yrityksen elintarvikkeista, mutta informaation ilmaiseminen yhtenäisesti eri yritysten välillä on mahdollinen haaste. Lisäksi tulee huomioida, että tämän käyttäjäryhmän jäsenten tietotekniset taidot voivat vaihdella suuresti, eikä kirjallista ohjeistusta parempaa koulutusta ole välttämättä kaikissa tilanteissa tarjolla.

Ruokaohjeiden tuottajat valitaan tehtävänsä verkkopalvelun ulkopuolisten kontaktien perusteella, eivätkä käyttäjät siis voi rekisteröinnin yhteydessä hakea ruokaohjeiden tuottajaksi. Ainakin palvelun julkaisun yhteydessä palvelun ylläpitotehtävistä ja ruokaohjeiden tuottamisesta vastaa yksi ja sama henkilö, mutta tulevaisuudessa tehtävät mahdollisesti hajautetaan eri käyttäjille. Tästä johtuen myös ruokaohjeiden julkaisu toteutetaan elintarviketuotteiden yhteydessä esitellyn kaltaisen tarkistuskäytännön kautta. Ruokaohjeiden tuottajien tavoite on sama kuin ylläpitäjillä: laadukkaan verkkopalvelun tuottaminen loppukäyttäjille.

5.2.3 Loppukäyttäjät

Loppukäyttäjät jaetaan ryhmiin toiminnallisten rooliensa perusteella. Tällaisia käyttäjäryhmiä ovat ruokavalioita noudattavat ja heidän läheisensä, ammattikeittiöt, ruoka-alan ammattilaiset ja kaupan alan toimijat. Mikäli käyttäjä ei ole rekisteröitynyt tai kirjautunut palveluun, luokitellaan hänet vierailijaksi.

Kuluttajat

Kuluttajat edustavat itseään, sekä mahdollisesti perhettään tai vastaavaa ryhmää. Erityisruokavaliota noudattavat kuluttajat etsivät verkkopalvelusta päivittäisten valintojensa tueksi ajantasaista tuotetietoa elintarvikkeista. Tällaisia tuotetietoja ovat esimerkiksi elintarvikkeiden ainesosaluettelot ja ravintoarvot. Lisäksi käyttäjät etsivät uusia ruokaohjeita ja informaatiota ruoka-aineallergioista. Erityisen tärkeää kuluttajille on informaatio elintarvikkeiden ja ruokaohjeiden soveltuvuudesta heidän ruokavalioonsa.

Kuluttajien palveluun tuottama sisältö on vähäistä. Tärkeimmäksi nousevat käyttäjien omat ja perheenjäsenten ruokavaliot. Näitä ruokavaliota käytetään tarkistettaessa soveltuuko tietty tuote tai ruokaohje käyttäjälle. Ruokavalion lisäksi käyttäjä kertoo rekisteröitymisen yhteydessä palvelulle normaalit henkilötiedot, kuten asuinpaikkakunta, jota käytetään esimerkiksi tilastojen laatimiseen.

Elintarvikkeiden ja ruokaohjeiden hakemisen sekä ruokavaliotietojen ylläpitämisen jälkeen kuluttajalle tärkeimmät toiminnot ovat yhteisölliset toiminnot, kulutta-

javiestintä sekä vahtipalvelu. Verkkopalvelussa yhteisöllisiä toimintoja ovat esimerkiksi oman ruokavalion vapaaehtoinen tuominen muiden nähtäväksi sekä ruokaohjeiden ja elintarviketuotteiden kommentointi. Kuluttaja voi kommentoida ruokaohjeita ja elintarviketuotteita merkitsemällä niitä suosikeikseen tai lisäämällä niihin lyhyitä vapaamuotoisia avainsanamerkintöjä (engl. tags). Nämä avainsana- ja suosikkimerkinnät ovat vapaasti yhteisön nähtävillä, jolloin kuluttajat voivat päätellä mitä mieltä muut käyttäjät ovat olleet tietystä ruokaohjeesta tai tuotteesta.

Erityisruokavaliotuotteisiin erikoistunut verkkopalvelu tarjoaa oivallisen väylän palautteen antamiseen tuotteiden valmistajille. Suoran palautteen lisäksi tuottajat näkevät mitä merkintöjä käyttäjät ovat tuotteisiin lisänneet. Tulevaisuudessa verkkopalvelun kautta voidaan mahdollisesti suorittaa erilaisia kyselyjä ja tutkimuksia.

Vahtipalvelu seuraa palveluun lisättäviä ruokaohjeita ja tuotteita. Kun palveluun lisätään uusi käyttäjälle soveltuva ruokaohje tai elintarviketuote, tai jo palvelussa oleva tuote muuttuu, vahtipalvelu ilmoittaa tästä kuluttajalle tämän kirjautuessa palveluun seuraavan kerran. Näin kuluttaja pysyy perillä palvelussa tapahtuvista muutoksista ilman, että tämän täytyy erikseen tarkkailla tuotteita omatoimisesti.

Ammattikeittiöhenkilökunta

Ammattikeittiö-käyttäjryhmän jäsen voi edustaa suurtalouskeittiötä, kuten vaikkapa koulun ruokalaa tai hotelli- ja ravintola-alan yritystä. Koska tämän käyttäjryhmän jäsenet valmistavat ruokaa suurelle joukolle erilaisia erityisruokavalioita noudattavia henkilöitä, kaipaavat he tuotevalinta- ja erityisruokavaliotietoa, vaikka eivät itse erityisruokavaliota noudattaisikaan. Käyttäjryhmän toiminta tiedonhaun, sisällöntuotannon ja muiden toimintojen hyödyntämisen kannalta on pääsääntöisesti vastaava kuin kuluttajien käyttäjryhmällä. Oleellisin ero on hallinnoitavien ruokavalioiden muodostuminen esimerkiksi päiväkotiryhmän lapsista, joille kaikille halutaan valmistaa soveltuvaa ruokaa vähällä vaivalla. Tällöin on tärkeää voida vastaanottaa eri käyttäjien lähettämiä ruokavaliotietoja, muodostaa näistä erilaisia yhdistelmiä ja tallentaa yhdistelmät myöhempä käyttöä varten.

Ruoka-alan ammattilaiset

Ruoka-alan ammattilaisia ovat esimerkiksi kotitalousopettajat, ravitsemusterapeutit ja ruokavalioneuvontaa antavien järjestöjen ja liittojen edustajat. He toimivat verkkopalvelussa pitkälti kuluttajien tavoin. Keskeisimmät toiminnot ovat uusiin tuotteisiin ja ruokaohjeisiin tutustuminen, haut sekä muistilistat.

Kaupan alan toimijat

Kaupan alan toimijoita ovat esimerkiksi tukkuliikkeet ja vähittäiskauppiat, joille tärkeimpiä toimintoja ovat elintarviketuotteiden ja -tuottajien tietosisällön seuranta. Erityisesti uutuustuotteet kiinnostanevat kaupan alan toimijoita. Lisäksi tarkoituksena on tukea yhteistyötä ja kommunikointia elintarviketuottajien kanssa, mutta nämä toiminnot eivät ole tämän työn kannalta olennaisia.

Vierailijat

Vierailijoiksi lasketaan kaikki käyttäjät, jotka eivät ole kirjautuneet verkkopalveluun. Näille käyttäjille tarjotaan mahdollisuus suorittaa hakuja, selata palvelun tietosisältöä ja rekisteröityä käyttäjäksi. Vierailijoille sallitaan siis vain sisällön tutkiminen, mutta ei sen tuottamista. Sisällön tuottamista varten käyttäjän tulee tarvittaessa rekisteröityä ja kirjautua sisään palveluun.

5.3 Toteutustekniikat

Palvelun palvelinpään toteutuksessa hyödynnettiin Django-verkkosovelluskehystä, joka tuottaa käyttäjille esitettävät HTML-sivut dynaamisesti. Asiakaspäässä käyttöliittymän toiminnallisuutta laajennettiin ECMAScript- ja AJAX-tekniikoilla.

5.3.1 Django-verkkosovelluskehys

Django on Lawrence Journal-World -sanomalehden tuottama avoimen lähdekoodin verkkosovelluskehys, jonka kehittäminen alkoi vuonna 2003. Yleisesti kehysten ideana on nopeuttaa kehitystyötä tarjoamalla valmiit työkalut tavanomaisimpien verkkosovellusten kehittämiseen liittyvien ongelmien ratkaisemiseen. Djangon keskeisenä ajatuksena on ollut säästää toteuttajan aikaa mahdollistamalla korkealla abstraktiotasolla työskentely, mutta tarvittaessa tarjota mahdollisuus myös asioiden hoitamiseen itse matalammalla tasolla. Seuraava lista sisältää keskeisimmät Djangon tarjoamat ratkaisut.

Kuvaavat URL-osoitteet Django tarjoaa työkalut eleganttien, ja käyttäjäystävällisten tiedostopäätteettömien URL-osoitteiden toteuttamiseen.

Lomakkeiden käsittely Lähes poikkeuksetta verkkosovelluksissa työskennellään lomakkeiden kanssa. Lomakkeiden ja niihin liittyvien avustetekstien näyttä-

minen, käyttäjän syötteiden validointi ja uudelleennäyttäminen virhetilantaisissa ovat toistuvia, työläitä ohjelmointitehtäviä. Django pyrkii helpottamaan ja nopeuttamaan lomakkeiden käsittelyä sopivien kirjastojen avulla.

Tietokantatoimintojen abstrahointi Django lähestyy tiedon mallintamista oliopohjaisesti (ns. object-relation mapper). Ohjelmoija luo tietomallia kuvaavia Python-olioita, joiden pohjalta Django luo automaattisesti tarvittavat tietokantataulut. Käyttäjä työskentelee Python-olioiden kanssa, ja Django vastaa muutosten välittämisestä tietokantaan.

Sivupohjat Sisältö ja esitystapa erotetaan toisistaan käyttämällä sivupohjia. Sivupohjat sisältävät HTML-merkkauksen lisäksi Djangon omaa, laajennettavaa sivupohjamerkkausta.

Kansainvälistäminen Django tukee monikielisten sovellusten kehittämistä tarjoamalla kirjastoja käännostekstien ja kieliriippuvaisten toimintojen käsittelyyn.

Ylläpitäjien käyttöliittymä Django tarjoaa automaattisen ylläpitokäyttöliittymän sovelluksen tietosisällön hallintaan. Tämä nopeuttaa työskentelyä varsinkin sovelluksen kehitysvaiheessa.

Django on toteutettu Python-skriptikielellä, joten sen käyttö tapahtuu kirjoittamalla Python-koodia joka hyödyntää Djangon tarjoamia kirjastoja. Python on tiivis ja ilmaisuvoimainen tulkattava kieli, jonka kirjoittaminen ja käyttäminen on nopeaa, koska sillä kirjoitettuja ohjelmia ei tarvitse erikseen kääntää.

Osana nopeaa sovelluskehitystä Django kannustaa tuottamaan sovelluksen eri osat omina moduuleinaan, jotka eivät tee tarpeettomia oletuksia toistensa sisäisestä toteutuksesta, vaan kommunikoivat toistensa kanssa selkeiden rajapintojen kautta (ns. loose coupling -periaate). Näin sovelluksen eri osia voidaan tarvittaessa vaihtaa toisiin ja uudelleenkäyttää myöhemmin muissa projekteissa.

Django toimii ohjelmistokehityksessä toimivaksi havaitun MVC-arkkitehtuurin mukaisesti. MVC-mallissa (model-view-controller) sovellus jaetaan osiin, joilla jokaisella on omat vastualueensa. Malli (engl. model) vastaa sovelluksen käyttämän tietosisällön ylläpitämisestä ja käsittelemisestä. Näkymä (engl. view) huolehtii mallilta saatujen tietojen esittämisestä käyttäjälle. Ohjain (engl. controller) reagoi käyttäjän syötteisiin käskyttämällä mallia ja näkymää.

Djangossa on käytössä MVC-mallia vastaava MTV-malli (model-template-view). Malli piilottaa tietokannan ilmaisuvoimaisen oliorelaatiokuvauksen avulla. Näkymä muodostuu Python-koodista, joka määrittelee mitä tietosisältöä esitystavan

valitseville sivupohjille (engl. templates) tarjotaan. Sivupohjat tuotetaan HTML-merkkauksen ja Django'n oman sivupohjamerkkauksen (engl. template language) yhdistelmänä.

Django'n käytöllä ei ole merkittävää vaikutusta sisällön mukauttamiseen. Kaikki perinteisten web-teknologioiden avulla toteutettavissa olevat mukautustoiminnot voidaan toteuttaa myös Djangoa käytettäessä, mutta erillisiä työkaluja tähän ei tarjota. Yleinen tuki sivupohjille ja mahdollisuus omien sivupohjatoimintojen tuottamiseen kuitenkin luovat hyvät edellytykset mukautustoiminnallisuuden toteuttamiselle.

Django tarjoaa valmiin ylläpitokäyttöliittämän verkkopalvelun tietosisällön käsittelyyn. Ylläpitokäyttöliittymän kautta tietosisältöalkioita voidaan lisätä, muokata ja poistaa. Ylläpitokäyttöliittymä ei kuitenkaan sovellu laajan käyttäjäkunnan omaavan verkkopalvelun loppukäyttäjien käytettäväksi. Ensinnäkin käyttöoikeuksien rajoittaminen tuottaa ongelmia, koska käyttäjät pääsevät näkemään ja käsittelemään muidenkin käyttäjien yksityisiä tietoja. Lisäksi ylläpitokäyttöliittymän käyttö edellyttää tiettyjen teknisten ratkaisujen, kuten oliopohjaisen mallintamisen ja vierasavainviittausten, perusajatusten ymmärtämistä. Kaikilta käyttäjiltä tällaisen tietämyksen odottaminen on perusteetonta. Ylläpitokäyttöliittymä on kuitenkin tehokas työkalu koulutetulle ylläpitäjälle, jolla on oikeudet palvelun koko tietosisällön käsittelyyn.

Django lähestyy tietosisällön mallintamista oliopohjaisesti. Esimerkiksi ruokaohjetta mallinnettaessa luodaan Python-luokka, jossa määritellään ruokaohjeiden yleiset ominaisuudet, kuten ruokaohjeen nimi ja pääraaka-aine, sekä ruokaohjeiden suhteet muihin tietosisällön luokkiin, kuten käyttäjiin, jotka toimivat ruokaohjeiden kirjoittajina. Lisättäessä palveluun uutta ruokaohjetta, luodaan luokkaan perustuva olio, jonka ominaisuuksille ja suhteille annetaan kyseiselle ruokaohjeelle ominaiset arvot.

Django tukee neljää eri tietokannan hallintaohjelmistoa: MySQLää, PostgreSQLää, Oraclea ja SQLiteä. Koska tietosisällön hallinta tapahtuu olioita käsittelemällä, ei ohjelmoijan tarvitse välittää siitä minkä hallintaohjelmiston päälle tietokanta lopulta rakentuu. Tietosisällön käsittely olioiden kautta on yksinkertaista ja tehokasta, mutta kaikkein monimutkaisimmat tietokantakyselyt eivät tätä kautta onnistu. Tällaisia tilanteita varten Django tarjoaa ohjelmoijalle mahdollisuuden käyttää myös normaaleja SQL-kyselyjä (Structured Query Language).

5.3.2 ECMAScript ja AJAX

ECMAScript on kansainvälisen ECMA-standardointijärjestön (European Computer Manufacturers Association) standardoima, alunperin verkkopalveluiden asiakaspäässä tapahtuvan käyttöliittymän elävöittämiseen tarkoitettu komentosarjakielesi [ECMA 1999]. ECMAScript perustuu Netscapen tuottamaan JavaScript-kieleen, ja käytännössä eri selainohjelmistot toteuttavat standardia eri tavoin. Arkikielessä nimeä JavaScript usein käytetään viittaamaan kaikkiin eri selainohjelmistojen toteuttamiin ECMAScript-murteisiin.

Palvelinpään ohjelmointi perustuu verkkoselaimen lähettämiin HTTP-pyyntöihin, jolloin näiden pyyntöjen välillä ei voida vaikuttaa käyttäjälle näytettävään sisältöön tai reagoida tämän toimiin. ECMAScript mahdollistaa monipuolisemmat käyttöliittymät, koska se toimii osana selainta, ja voi näin ollen esimerkiksi muokata näytettävää dokumenttia hiiren tai näppäimistön käytön perusteella.

ECMAScriptin käyttöön liittyy kuitenkin omat ongelmansa. Eri tavalla standardia toteuttavat selainohjelmistot saattavat tulkita koodia eri tavoin, jolloin seuraukset voivat olla odottamattomat. Kaikki selainohjelmistot eivät tue ECMAScriptiä, ja lisäksi käyttäjä voi poistaa sen käytöstä. Verkkopalvelun toteuttaja ei siis voi luottaa, että komentosarjalla toteuttu toiminnallisuus todella toimisi kaikilla käyttäjillä oletetulla tavalla. Tästä johtuen verkkopalvelun perustoiminnallisuus tulisi olla käytettävissä myös ilman komentosarjatukea. Komentosarjoilla voidaan parannella tätä perustoiminnallisuutta erilaisilla käyttäjälle lisäarvoa tuottavilla ominaisuuksilla.

Ajax-termiä (Asynchronous JavaScript and XML) käytetään kuvaamaan tekniikkapatteria, jolla voidaan toteuttaa entistä monipuolisempia verkkosovelluksia. XMLHttpRequest-pyyntöllä voidaan palvelimelta hakea tietoa siten, ettei koko HTML-dokumenttia täydy päivittää uudella HTTP-pyyntöllä. Asiakaspäässä toimiva JavaScript muokkaa käyttäjälle näytettävää dokumenttia dynaamisesti saadun vastauksen perusteella. Näin verkkopalvelun vuorovaikutus on askeleen verran sujuvampaa.

5.4 Mukauttaminen Aatussa

Aatu-verkkopalvelun toiminnallista määrittelydokumenttia tutkimalla selviää, että palvelua ideotaessa suunnitelmana oli luoda ruokaohjetietokanta, jossa ruokaohjeet ovat mallinnettu niin tarkasti, että valmistusaineiden mukauttaminen käyttäjän ruokavalioprofiilin sopiviksi onnistuisi ajonaikaisesti. Suunnittelutyön edetessä ilmeni, että tällaisen järjestelmän luominen olisi äärimmäisen haastavaa, sillä yhden val-

mistusaineen vaihtaminen saattaa vaikuttaa yllättävällä tavalla siihen, miten muut valmistusaineet käyttäytyvät. Esimerkiksi jauholajin vaihtaminen toiseen saattaa vaikuttaa tarvittavan nesteen määrään, joka puolestaan voi vaikuttaa muihin valmistusaineisiin.

Tarjottavan informaation luotettavuus on toiminnallisessa määrittelydokumentissa nostettu kriittisen tärkeäksi kriteeriksi verkkopalvelulle. Mikäli käyttäjä epäonnistuu palvelusta löytämänsä ruokalajin valmistamisessa, käyttäjän luottamus palvelua kohtaan kärsii. Pahimmillaan käyttäjä hylkää koko palvelun. Tästä johtuen esimerkiksi 95% todennäköisyydellä kelvollisia ruokaohjeita tuottavaa tekoälymukautusta ei pidetty riittävänä: vaatimukseksi asetettiin 100% varmuudella toimivat ruokaohjeet. Käytännössä tämä tarkoittaa sitä, että ruokaohjeista vastaava sisällöntuottaja haluaa testata jokaisen ruokaohjeen toimivuuden ennen ohjeen julkaisemista verkkopalvelussa. Koska jokaisesta ruokaohjeesta tuotetaan useita muunnelmia (arviolta 5-10 kappaletta), siirtyi työn painopiste mukautusjärjestelmän toteuttamisesta tämän suuren tietosisältömäärän syöttämistä tukevien menetelmien kehittämiseen.

Mukautuvuuden ajatusta ei kuitenkaan kokonaan hylätty, vaan päätettiin toteuttaa erilaisia loppukäyttäjän tavoitteita tukevia toimintoja. Kuten kohdassa 2.3 mainittiin, mukautus toteutetaan käyttäjästä kerätyn informaation mukaan luodun käyttäjämallin perusteella. Aatu-palvelussa mukautus perustuu lähinnä käyttäjän ruokavalioon. Koska tällaisen tiedon kerääminen implisiittisesti on käytännössä mahdotonta, turvaudutaan palvelussa eksplisiittiseen ruokavaliotiedon keräämiseen. Ruokavaliotiedon keräämisessä käytetty prosessi kuvataan tarkemmin alikohdassa 5.6.2. Palveluun kirjautumattomat käyttäjät voivat palveluun tutustumisen helpottamiseksi käyttää oletusruokavaliota, joiden perusteella mukautukset voidaan suorittaa. Nämä oletusruokavaliot vastaavat alikohdassa 5.6.1 määriteltyjä ruokaohjeisiin liittyviä soveltuvuustietoja.

Seuraavassa käsitellään yksinkertaisuuden nimissä palvelussa suoritettavia mukautuksia ainoastaan ruokaohjeiden osalta. Vastaavat mukautustoiminnallisuudet toteuttiin myös elintarviketuotetiedoille, ellei erikseen toisin mainita.

Palvelun mukautukset perustavat käyttäjän ruokavalioprofilin vertaamiseen näytettäviin muunnelmiin. Muunnelmat sisältävät tiedot käytetyistä valmistusaineista, jotka voidaan pelkistää listaksi ruoka-aineita. Ruokavalioprofiili puolestaan määrittelee, mitä ruoka-aineita käyttäjä ei saa tai halua syödä. Muunnelman sisältämät ruoka-aineet käydään läpi yksi kerrallaan, ja mikäli yksikin niistä löytyy kiellettyjen ruoka-aineiden listalta, tulkitaan muunnelma käyttäjälle sopimattomaksi.

Tähän soveltuvuusvertailuun liittyy kaksi poikkeusta. Ensiksikin muunnelman valmistusaineet voidaan määritellä tarvittaessa poisjätettäviksi mikäli niiden merkitys lopputuloksen kannalta on vähäinen. Esimerkiksi tietyt mausteet voidaan jättää pois ilman, että valmistettava ruokalaji muuttuu merkittävästi. Tuotteiden osalta poisjätettävä valmistusaine ei ole mielekäs käsite, joten tämä poikkeus ei päde tuotteisiin. Toisen poikkeuksen muodostavat tuotealikategoriat, joiden soveltuvuus on määritelty siten, että mikäli alikategoriaan kuuluu yksikin käyttäjän ruokavalioon soveltuva tuote, katsotaan alikategoria käyttäjälle soveltuvaksi. Tuotteiden valmistusaineina ei voida käyttää tuotealikategorioita, joten tämäkään poikkeus ei koske tuotteita.

Yksittäisessä muunnelmassa saattaa olla kymmeniä valmistusaineita, joista jokainen voi sisältää kymmeniä ruoka-aineita. Tästä johtuen ruoka-aineiden käyminen läpi yksi kerrallaan saattaa olla raskas operaatio. Työmäärän keventämiseksi muunnelmiin päätettiin liittää tieto niiden soveltumisesta yleisimpiin allergiaruokavalioihin. Nämä ovat gluteeniton ruokavalio, gluteeniton ruokavalio kauralla sekä luontaisesti gluteeniton ruokavalio. Esimerkiksi gluteeniton ruokavalio kieltää kerralla yhdeksän eri ruoka-ainetta. Mikäli käyttäjän ruokavalio vastaa jotakin näistä yleisistä ruokavalioista, talletetaan tieto tästä käyttäjän ruokavalioprofiliiin. Vertaamalla näitä tietoja voidaan osa ruokaohjeista todeta käyttäjälle sopimattomaksi ilman, että valmistusaineet käytäisiin läpi yksitellen. Tuotteisiin liitetään lisäksi tieto sopivuudesta laktoosittomaan ja vähälaktoosiseen ruokavalioon. Ruokaohjeiden tapauksessa tätä ei katsottu tarpeelliseksi, koska käyttäjä voi halutessaan valmistaa ruokalajin käyttämällä alkuperäisiä vastaavia laktoosittomia valmistusaineita.

Palveluun lisättyjä ruokaohjeita listataan esimerkiksi nimien, luontipäivien, kategoriatiedon ja käyttäjän suorittamien hakujen perusteella. Osa näistä listauksista rajoittaa näytettävien ruokaohjeiden määrää. Tällaisista lyhyistä listauksista voidaan suodattaa pois käyttäjän ruokavalioon sopimattomat ruokaohjeet. Käyttäjille tarjotaan kuitenkin mahdollisuus nähdä listauksissa myös heille sopimattomat ruokaohjeet. Pitkissä, satoja ruokaohjeita sisältävissä listauksissa suodattamista ei katsottu järkeväksi, koska operaation raskaudesta johtuen sivulataus kestäisi useita sekunteja.

Oletusarvoisesti käyttäjän siirryessä ruokaohjeen sivulle, hänet ohjataan aktiiviseen ruokavalioon soveltuvan muunnelman sivulle. Käytännössä ruokaohjeen muunnelmat käydään läpi siinä järjestyksessä jossa ne on palveluun lisätty, koska muunnelmien lisäysjärjestys kuvastaa sitä miten samankaltaiseksi sisällöntuottaja kokee muunnelman ja ruokaohjeen perusmuunnelman. Mikäli sopivaa muunnelmia ei palvelusta löydy, ohjataan käyttäjä perusmuunnelmaan, mutta hänelle selkeästi huo-

aatu BETA

Tervetuloa, **teuvo!**

Käyttäjät Ruokaohjeet Tuotteet Haku Lisämateriaali

Etusivu > Ruokaohje > Mokkapalat > Mokkapalat riisistä soijakermalla

Valitse kieli Finnish Go

Ruokaohje Muunnelmät (4) Valokuvat Käyttäjät Merkinnot

Mokkapalat riisistä soijakermalla

Tämä muunnelma ei näytä soveltuvan aktiiviseen ruokavaliioosi, koska:

- Muunnelma ei ole soijaton.

Seuraavat tämän ohjeen muunnelmät kuitenkin näyttävät soveltuvan sinulle:

- [Perusohje](#)
- [Mokkapalat riisistä](#)

Valmistuksen kesto	Annosten määrä
Melko nopea (alle 1h)	20

Mokkapalojen tarjoiluohje: [Tähän ohjeeseen liittyvät valokuvat.](#)

Herkulliset mokkapalat ilman suklaata.

Valmistusaineet

Taikina

Määrä	Valmistusaine
3 dl	farinisokeria
4 dl	vettä
3 dl	riisijauho, hieno
2 dl	riisijauho, karkea
2 dl	riisijauho, täysjyvä
2 rkl	Carob-jauhetta
3-4 tl	soodaa
2 dl	öljyä

Ruokaohjekortti

Ruokaohjeen tyyppi: Kotitalousohje

Pääraaka-aine: Muut

Kategoriat: [iäkkiruoat](#) [makeat leivonnaiset](#)

Muunnelmät: [4 muunnelmaa.](#)

Kokeile myös näitä

- [Vuokapulla](#)
- [Puuro](#)
- [Sämpylät](#)

Suosijat

Tällä muunnelmalla ei ole vielä suosijoita. Ole ensimmäinen suosija!

Merkinnot

Tähän muunnelmaan ei ole vielä lisätty merkintöjä. Lisää ensimmäinen merkintä!

Käytössä oleva ruokavaliio

Lääkärin vahvistama

Vaihda ruokavaliio

Lääkärin vahvista

Ota käyttöön

[Hallitse ruokavaliioitasi](#)

Oma suosikki

Merkitse suosikiksesi

Miten merkitsen suosikkeja?

» Voit lisätä muunnelman suosikiksesi esimerkiksi muunnelman oman sivun kautta.

Omat merkinnät

Ei omia merkintöjä

Lisää merkintöjä

Miten lisään merkintöjä?

» Voit lisätä merkintöjä muunnelmiin muunnelman sivulla "merkinnät"-välilehdellä.

Kuva 5.1: Esimerkki muunnelman soveltumattomuuden korostamisesta.

mautetaan muunnelman epäsojivuudesta. Tuotteille vastaavaa toiminnallisuutta ei toteutettu, koska esimerkiksi tavallinen tuote ja sen laktoositon vastine miellettiin muunnelmien sijasta erillisiksi tuotteiksi.

Kuten kuvassa 5.1 esitetään, käyttäjän päätyessä aktiiviseen ruokavaliionsa sopimattoman muunnelman sivulle, kerrotaan hänelle kyseisen muunnelman sopimattomuudesta lisäämällä sivun alkuun ja loppuun sivun muusta väriteemasta selkeästi erottuva tekstikenttä, jossa kerrotaan mitkä muunnelman valmistusaineista eivät sovellu käyttäjälle. Näin käyttäjä näkee mihin mukautus perustuu ja voi itse arvioida sen oikeellisuutta. Lisäksi hän voi mahdollisesti soveltaa ohjetta korvaamalla sopimattomat valmistusaineet oman tietämyksensä perusteella. Jos palvelussa on ruo-

kaohjeesta käyttäjälle sopivia muunnelmia, tarjotaan tässä yhteydessä käyttäjälle myös linkit näihin muunnelmiin.

Palvelussa käytetään yksinkertaisia mukautuksia myös käyttöoikeuksien perusteella: ylläpitäjille ja sisällöntuottajille näytetään linkkejä näiden käyttämiin toimintoihin, mutta muilta käyttäjiltä nämä linkit piilotetaan. Vaikka itse mukautus onkin tällaisissa tapauksissa automaattinen, ei se perustu käyttäjistä kerätyn informaation perusteella luotuun käyttäjämalliin, vaan ylläpitäjän erikseen asettamiin käyttöoikeuksiin.

5.5 Tiedon mallintaminen Aatussa

Tässä kohdassa kuvataan lyhyesti palvelun tietomallin ruokaohjeisiin liittyviä osia siinä määrin kuin se on sisällönhallinnan ja mukautustoimintojen näkökulmasta tarpeellista. Palvelun tietomallia käsitellään laajemmin ja tarkemmin Teemu Immosen samassa hankkeessa tekemässä diplomityössä VERKKOPALVELUN TIETOMALLI JA INFORMAATIOSISÄLLÖN SÄILYVYYDENHALLINTA KETTERÄSSÄ OHJELMISTOPROJEKTISSA.

5.5.1 Lähtökohdat

Hankkeen keskeisenä yhteistyötahona toimi Pirkanmaan Allergia- ja Astmayhdistys ry. Yhdistys tarjosi erikoisruokavalioihin sopivia ruokaohjeita verkkosivullaan jo ennen hankkeen alkua. Nykyiset ruokaohjesivut ovat staattisia, käsin luotuja ja ylläpidettyjä HTML-tiedostoja. Ruokaohjeiden luokittelu sivustolla perustuu niiden soveltuvuuteen eri erityisruokavalioihin. Mikäli ohje kuuluu useampaan eri luokkaan, on ohjeet kopioitu useammalle eri sivulle. Tämä vaikeuttaa ruokaohjeiden ylläpitoa. Kaikille käyttäjille näytetään sama sisältö, ja heidän täytyy itse arvioida soveltuuko ruokalaji heille.

Ruokaohjeet on mallinnettu kevyesti rekenteisessä muodossa. Ohjeen nimi toimii otsikkona. Soveltuvuustiedot kerrotaan vapaatekstinä, jolloin tiedon mekaaninen käsittely on haastavaa. Valmistusaine, tarvittava määrä ja mahdolliset korvaavat valmistusaineet kerrotaan niin ikään vapaatekstinä. Kotitalousohjeiden tapauksessa valmistusaineet annetaan listassa, kun taas ammattikeittiöohjeilla informaatio esitetään taulukossa, jossa kerrotaan erikseen valmistusaineen osto- ja käyttöpaino. Lisäksi ammattikeittiöohjeilla kerrotaan ohjeen vapaapaino, saanto sekä annosten määrä ja koko, myöskin vapaatekstinä. Valmistusohje selostetaan yhdessä teksti-

kappaleessa, jolloin ohjeen silmäiltävyys on heikko ja ruoanvalmistukseen keskittyvä lukija kadottaa helposti paikkansa tekstissä.

Yhteistyötahoilta vapaamuotoisen ryhmäkeskustelun kautta kerätyn informaation perusteella tietomallin rakenteesta tehtiin sivistynyt arvaus, joka on dokumentoitu hankkeen toiminnallisessa määrittelyssä. Ohjelmistotuotannolle tyypillinen ongelma havaittiin myös tällä kertaa: Ruoanlaiton ja ruoka-aineallergioiden asiantuntijoilla ei ollut tiedon mallintamiseen liittyvää tietotaitoa, kun taas ryhmän teknisesti suuntautuneilla jäsenillä ei ollut käsitystä siitä, millaisia erikoistapauksia ruokaohjeita ja -aineita kuvaavan tietomallin tulee voida käsitellä.

Sivistyneen arvauksen pohjalta lähdettiin ketterän ohjelmistokehityksen hengessä kehittämään ensimmäisiä käyttöliittymäprototyyppejä. Tietomallia muokattiin ja laajennettiin käytännön kehitystyön yhteydessä esille tulleiden huomioiden perusteella. Lisätietoa saatiin esimerkiksi allergialääkäreiltä ja Elintarvikeviraston pakkausmerkintäopasta tutkimalla. Kehitystyön alussa tietomalli koki jatkuvia merkittäviä muutoksia, mutta työn edetessä se hiljalleen vakaantui.

5.5.2 Ratkaisut

Seuraavassa tutkitaan miten ruoka-aineiden, ruokaohjeiden ja ruokavalion tietomallin suunnittelussa tehdyt ratkaisut vaikuttavat tietosisällön mukauttamiseen ja sisällönhallintaan. Dokumentointi on rajattu näihin kolmeen luokkaan, koska ne kuvaavat kattavasti palvelun kannalta olennaiset mukautus- ja sisällönhallintatoiminnot. Karsittua tietomallia kuvataan UML-kaaviona liitteessä A.

Ruoka-aineet

Keskeisin osa palvelun tietomallia ovat ruoka-aineet, sillä niin ruokaohjeet, tuotteet kuin ruokavaliot hyödyntävät niitä omissa toteutuksissaan. Ruoka-ainelista pohjautuu Kansanterveyslaitoksen ravitsemusyksikön ylläpitämän Fineli-tietopankin listaukseen elintarvikkeiden valmistukseen käytetyistä raaka-aineista (ks. [Kansanterveyslaitos 2008]). Tästä listasta karsittiin sellaisia aineita, joiden katsottiin olevan toteutettavan palvelun näkökulmasta elintarviketuotteita. Listan työstämistä jatkettiin ruoka-alan ammattilaisita koostuvissa työryhmissä vapaamuotoisen keskustelun perusteella. Lisäksi listaa on täydennetty palvelun testauksen yhteydessä — esimerkiksi ruokaohjeita lisättäessä — havaittujen puutteiden osalta.

Ruoka-ainetta mallinnettaessa huomioitiin mahdolliset samalle ruoka-aineelle kuuluvat eri nimet ja nimien kirjoitusasut. Esimerkiksi adukipapu kirjoitetaan ylei-

sesti myös muodossa ”adsukipapu” ja ”adzukipapu”. Sisältöä tuottavat palvelun käyttäjät voivat käyttää valitsemaansa kirjoitusasua, mutta järjestelmä tallentaa esimerkiksi muunnelman yhteyteen yleisimmäksi kirjoitusasuksi määritellyn nimen. Väärinymmärrysten välttämiseksi käyttäjälle ilmoitetaan nimen korvaamisesta synonyymillä. Näin välttytään BigOvenin käytössä ilmenneiltä ongelmilta. Käyttämällä ainoastaan yleisimpää nimeä kaikki palveluun tuotettu sisältö on yhtenäisessä muodossa, josta loppukäyttäjät hyötyvät helpomman tietosisältöjen vertailun ansiosta ja palvelun tuottajat toteutustyön yksinkertaistuessa.

On myös mahdollista, että ruoka-aine on niin sanottu koosteaine, joka koostuu useammasta ruoka-aineesta, kuten alikohdassa 4.2.3 esitellyssä Dish_Master-järjestelmässä. Esimerkiksi maito koostuu maitoproteiineista ja laktoosista. Käyttäjä voi olla allerginen yhdelle tai useammalle näistä ainesosista. Tällaisia tilanteita varten tietomalliin luotiin koosteaineen käsite. Käyttäjä voi merkitä itselleen sopimattomaksi joko kooste-aineen yksittäisen osa-aineen, tai koko koosteaineen, jolloin kaikki sen osa-aineet ovat hänelle sopimattomia. Esimerkiksi maitoa sisältävä ruokaohje tai tuote on sopimaton käyttäjälle, joka on merkinnyt yhdenkin maidon ainesosista sopimattomaksi.

Ruokaohjeet

Ruokaohje on mallinnettu siten, että yksittäiseen ruokaohjeeseen liittyy yksi tai useampi muunnelma. Näin kaikille käyttäjille ei tarvitse näyttää samaa sisältöä, vaan heitä voidaan palvella paremmin tarjoamalla samankaltaisia, juuri heidän ruokavalionsa soveltuvia muunnelmia.

Muunnelmat päätettiin mallintaa siten, että valmistusaineena voi toimia yksittäinen ruoka-aine, elintarviketuote, useita tuotteita sisältävä tuotealikategoria tai toinen muunnelma. Koska myös tuote on mallinnettu siten, että sen sisältämät ruoka-aineet ovat tiedossa, voidaan muunnelman valmistusaineet lopulta pelkistää joukoksi viittauksia ruoka-aineisiin.

Valmistusaineen rakenne kuvataan vapaatekstinä. Jos kaikki mahdolliset vaihtoehdot haluttaisiin tarjota valmiina, tulisi valintalistasta pitkä, ja kaikki vaihtoehdot eivät muodostaisi järkevää yhdistelmää kaikkien valmistusaineiden kanssa. Vapaatekstiä yksikäsitteisemmälle määrittelytavalle ei keksitty merkittäviä sovelluksia. Yhtenäisen merkintätavan saavuttaminen jää nyt kuitenkin sisällöntuottajien vastuulle.

Tarvittava valmistusaineen määrä kuvataan tekstikentän ja valmiiksi annettujen yksikkövaihtoehtojen yhdistelmällä. Tekstikenttään voidaan syöttää yksikäsitteinen

numeroarvo, arvoväli tai tekstikuvaus kuten ”hyppysellinen”. Kenttä voidaan jättää myös tyhjäksi. Yhdistämällä tähän tarvittaessa yksi valmiiksi annetuista mittayksiköistä voidaan valmistusaineen määrä kuvata monipuolisesti. Kattavan yksikkölistauksen varmistamiseksi mittayksiköt on koottu reseptiikan asiantuntijoilta. Valmistusaineen määrän kuvauksessa ei päästä aivan alikohdassa 4.2.2 esitellyn RecipeML-dokumenttimäärittelyn tarkkuuteen, mutta saavutettu tarkkuustaso arvioitiin käytötarkoitukseen riittäväksi.

Valmistusaineeseen liitetään myös tieto siitä, onko valmistusaine pakollinen vai poisjätettävissä. Oletusarvoisesti valmistusaine on pakollinen, mutta esimerkiksi tietty mausteet voidaan jättää pois ohjeesta ilman, että ruokalaji merkittävästi muuttuu. Poisjätettäviä valmistusaineita ei huomioida vertailtaessa muunnelman sopivuutta käyttäjän ruokavalioon, koska käyttäjä voi valmistaa ruokalajin myös ilman kyseistä valmistusainetta. Dish_Master-järjestelmässä voitiin valmistusaineelle antaa painoarvot, jotka kuvasivat tarkemmin valmistusaineen vaikutusta lopputuloksen makuun, ulkonäköön ja ravintoarvoon. Katsoimme yksinkertaisen totuusarvon olevan kuitenkin mukautuksen kannalta parempi vaihtoehto, sillä näin ruokaohjeesta saadaan käyttäjälle yksiselitteisesti soveltuva tai soveltumaton. Lisäksi sisällöntuottajan olisi työlästä arvioida jokaisen valmistusaineen tarkkaa vaikutusta ruokalajiin, eikä tästä ylimääräisestä työstä saataisi merkittävää lisäarvoa.

Pirkanmaan Astma- ja Allergiayhdistyksen verkkosivuilla ruokaohjeet on luokiteltu ylemmällä tasolla sen perusteella millaiseen ruokavalioon ruokalaji on suunniteltu, ja alemmalla tasolla ruokalajin tyyppin (esimerkiksi ”pääruoka”) perusteella. Koska Aatu-palvelussa eri ruokavaliot huomioidaan suodattamalla sisältöä, ei ruokavaliotietoa tarvitse käyttää ruokaohjeiden luokittelussa. Sen sijaan ruokaohjeet luokitellaan ruokaohjeen tyyppin (kotitalous- tai ammattikeittiöohje), kategoriatiendon ja pääraaka-aineen perusteella. Näille ominaisuuksille annetaan valmiit vaihtoehdot, jolloin sisällöntuottajien on helpompaa käyttää yhtenäistä merkkaustapaa. Jotta valmiiksi annetut vaihtoehdot eivät rajoittaisi sisällöntuottajien työtä, päätettiin käytettävistä vaihtoehdoista yhdessä ruokaohjeiden asiantuntijoiden kanssa. Mikäli annetut vaihtoehdot eivät sovi lisättävään ruokaohjeeseen, voi sisällöntuottaja valita arvon ”muu”. Lisäksi palveluun on helppo lisätä uusia vaihtoehtoja tarpeen ilmetessä.

Muunnelmiin ja tuotteisiin voidaan liittää ominaisuuksia (liitteessä A esitetyn UML-kaavion Feature-luokka), kuten ”luontaisesti gluteeniton” tai ”maidoton”. Näiden ominaisuuksien pääasiallinen tarkoitus on tukea silmäiltävyyttä niillä käyttäjillä, joilla syystä tai toisesta ei ole ruokavalioprofilia käytössä. Tästä voi kuitenkin olla hyötyä myös mukautukselle. Mikäli jo näiden ominaisuuksien yhteensopivuuden

vertailun perusteella ruokaohje voidaan tulkita sopimattomaksi, ei raskasta valmistusainelistan läpikäyntiä tarvitse suorittaa.

Ruokavalioprofiili

Mukautuksen perustana toimivat ruokavalioprofiilit muodostuvat viittauksista ruoka-aineisiin, sekä tiedosta ovatko viitatut ruoka-aineet kiellettyjä vai sallittuja. Mahdollisuus tallentaa informaatio kahdella eri tavalla perustellaan tehokkuusnäkökohdilla: mikäli vain muutama ruoka-aineista on sallittu, täytyisi soveltuvuusvertailu suorittaa erikseen satojen sopimattomien ruoka-aineiden osalta. Käytännössä sallittujen ruoka-aineiden määrittely määrittelee negaation kautta myös kielletyt ruoka-aineet.

Soveltuvuusvertailun nopeuttamiseksi ruokavalioon liitetään erikseen gluteenia ja laktoosia koskevat tiedot, jolloin valmistusaineita ei tarvitse välttämättä käydä läpi ollenkaan. Esimerkiksi jos ruokavaliio on merkitty luontaisesti gluteenittomiksi, mutta muunnelma kyseiseen ruokavalioon sopimattomaksi, on muunnelma käyttäjälle sopimaton valmistusainelistasta välittämättä. Tämä tiedon redundantti tallentaminen perustellaan siis tehokkuussyillä. Lisäksi nämä ominaisuudet esitetään julkisten ruokavalioiden yhteydessä, jolloin muiden käyttäjien on helpompaa muodostaa käsitys ruokavalion rajoitteista.

Käyttäjä voi lisätä itselleen rajoittamattoman määrän erilaisia ruokavalioprofiileja. Käyttäjä voi esimerkiksi luoda oman ruokavalioprofiilin jokaiselle lapselleen. Lisäksi näistä profiileista voi olla erilliset kotikäyttöön ja hoitopaikalle tarkoitetut, lääkärin vahvistamat versiot. Joillekin käyttäjille voi olla tarpeen luoda erilliset profiilit siitepölykaudeksi ja talviajaksi.

Kaikista luomistaan ruokavalioprofiileista käyttäjä voi valita kerrallaan yhden aktiivisen ja yhden julkisen ruokavalion. Aktiivista ruokavaliota käytetään sisällön mukauttamiseen, kuten esimerkiksi tuotelistausten suodattamiseen ja muunnelmien soveltuvuuden tarkistamiseen. Julkinen ruokavalioprofiili on ainoa profiili, joka näytetään käyttäjän omalla sivulla, mikäli käyttäjä on sallinut sivujensa näyttämisen. Näin käyttäjä voi kätevästi näyttää esimerkiksi ystävilleen ja muille palvelun käyttäjille, millaista ruokavaliota hän noudattaa. Aktiivinen ja julkinen ruokavaliio voivat olla yksi ja sama, käyttäjä voi vaihtaa niitä milloin tahansa, eikä hänen ole pakko valita aktiivista tai julkista ruokavaliota lainkaan.

5.6 Sisällönhallinta Aatussa

Aatu-verkkopalvelun keskeistä tietosisältöä hallitaan web-pohjaisen käyttöliittymän kautta. Varsinkin uuden sisällön tuotanto riittävän rakenteisessa muodossa saattaa olla työläs ja toteuttajalleen raskas tehtävä. Tästä johtuen palvelua suunniteltaessa ja toteutettaessa kiinnitettiin erityistä huomiota tämän sisällöntuotantoprosessin tukemiseen ja helpottamiseen.

Sivuston informaatioarkkitehtuuri on suunniteltu etukäteen ennen palvelun julkaisua. Jokaiselle sisältötyypille on määritelty sivustolla omat sijaintinsa, jossa julkaistu materiaali näytetään. Ruokaohjeet, tuotteet, ja muu tietosisältö listataan omilla alasuillaan, jolloin sivuston informaatioarkkitehtuuri ei pääse rämettymään, vaan käyttäjät löytävät tietyn sisällön aina sille määrättyltä paikalta. Edes ylläpitäjätason käyttöoikeudet omaavat käyttäjät eivät voi valittuun informaatioarkkitehturiin vaikuttaa, koska tällaiset muutokset vaativat kooditason muutoksia.

5.6.1 Ruokaohjeiden lisääminen palveluun

Seuraavassa tutkitaan ruokaohjeiden lisäämisen tueksi toteutettuja yksittäisiä ratkaisuja. Vastaavia ratkaisuja sovellettiin myös tuotetietojen lisäämisen tukemiseen.

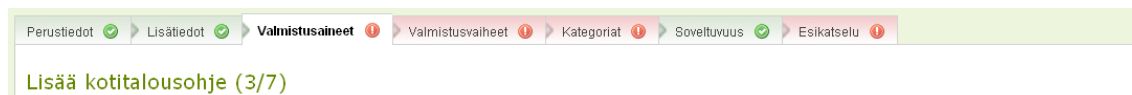
Vaiheistettu lomake

Mikäli kaikki ruokaohjemuunnelmasta kerättävä informaatio kerättäisiin yhden HTML-lomakkeen kautta, tulisi lomakkeesta todella pitkä. Sen lisäksi, että käyttäjä joutuisi lomaketta täyttäessään vierittämään sivua, ongelmaksi muodostuisi syöteentarkistuksen virheelliseksi merkitsemien kenttien löytäminen tarvittaessa. Tältä vältytään jakamalla tietojen keruu erillisille, loogisia kokonaisuuksia muodostaville lomakkeille. Työn vaiheistaminen pienentää myös sisällöntuottajan kognitiivista kuormitusta.

Ruokaohjeiden ja muunnelmien syöttäminen päätettiin jakaa kuuteen vaiheeseen. Ensimmäisessä vaiheessa syötetään ruokaohjeen tai muunnelman nimi sekä kuvaus. Uutta ruokaohjetta luotaessa valitaan myös ruokaohjeen pääraaka-aine. Muunnelma käyttää aina samaa pääraaka-ainetta kuin ruokaohje, johon muunnelma liittyy. Seuraavassa vaiheessa kerätään muunnelmaan liittyviä lisätietoja, eli valmistuksen kesto ja ohjeesta saatavien annosten määrä. Ammattikeittiöohjetta luotaessa lisätietoihin kuuluvat myös yksittäisen annoksen koko ja ohjeen raakapaino sekä saanto. Kahdessa seuraavassa vaiheissa kerätään valmistusaineet sekä valmistusoh-

je. Viimeisissä vaiheissa käyttäjä voi valita mihin kategorioihin muunnelma kuuluu, ja millaiset soveltuvuustiedot muunnelmaan liitetään.

Käyttäjän syötettyä yhden vaiheen tiedot, siirtyy lomakkeen täyttö automaattisesti seuraavaan vaiheeseen. Tämän lisäksi käyttäjä voi vapaasti liikkua eri vaiheiden välillä välilehtimetaforaan perustuvan navigoinnin kautta. Kuvassa 5.2 esitetään esimerkinäkymä välilehtimetaforasta. Lisäksi välilehdet kertovat värikoodauksen ja ikonin avulla mitkä välilehdet käyttäjä on jo täyttänyt.



Kuva 5.2: Esimerkki vaiheistetun lomakkeen välilehtimetaforasta.

Missä tahansa tietojen syötön vaiheessa käyttäjä voi esikatsella muunnelmaa blogeista ja wikeistä tutulla tavalla, jossa käyttäjän jo syöttämät tiedot upotetaan sivupohjaan, jotta sisällöntuottaja näkee miltä ruokaohje tulee loppukäyttäjien silmissä näyttämään. Tietojen tallentamiseen käytetty painike on sijoitettu tämän esikatselunäkymän loppuun, jotta tietojen esikatselu ennen tallentamista muodostuisi tavaksi. Esikatselunäkymässä myös huomautetaan puutteellisesti täytetyistä kentistä ja tarjotaan linkit kyseisen kentän sisältämään vaiheeseen. Käyttäjän on pakko korjata virheet, koska puutteellisten ruokaohjeiden tallentaminen palveluun on estetty.

Soveltuvissa kohdissa käyttäjälle tarjotaan joko pudotusvalikon tai valintaruutulistan muodossa valmiit vaihtoehdot tekstikentän sijaan. Erityisesti näin toimitaan kerätessä luokitteluun käytettävää tietoa, koska yhtenäisestä merkintätavasta saatava hyöty korostuu tällaista tietoa käytettäessä. Lisäksi välttyään kirjoitusvirheilta ja tiedon syöttäminen nopeutuu, kun käyttäjä voi kirjoittamisen sijasta valita sopivat vaihtoehdot. Jotta valmiit vaihtoehdot eivät liikaa rajoittaisi sisällöntuotantoa, päädyttiin tarjoamaan vaihtoehto ”muut” niissä kohdissa, joissa se katsottiin järkeväksi.

Ulkoamuistamisen tarvetta pyritään vähentämään tarjoamalla yhteenvedo toisessa työvaiheessa syötetystä tiedosta sellaisessa työvaiheessa, jossa tiedon muistaminen saattaa olla tarpeen. Valmistusaineita ja -vaiheita syötettäessä käyttäjä näkee aiemmin syöttämiensä valmistusaineiden ruoka-aineet tai valmistusvaiheiden alut. Lisäksi valmistusvaiheita syötettäessä kerrotaan mitä työvaiheotsikoita sisällöntuottaja käytti valmistusaineiden yhteydessä ja päinvastoin. Vaiheessa, jossa syötetään soveltuvuustietoja kerrotaan käytetyt valmistusaineet, jotta päätöksen tekeminen helpottuisi. Vastaavanlainen toiminto on tyypillinen esimerkiksi blogeissa,

joissa käyttäjälle näytetään aiemmin käytetyt avainsanat, jottei käyttäjä loisi useita samaa tarkoittavia avainsanoja.

Ruokaohjeen lisäämisessä käytettyihin lomakkeisiin toteutettiin myös muita käyttöä helpottavia toimintoja. Lomakkeiden jokaiseen kenttään liitetään opasteita. Koska varsinkin rutinoituneelle käyttäjälle nämä tekstit saattavat olla vain hidasteena lomakkeen täytölle, ovat ohjeet oletusarvoisesti piilossa, mutta käyttäjä saa ne halutessaan helposti näkyville ja takaisin piiloon. Mikäli käyttäjä antaa virheellisen syötteen tai jättää pakollisen kentän täyttämättä, huomautetaan tästä korostamalla virheelliset kentät, jolloin käyttäjä näkee välittömästi mitä kohtia lomakkeesta hänen täytyy korjata.

Toistuvat työvaiheet

Aatu-palvelusta on tarkoitus rakentaa erityisruokavalioihin sopivia ruokaohjeita sisältävä tietopankki. Keskeistä on, että jokaisesta ruokaohjeesta luodaan useita, erilaisten ruokavalioiden erikoistarpeet huomioivia muunnelmia. Käytännössä nämä muunnelmat saattavat erota alkuperäisestä ohjeesta vain vähän. Oletettavasti tyyppisimpiä muutoksia ovat muutaman valmistusaineen (ja niitä vastaavien valmistusvaiheiden) muuttaminen.

Ruokaohjeita lisäävän käyttäjän työmäärää pyritään vähentämään tarjoamalla mahdollisuus hyödyntää jo syötettyjä ruokaohjeita tai muunnelmia uutta muunnelmaa luotaessa. Tällöin käyttäjä voi esimerkiksi valmistusaineita lisätessään yhdellä painalluksella täyttää uuden muunnelman valmistusainelistan pohjana käyttämänsä ruokaohjeen tai muunnelman valmistusainelistalla. Tähän valmistusainelistaan tarvittavat muutokset voidaan tehdä muokkaamalla, uudelleenjärjestämällä ja poistamalla yksittäisiä valmistusaineita. Näin käyttäjä välttyy puuduttavalta lähes identtisten valmistusainelistojen uudelleenkirjoittamiselta jokaisen muunnelman lisäämisen yhteydessä.

Olemassa olevan tiedon hyödyntäminen on käytössä vaiheistetun lomakkeen vaiheissa, joissa syötetään lisätietoja, valmistusaineita, valmistusvaiheita, ja kategorioita. Sen sijaan toiminto ei ole käytössä perustietoja ja soveltuvuustietoja lisättäessä. Perustiedot ovat muunnelmakohtaisia, joten olemassa olevien tietojen käyttäminen ei tässä yhteydessä ole mielekäästä. Soveltuvuustiedot puolestaan pyritään päättelämään valmistusainetietojen perusteella, joten oletusarvoisesti käyttäjän tarvitsee vain hyväksyä ehdotetut soveltuvuustiedot niiden syöttämisen sijasta.

Valmistusaineet

Uuden ruokaohjeen lisäämisen työläin vaihe on valmistusaineiden lisääminen, koska jotta ruokaohjeen soveltuvuus käyttäjän ruokavalioprofiiliin voidaan tarkistaa, tulee jokainen ruokaohjeen valmistusaine ilmoittaa koneen ymmärtämässä, yksikäsitteisessä muodossa. Valmistusaineella tarkoitetaan tässä yhteydessä tietorakennetta, johon kuuluu itse käytettävän aineen lisäksi tieto tarvittavasta määrästä, aineen pakollisuudesta, sekä työvaiheesta jossa valmistusaine käytetään. Käytettävä aine voi käytännössä olla yksittäinen ruoka-aine, palvelusta löytyvä elintarviketuote, useita tuotteita sisältävä tuotealikategoria tai ruokaohjemuunnelma, mutta seuraavassa listassa käytetään edellä mainituista yhteisnimitystä ”ruoka-aine” ymmärrettävyyden tähden. Valmistusainetta syötettäessä käyttäjälle esitetään seuraavat kentät:

Työvaihe Työvaihe-kenttä kuvastaa mihin valmistuksen työvaiheeseen valmistusaine kuuluu. Esimerkiksi leivonnaisia valmistettaessa työvaiheet voisivat olla taikinan, täytteen ja kuorrutteen valmistaminen. Oletuksena valmistusaine kuuluu aina edelliseen nimettyyn valmistusvaiheeseen. Tieto ei ole pakollinen.

Valmistusaine Valmistusaineina voidaan käyttää vain palvelun entuudestaan tuntemia ruoka-aineita, eli niiden on löydyttävä palvelun tietokannasta. Periaatteessa tämä saattaa estää tietyn ruokaohjeen lisäämisen kokonaan kunnes kyseinen ruoka-aine on lisätty palveluun. Tämä katsottiin kuitenkin perustelluksi, koska ilman yksikäsitteistä tietoa valmistusaineen sisältämistä ruoka-aineista putoaa pohja koko mukautusjärjestelmän luotettavuudelta.

Kuvaus Valmistusainekentällä kerätään tieto ruoka-aineesta koneellisen käsittelyn mahdollistavassa muodossa. Koska tällainen muoto ei välttämättä ole ruokaohjetta lukevan käyttäjän kannalta ymmärrettävin, kerätään valmistusaineesta vapaamuotoinen kuvaus, joka ruokaohjetta näytettäessä näytetään käyttäjälle. Sisällöntuottaja voi liittää kuvaukseen täsmentävää tietoa ruoka-aineesta ja esimerkiksi kirjoittaa ruoka-aineen nimen tarvittavaan taivutusmuotoon. Tieto on pakollinen.

Määrä Valmistusaineen tarvittava määrä syötetään tekstimuodossa, jolloin käyttäjä voi antaa yksikäsitteiden numeroarvon, arvovälin tai kuvailevamman vapaamuotoisen arvon. Tieto ei ole pakollinen.

Yksikkö Yksikkö-kenttä tarjoaa listan valmiiksi annettuja arvoja, kuten ”kappaletta” ja ”litraa”. Tällä pyritään yhtenäiseen merkkaustapaan valmistusaineen määriä määritellessä. Tieto ei ole pakollinen.

Poisjätettävissä Oletusarvoisesti kaikki luetellut valmistusaineet ovat ruokaohjeen valmistamisen kannalta pakollisia. Valintaruudun avulla sisällöntuottaja voi merkata tietyn valmistusaineen tarvittaessa poisjätettäväksi. Poisjätettäviä valmistusaineita ei huomioida tarkistettaessa ruokaohjeen soveltuvuutta käyttäjän ruokavalioon.

Ammattikeittiöohjeita määriteltäessä valmistusaineen syöttölomakkeessa on kahdet määrä- ja yksikkökentät. Näin ruokaohjeessa voidaan kertoa suurtalouskeittiöille tärkeät osto- ja käyttöpainot, jotka eivät kaikille valmistusaineille ole yksi ja sama asia. Esimerkiksi hedelmien kuoret, joita ei käytetä valmistuksessa, sisältyvät ostopainoon, mutta eivät käyttöpainoon.

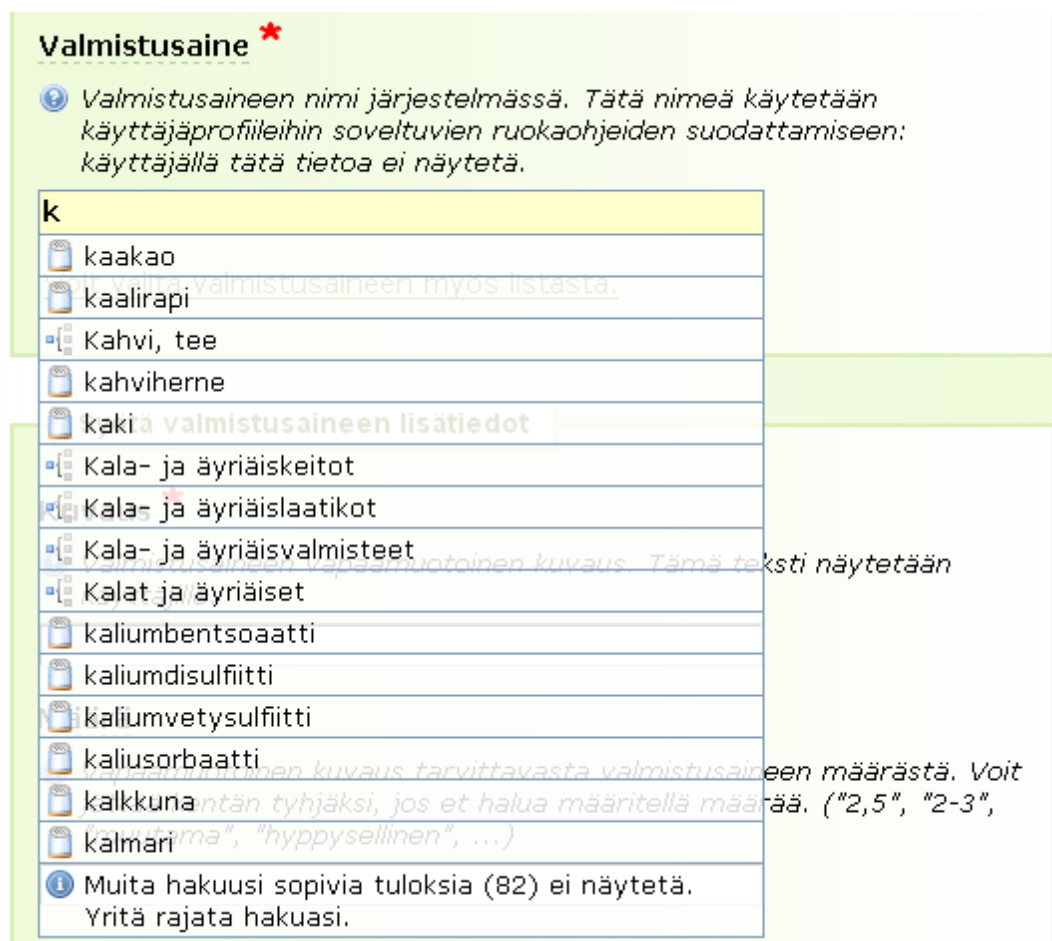
Mahdollisuus käyttää olemassa olevaa muunnelmaa toisen muunnelman valmistusaineena lisää palveluun niin sanotun aliohjeen käsitteen. Esimerkiksi perunasose toimii itsenäisenä ruokaohjeena, mutta sitä voidaan käyttää myös yhtenä liha-perunasoselaatikon valmistusaineena. Kun palvelusta löytyy jo perunasoseen ohje, ei sisällöntuottajan tarvitse kirjoittaa ohjetta uudestaan liha-perunasoselaatikon ohjetta lisättäessä. Lisäksi perunasoseen ohjeeseen liittyvät muutokset tarvitsee päivittää vain yhteen paikkaan.

Valmistusaineena käytettäviä aineita on niin suuri määrä, että kaikkien ulkoamuistaminen on käytännössä mahdotonta. Esimerkiksi pelkästään ruoka-aineita palvelusta löytyy noin 500. Tämä asetti haasteen tiedon esittämiselle: miten todella pitkä valintalista voidaan esittää HTML-lomakkeessa siten, että sen käyttäminen on tehokasta? Pitkä listaus valintaruutuja (engl. checkbox) pakottaisi käyttäjän vierittämään sivua ja oikean aineen löytäminen kestäisi tarpeettoman pitkään. Alasvetovalikkona (engl. drop down box) esitettyä valintalista vaivaisi samat ongelmat.

Koska valmistusaineen löytyminen tietokannasta annetussa kirjoitusasussa on välttämätöntä, kehitettiin sisällöntuotannon tueksi kaksi erillistä ratkaisua. Ensimmäinen ratkaisuista on valmistusaineen valitseminen listasta. Valmistusainetietojen keräyslomakkeeseen sijoitettiin linkki, jonka kautta sisällöntuottaja pääsee erilliselle sivulle, joka sisältää palstoitetut listauksesta kaikista palvelusta löytyvistä ruoka-aineista, elintarviketuotteista, tuotealukategorioista ja ruokaohjemuunnelmista. Listauksessa jokaisen aineen nimi toimii linkkinä, joka palauttaa käyttäjän takaisin lomakesivulle ja täyttää kyseisen aineen nimen lomakkeeseen. Listaussivulla käynti ei hukkaa muita lomakkeeseen jo syötettyjä tietoja. Toinen ratkaisuista on niin sanottu ehdottaja-järjestelmä, joka esitellään seuraavaksi.

Ehdottaja-järjestelmä

Ensimmäisenä pitkien valintalistojen korvaajana mieleen tullut ratkaisuna päätettiin testata esimerkiksi Googlen Suggest-hakupalvelusta ja BigOven-ohjelmistosta tuttua ehdottaja-järjestelmää. Ehdottaja-järjestelmässä käyttäjälle näytettävää vaihtoehtojen listaa suodatetaan dynaamisesti käyttäjän syötteiden perusteella. Mikäli jokin ehdotetuista vaihtoehtoista on käyttäjän hakema, ei tämän tarvitse kirjoittaa sanaa loppuun, vaan sen voi valita suodatetusta ehdotuslistasta. Esimerkiksi käyttäjän syöttäessä valmistusainekenttään kirjaimen k, listataan tälle k-kirjaimella alkavia valmistusaineita. Koska tämä lista sisältää vielä kymmeniä valmistusaineita, esitetään käyttäjälle 15 (aakkosjärjestyksessä) ensimmäistä valmistusainetta, sekä tieto siitä, montako hakuun sopivaa valmistusainetta listalle ei mahtunut. Esimerkinomainen ruutukaappaus ehdottaja-järjestelmästä on esitetty kuvassa 5.3. Jos käyttäjä seuraavaksi painaa l-kirjainta, ehdotuslista rajautuu pelkästään ”klementtiin”. Näin käyttäjä saa samalla tiedon, että palvelusta ei löydy muita ”kl”-alkuisia valmistusaineita.



Kuva 5.3: Esimerkki ehdottaja-järjestelmän toiminnasta.

Esitetty ratkaisu koettiin yhteistyötahojen kanssa järjestetyssä läpikäynnissä toimivaksi. Ratkaisun haittapuolena voidaan pitää vaatimusta JavaScriptia tukevista selainohjelmista. Tästä johtuen toiminto toteutettiin siten, että JavaScript-tuen puute ei estä lomakkeen käyttöä: lomake toimii muuten normaalisti, mutta ehdotuksia sisältävä listaa ei näytetä. Lisäksi päätettiin tarjota vaihtoehtona edellä mainittu, ilman JavaScriptiä toimiva valitse listasta -toiminto.

Soveltuvuustietojen ehdottaminen

Soveltuvuusvertailujen tehostamiseksi ruokaohjemuunnelmiin liitetään tieto niiden soveltuvuudesta erilaisiin keliakikkojen ruokavalioihin. Lisäksi ruokaohjesivujen silmäiltävyyden parantamiseksi muunnelmia luotaessa kerätään tieto sen soveltuvuudesta maidottomaan, kananmunattomaan ja soijattomaan ruokavalioon. Pelkästään ruoka-aineista koostuvan ruokaohjeen soveltuvuustietojen arvioiminen on sisällöntuottajana toimivalle reseptiikan asiantuntijalle rutiininomainen tehtävä. Tehtävä muuttuu työläämmäksi, jos valmistusaineina käytetään elintarviketuotteita, tuotealikelkategorioita tai toisia muunnelmia.

Sisällöntuotannon tueksi ruokaohjeen vaiheistettuun lisäyslomakkeeseen liitettiin toiminto, joka käy valmistusainelistan läpi ja ehdottaa tämän perusteella sopivia soveltuvuustietoja. Koska elintarviketuotteet, tuotealikelkategoriat ja muunnelmat voidaan pelkistää ruoka-ainelistoiksi, onnistuu tällaisen listan käsittely koneelta nopeammin kuin ihmiseltä. Sen lisäksi, että toiminto suosittelee tiettyjä soveltuvuusarvoja, se kertoo mitkä valmistusaineet tekivät muunnelmasta soveltumattoman tiettyyn ruokavalioon. Näin sisällöntuottajan on helpompi arvioida konepäätelmien paikkansapitävyyttä. Koska konepäätelyyn ei täysin luoteta, ei soveltuvuustietota talleta suoraan järjestelmän suositusten perusteella, vaan sisällöntuottajan täytyy hyväksyä soveltuvuustiedot. Vastuu tietojen oikeellisuudesta on siis lopulta sisällöntuottajalla.

5.6.2 Ruokavalioprofilien lisääminen palveluun

Myös uuden ruokavalion lisääminen palveluun tapahtuu vaiheistetun lomakkeen avulla. Käyttäjä aloittaa uuden ruokavalion lisäämisen palveluun valitsemalla onko kyseessä salliva vai kieltävä ruokavalio. Salliva ruokavalio määrittelee kaikki ne ruoka-aineet, joita käyttäjä voi syödä, kun taas kieltävä ruokavalio kertoo mitä aineita käyttäjä ei voi syödä. Sallittuja ruokavalioita käyttävät ne vakavasti allergiset henkilöt, joiden on helpompaa määritellä ruokavalionsa muutamia sallitut ruoka-aineet, kuin kieltää lähes kaikki palvelusta löytyvät ruoka-aineet. Yhteistyötahoihin

kuuluvat allergia-asiantuntijat arvioivat, että tällaisia käyttäjiä on pieni vähemmistö kaikista käyttäjistä.

Kieltävää ruokavaliota määriteltäessä ensimmäinen vaiheessa käyttäjälle tarjotaan pikavalitsimet, jotka vastaavat yleisimpiä erityisruokavalioita. Keliakikoille nämä ruokavaliot ovat gluteeniton ruokavalio, gluteeniton ruokavalio jossa käytetään myös puhdasta kauraa ja luontaisesti gluteeniton ruokavalio. Laktoosi-intoleraatioille tarjotaan vähälaktoosinen tai laktoositon ruokavalio. Pikavalitsimen käyttö määrittelee automaattisesti kaikki kyseisen ruokavalion kieltämät ruoka-aineet kielletyiksi seuraavan vaiheen lomakkeeseen. Käyttäjä ei voi muuttaa pikavalitsimella kielletyksi määriteltyä ainetta sallituksi muuttamatta ensin pikavalitsimen arvoa, jotta välttyttäisiin ristiriitaista tietoa sisältäviltä ruokavalioprofiileilta. Sallivien ruokavalioiden yhteydessä pikavalitsimia ei katsottu mielekkääksi ratkaisuksi.

Kieltävän ruokavalion määrittelyn toinen ja sallivan ruokavalion määrittelyn ensimmäinen vaihe on ruoka-aineiden valinta. Riippumatta siitä, valitseeko käyttäjä kiellettyjä vai sallittuja ruoka-aineita, on käytössä samanlainen lomake, josta osa on näkyvissä kuvassa 5.4. Sekaannusten välttämiseksi käyttäjälle kerrotaan selkeästi sivun otsikossa, painikkeissa ja ohjeteksteissä onko hän lisäämässä kieltävää vai sallivaa ruokavaliota. Lomake koostuu kategorioittain listatuista ruoka-aineista. Oletusarvoisesti kategoriat on luhistettu kokoon (engl. collapsed) siten, että vain kategorian nimi ja siihen kuuluvien ruoka-aineiden määrä on näkyvissä. Kuvassa 5.4 luhistettuja kategorioita ovat ”Muna”, ”Muu” sekä ”Muu vihannes tai juures”. Kategorian nimen valitsemalla lomake laajenee kategoriaan kuuluvien ruoka-aineiden valintaruutulistaksi, josta käyttäjä voi valita haluamansa aineet. Valinnan suorittamisen jälkeen kategoria luhistuu jälleen kokoon. Jotta käyttäjän ei tarvitse muistaa, mitä ruoka-aineita hän on jo valinnut, listataan valitut ruoka-aineet kategorian nimen alla. Tämän lisäksi käyttäjä voi käyttää edellä esiteltyä ehdottaja-järjestelmää ruoka-aineiden valintaan yksi kerrallaan.

Ruoka-aineiden valitsemista helpotetaan erilaisilla apukeinoilla. Alikohdassa 5.5.2 esiteltyistä koosteaineista on kuvassa 5.4 esimerkkinä maissi, jonka valinta valitsee kerralla kaikki palvelusta löytyvät neljä maissipohjaista ruoka-ainetta. Toisen käyttöä helpottava keino on ruoka-aineiden löytyminen useammasta kategoriasista. Käyttäjä voi etsiä esimerkiksi soijaa kategorioista ”herneet ja pavut”, ”muu” ja ”öljyt”, joten se listataan kaikkien näiden kategorioiden alla. Käyttäjän valitessa tällaisen aineen yhdestä kategoriasta, valinta näytetään siirryttäessä muihin kategorioihin, joihin ruoka-aine kuuluu. Käyttäjälle tarjotaan mahdollisuus myös kaikkien kategorian ruoka-aineiden valintaan kerralla: tämä auttaa esimerkiksi pavuille allergisia henkilöitä, jotka ovat usein allergisia tietyn papulajin sijasta kaikille pavuille.

Muna (6)

Muu (69)

Tästä kategoriasta kielletyiksi valittuna: Soijaproteiini ja liivate.

Muu vihannes tai juures (43)

Vilja tai jauho Valitse kaikki tämän kategorian ruoka-aineet

amarantti
 Samaa tarkoittavat ruoka-aineet: **E 123, revonhätä.**

arrow

durra

hirssi

kaura *Valitsemasi pikavalinta pakottaa tämän ruoka-aineen kielletyksi.*

kinoa
 Samaa tarkoittavat ruoka-aineet: **kinoajauho, kinoajuoma, kvinoa, quinoa.**

maissi

maissi (vihannes) *(kategoriasta: Muu vihannes tai juures)*

maissijauho

maissitärkkelys

maissiöljy *(kategoriasta: Öljy)*

ohra *Valitsemasi pikavalinta pakottaa tämän ruoka-aineen kielletyksi.*
 Samaa tarkoittavat ruoka-aineet: **ohratärkkelys, ohratärkkelyssiirappi.**

Kuva 5.4: Sallittujen tai kiellettyjen ruoka-aineiden valintaan käytettävä lomake

Vaiheistetun lomakkeen viimeisessä vaiheessa kerätään ruokavalion nimi ja vapaaehtoinen kuvailuteksti. Nimen tulee olla käyttäjäkohtaisesti ainutkertainen: usealla käyttäjällä voi olla ruokavalio nimellä ”gluteeniton”, mutta yhdellä käyttäjällä voi sama nimi olla käytössä vain kerran.

Käyttäjä voi luoda yhdistelmäruokavalioita kahdesta tai useammasta jo olemassa olevasta ruokavaliostaan. Tähän käytetään edellä esiteltyä vaiheistettua lomaketta yksinkertaisempaa lomaketta, jossa kysytään luotavan ruokavalion nimi, kuvailuteksti sekä mitkä käyttäjän ruokavalioprofileista yhdistetään. Oletusarvoisesti yhdistelmäruokavaliosta tulee kieltävä ruokavalio, mutta salliva ruokavalio luodaan mikäli yli puolet palvelun sisältämistä ruoka-aineista olisi kiellettyjä, kuten sallivia ruokavaliota yhdistäessä helposti käy. Yhdistelyn tuloksena syntyy täysin uusi ruokavalioprofiili, eli alkuperäisiä profileja ei muuteta.

5.6.3 Tietosisällön muokkaaminen

Tietosisällön muokkaaminen tapahtuu muunnelman, tuotteen tai ruokavalion oman sivun kautta. Mikäli käyttäjällä on sisällön muokkaamiseen tarvittavat käyttöoikeu-

det, näytetään tälle wikeistä tutut muokattavan sisällön läheisyyteen sijoitetut linkit, jotka johtavat muokausnäkymään. Esimerkiksi muunnelman tapauksessa valmistusaineita pääsee muokkaamaan heti valmistusainelistan jälkeen sijaitsevan linkin kautta.

Uutta tietosisältöä tuotettaessa tiedot kerättiin loogisista kokonaisuuksista koostuvan vaiheistetun lomakkeen avulla, mutta tietosisällön muokkaus tapahtuu yksittäisen lomakkeen kautta. Tietosisältöä muokattaessa voidaan olettaa kyseessä olevan useimmiten pienten virheiden korjaaminen, jolloin koko vaiheistetun lomakkeen läpikäyminen olisi tarpeettoman hidasta. Myös wikeissä vastaavia muokattavan tietosisällön läheisyyteen sijoitettuja linkkejä käytetään koko kyseisen sivun sijasta vain sivun pienen osan muokkaamiseen, jolloin käyttäjän on helpompi hahmottaa käsittelemänsä kokonaisuus. Muokausnäkyvässä ei siis ole mahdollista siirtyä toisille välilehdille kuten tietosisältöä luotaessa. Tällä pyritään selkeyttämään käyttäjälle koska hän on lisäämässä uutta sisältöä, ja koska muokkaamassa jo olemassa olevaa. Myöskään tehtyjen muutosten esikatselua ei ole käytössä, vaan muutokset näkyvät välittömästi sivulla, jossa kyseinen tietosisältö käyttäjälle esitetään.

Esimerkiksi muunnelman perustietoja muokatessa käyttäjälle näytetään sama lomake kuin tietoja kerättäessä, sillä poikkeuksella, että muunnelman tiedot on täytetty valmiiksi lomakkeeseen. Käyttäjä voi muokata tietoja ja tallentaa muutokset. Mikäli annetut tiedot läpäisevät samat syötteiden virhetarkastelut kuin muunnelmaa lisättäessäkin, talletaan muutokset ja käyttäjä ohjataan takaisin muokatun muunnelman sivulle. Vaihtoehtoisesti käyttäjä voi palata muunnelman sivulle tallentamatta muutoksia.

Kieltävän ruokavalionprofiilin muokkauksessa tulee huomioida, että vaiheistetun lomakkeen kaksi vaihetta vaikuttavat suoraan toisiinsa: pikavalitsimet pakottavat tietyt ruoka-aineet osaksi kiellettyjen ruoka-aineiden listaa. Käyttäjän vaihtaessa pikavalinnan sallivammasta kieltävämpään, voidaan uudet kielletyt ruoka-aineet lisätä ruokavalion tietoihin ongelmitta. Jos käyttäjä sen sijaan vaihtaa pikavalinnan kieltävämmästä sallivampaan, ei voida tietää haluaako hän todella poistaa kiellettyjen ruoka-aineiden listalta kaikki ne ruoka-aineet, jotka vanha valinta kielsi, mutta uusi sallisi. Tällaisissa tilanteissa käyttäjälle kerrotaan, että ruoka-ainelistaan on voinut jäädä ylimääräisiä ruoka-aineita, ja pyydetään käyttäjää muokkaamaan ruoka-ainelistaa tarvittaessa.

5.6.4 Julkaisutilojen hallinta

Palvelussa käytetään kahdentasoisia käyttöoikeuksia. Sisällöntuottajat voivat luoda uutta ja muokata olemassa olevaa tietosisältöä, mutta eivät julkaista niitä yleisön nähtäväksi. Ylläpitäjät puolestaan voivat luonnin ja muokkaamisen lisäksi julkaista tietosisältöä. Ruokaohjeiden tapauksessa sisällöntuottajat voivat muokata muiden tuottamaa sisältöä, mutta elintarviketuottajat voivat käsitellä vain oman yrityksensä tuotteita.

Osana tietosisällön hallintaa palveluun toteutettiin blogijärjestelmistä tuttu yksinkertainen julkaisutilojen hallintajärjestelmä, jonka avulla muunnelmiin ja tuotteisiin voidaan liittää metatietoa, joka vaikuttaa siihen mille käyttäjäryhmille sisältö on näkyvissä. Ylläpitäjille tuotettiin erillinen yksinkertainen ylläpitonäkymä, jossa muunnelmia ja tuotteita voi listata eri ominaisuuksien mukaan järjestettynä ja muokata niiden julkaisutiloja. Seuraavassa listataan palvelun yhdeksän julkaisutilaa ja esitellään miten ne vaikuttavat eri käyttäjäryhmien toimintaan.

- **Keskeneräinen**

- Sisällöntuottaja voi merkitä sisällön keskeneräiseksi, mikäli hän joutuu syystä tai toisesta keskeyttämään kirjoitustyön ennen sisällön valmistamista.
- Ylläpitäjä näkee, että sisältö on keskeneräistä eikä siis ole tarkoitettu vielä julkaistavaksi.
- Loppukäyttäjä ei näe keskeneräistä sisältöä.

- **Odottaa hyväksyntää**

- Sisällöntuottaja, jolla ei ole oikeuksia julkaista sisältöä palveluun merkitsee valmiiksi saamansa sisällöt odottamaan ylläpitäjän hyväksyntää.
- Ylläpitäjä näkee ylläpitokäyttöliittymässä listauksen hyväksyntää odottavasta sisällöstä ja tarkastettuaan sisällön voi joko julkaista, hylätä tai palauttaa sisällön sisällöntuottajalle korjattavaksi.
- Loppukäyttäjä ei näe hyväksyntää odottavaa sisältöä.

- **Hylätty**

- Sisällöntuottaja ei voi muokata ylläpitäjän hylkäämää sisältöä.
- Ylläpitäjä voi hylätä julkaistavaksi tarjotun sisällön, mikäli hän katsoo, ettei sisältö edes muokattuna tulisi täyttämään sisällölle asetettuja laatuksiteerejä.

- Loppukäyttäjä ei näe hylättyä sisältöä.

- **Palautettu korjattavaksi**

- Sisällöntuottaja voi muokata korjattavaksi palautettua sisältöä ylläpitäjältä saamansa palautteen mukaan, jonka jälkeen sisältö siirtyy takaisin odottamaan ylläpitäjän hyväksyntää.
- Ylläpitäjä näkee ylläpitokäyttöliittymästä, että sisällöntuottaja ei ole vielä tehnyt vaadittuja muutoksia, eikä hänen siis tarvitse tehdä sisällölle mitään.
- Loppukäyttäjä ei näe korjattavaksi palautettua sisältöä.

- **Julkaistu**

- Sisällöntuottaja voi muokata, piilottaa ja poistaa jo julkaistua sisältöä. Mikäli muokatun sisällön valmistusaineet on annettu täydellisenä koneen ymmärtämässä muodossa, pysyy sisältö julkaistuna. Jos valmistusainelista on puutteita, siirtyy sisältö tilaan *puutteellinen valmistusainelista*.
- Ylläpitäjä voi poistaa jo julkaistua sisältöä poikkeustapauksissa, kuten jos elintarviketuottaja on lopettanut toimintansa, eikä tuotetta ole enää saatavilla.
- Loppukäyttäjä voi katsella julkaistua sisältöä.

- **Puutteellinen valmistusainelista**

- Sisällöntuottaja voi halutessaan muokata puutteellisen valmistusainelistan sisältävää sisältöä. Mikäli muokkauksen tuloksena saadaan kelvollinen valmistusainelista, siirtyy sisältö automaattisesti takaisin *julkaistu*-tilaan.
- Ylläpitäjä voi muokata puutteellisen valmistusainelistan sisältävää sisältöä. Lisäksi hän näkee mitä valmistusaineita sisällöntuottaja on sisältöön ehdottanut, ja voi tarvittaessa lisätä nämä palvelun ruoka-ainelistaan.
- Loppukäyttäjä voi katsella tässä tilassa olevaa sisältöä, mutta hänelle huomautetaan selkeästi, että sisältöä ei ole vahvistettu, joten siihen tulisi suhtautua varauksella.

- **Piilotettu**

- Sisällöntuottaja voi piilottaa jo julkaistua sisältöä loppukäyttäjiltä. Esimerkiksi kausituotteet voidaan piilottaa ajankohtina, jolloin tuotetta ei ole saatavilla. Piilotetut tuotteet voidaan palauttaa *julkaistu*-tilaan ilman ylläpitäjän erillistä hyväksyntää.

- Ylläpitäjä näkee piilotetun sisällön erillisessä ylläpitoäkymässä.
- Loppukäyttäjä ei näe piilotettua sisältöä.

- **Poistettu**

- Sisällöntuottaja voi poistaa jo julkaistua sisältöä esimerkiksi silloin, kun tuote poistuu markkinoilta. Sisällöntuottaja ei näe poistettua sisältöä, eikä siten voi myöskään palauttaa sitä julkaistuu tilaan.
- Ylläpitäjä näkee poistetun sisällön erillisessä ylläpitoäkymässä, ja voi poikkeustapauksissa palauttaa sen julkaistuksi.
- Loppukäyttäjä ei näe poistettua sisältöä.

- **Poistettu ylläpidon toimesta**

- Sisällöntuottaja ei näe eikä voi käsitellä ylläpidon poistamaa sisältöä.
- Ylläpitäjä voi poistaa erikoistapauksissa mitkä tahansa palveluun lisätyt ruokaohjeet ja tuotteet. Ylläpidon poistama sisältö näkyy ylläpitoäkymässä, ja se voidaan tarvittaessa palauttaa julkaistuksi.
- Loppukäyttäjä ei näe ylläpidon poistamaa sisältöä.

Julkaisutilajärjestelmän tarve havaittiin vasta hankkeen loppuvaiheilla, joten järjestelmä toteutettiin mahdollisimman yksinkertaisena ja pienellä työmäärällä. Tästä johtuen järjestelmässä on paljon parannettavan varaa. Järjestelmä ei esimerkiksi tue ylläpitäjien ja sisällöntuottajien välistä keskustelua tilanteessa, joissa sisältö palautetaan korjattavaksi, vaan kommunikointi tulee hoitaa esimerkiksi sähköpostin välityksellä. Jo julkaistusta ja muokatusta tietosisällöstä olisi hyvä säilyttää kaksi versiota, joista alkuperäistä näytettäisiin loppukäyttäjille sillä aikaa kun muokatun version tietojen oikeellisuutta tarkistetaan, mutta versionhallinta katsottiin liian työlääksi toteuttaa.

Yksinkertaisuudestaan huolimatta julkaisutilat ovat varsin hyödyllinen piirre verkkopalvelussa. Sen avulla voidaan sisällöntuotannon työtaakka hajauttaa useille käyttäjille, mutta silti keskittää vastuu julkaistavan materiaalin laadukkuudesta pienemmälle joukolle. Keskenäisen ja hyväksyntää odottavan tietosisällön piilottaminen loppukäyttäjiltä auttaa ylläpitämään laadukkaan ja luotettavan verkkopalvelun imagoa. Jo julkaistun tietosisällön piilottaminen ja poistetuksi merkitseminen lopullisen poistamisen sijaan voi säästää sisällöntuottajia tarpeettomalta työltä tilanteissa, joissa sisältö halutaan myöhemmin palauttaa palveluun.

5.7 Verkkopalvelun tietosisältö ja tekijänoikeudet

Verkkopalvelun tietosisällön tuottamiseen ja hallintaan liittyy edellä käsiteltyjen teknisten reunaehtojen lisäksi myös muita vaatimuksia, jotka tulee huomioida verkkopalvelua toteutettaessa. Useita yhteistyötahoja sisältävässä hakkeessa näistä tärkeimmäksi nousevat tekijänoikeuskysymykset. Tekijänoikeuslaki ([Tekijänoikeuslaki]) takaa kirjallisen tai taiteellisen teoksen tekijälle tekijänoikeuden (1§), joka takaa haltijalleen yksinomaisen oikeuden määrätä teoksesta valmistamalla siitä kappaleita ja saattamalla se yleisön saataviin (2§). Immateriaalioikeudellisen lainsäädännön, johon tekijänoikeuslaki kuuluu, tarkoituksena on näin edistää luovaa toimintaa (vrt. esim. [Haarmann 2001]).

Tekijänoikeussuojaa ei kuitenkaan myönnetä mille tahansa tuotokselle. Suojaa annetaan vain niissä tapauksissa, jolloin kysymyksessä on itsenäinen ja omaperäinen teos ([Haarmann 2001]). Haarmannin (2001) mukaan niin sanotun teoskynnyksen ylittyminen vaihtelee teostyypeittäin: kirjalliset teokset ylittävät teoskynnyksen suhteellisen helposti, kun taas taidekäsityön tapauksessa itsenäisyyden ja omaperäisyyden vaatimus on asetettu korkealle.

Tekijänoikeus ei suojaa teoksesta ilmenevää tietoa, ideoita, aiheita tai periaatteita ([Tekijänoikeus]). Tekijänoikeus ulottuu vain konkreettisiin teoksiin, ei niiden sisältöön, teemaan tai motiiveihin ([Haarmann 2001]). Vaikka tietty verkkopalvelu siis ylittäisi teoskynnyksen, ei tekijänoikeus koskisi sen tarjoamaa informaatiota, vaan se olisi vapaasti kaikkien hyödynnettävissä. Informaation esitystavan suora kopiointi ja uudelleenlevittäminen ilman lupaa voitaisiin kuitenkin tulkita lain kieltämäksi.

Edellä annettujen määritelmien perusteella on vaikeaa arvioida, ylittävätkö verkkopalvelussa julkaistavat ruokaohjeet tai elintarvikkeiden tuotetiedot teoskynnyksen. Toisaalta kyseessä ovat kirjalliset teokset, joiden teoskynnys on matala, mutta toisaalta omaperäisyyttä on vaikeaa perustella esimerkiksi tuotetiedoissa, joiden esitysmuotoon vaikuttaa osaltaan jo pakkausmerkintälaki. Riitatapauksissa teoskynnyksen ylittämisestä päättää oikeusistuimien tapauskohtaisesti.

Tekijänoikeuslain ([Tekijänoikeuslaki], 49§) mukaan ”sillä, joka on valmistanut tietokannan, jonka sisällön kerääminen, varmistaminen tai esittäminen on edellyttänyt huomattavaa panostusta, on yksinomainen oikeus määrätä työn koko sisällöstä tai sen laadullisesti tai määrällisesti arvioiden olennaisesta osasta valmistamalla siitä kappaleita ja saattamalla se yleisön saataviin”. Aatu-verkkopalvelun tietosisältö taltioidaan tietokantaan, ja sen kerääminen edellyttää huomattavaa panostusta. On jälleen vaikeaa arvioida, mille taholle tai tahoille oikeus määrätä tietokannan

sisällöstä kuuluu hankkeessa, jossa eri tahot ovat eri rooleissa, mutta yhteistyössä tuottaneet verkkopalvelun tietosisällön.

Luku 6

Toteutuksen onnistumisen arviointi

Tässä luvussa arvioidaan verkkopalvelun toteutuksen onnistumista ja käsitellään hankkeen aikana esille tulleita jatkokehitysideoita.

Hankkeen loppuvaiheessa, ennen palvelun virallista julkistamista, palvelua testattiin kolmessa suljetussa katselmointitilaisuudessa. Katselmointeihin osallistui tarkkailijoina toimivien toteutusryhmän jäsenten lisäksi 11 testaajaa, jotka edustivat kohdassa 5.2 esiteltyjä ylläpitäjä-, sisällöntuottaja- ja loppukäyttäjärühmiä. Testaajat suorittivat 12 katselmointitehtävää, jotka olivat:

- Ruokavalioprofiilin luonti ja käyttö (kuusi testaajaa).
- Ruokaohjeiden tuottaminen (kaksi testaajaa).
- Elintarviketuotetietojen tuottaminen (neljä testaajaa).

Testit suoritettiin pääosin keskustelevan havainnoinnin menetelmällä, jossa testaaaja suoritti testitehtävän ja samanaikaisesti kertoi tarkkailijoille ajatuksiaan palvelun käytöstä. Poikkeuksena tähän suoritettiin yksi tuotetietojen lisäämistä käsittelevä kolmen testaaajan ryhmäläpikäynti, jossa testaajat ja tarkkailijat vapaamuotoisesti keskustelivat tuotteen lisäämiseen liittyvistä ongelmista näkymä kerrallaan. Varsinaiset testit kestivät noin tunnin. Näiden testien jälkeen jokaisessa katselmointitilaisuudessa pidettiin keskustelutilaisuus, jossa testaajat saivat kertoa mielipiteensä palvelusta yleisesti sekä mainita palvelusta löytyviä hyviä ja vielä kehitystä kaipaavia ominaisuuksia.

Katselmointeja suoritettaessa verkkopalvelun toiminnoista oli toteutettu loppukäyttäjien prototyypit, joissa:

- Toiminnon toteuttaja on suorittamiensa testien perusteella todennut sen toimivaksi.
- Virhetarkastelu toimi karkealla tasolla esimerkiksi syötteiden osalta.
- Toiminnon avulla tuotettu tietosisältö voidaan tallentaa osaksi palvelun pysyvää tietosisältöä.

Testaajat kokivat palvelun käytön yleisellä tasolla positiiviseksi kokemukseksi. Erityisesti palvelun tarjoama runsas tietosisältö koettiin hyödylliseksi. Useat käyttäjät kuitenkin kokivat sisällön esitystavan sekavaksi, koska näkymiin oli sisällytetty liikaa informaatiota, jolloin oikean informaation ja tehtävien suorittamiseen käytettyjen toimintojen löytäminen vaikeutui. Ilman lisätestausta on vaikeaa arvioida, selkeytyisikö nykyinen sisällön esitystapa käyttäjille, mikäli he jatkaisivat palvelun käyttöä pidempään. Toisaalta lienee kohtuutonta odottaa vierailijan käyttävän verkkopalvelua edes testitehtävien suorittamiseen käytettyä tuntia, jos he eivät löydä etsimäänsä sisältöä.

Suuren sisältömäärän esittäminen pienessä tilassa on aina haastavaa. Yksinkertaisin ratkaisu olisi näytettävän sisällön karsiminen, mutta tätä varten tulisi tarkasti selvittää, mitkä sisällöt ja toiminnot ovat käyttäjille tärkeimpiä. Sisällön jakaminen useammalle sivulle ei sekään ole ongelmaton ratkaisu, sillä kaiken asiakokonaisuuteen liittyvän sisällön löytämisen edellyttämä siirtyminen sivulta toiselle vie aikaa, voi häiritä käyttäjää ja pahimmillaan johtaa eksymiseen. Palvelun julkistamisen yhteydessä tapahtuva massapilotointi tarjoanee lisätietoa, jonka pohjalta informaation esitystapaa voidaan kehittää.

Katselmointitilaisuuksien yhteydessä havaitut selkeimmät puutteet koskivat ohjetekstejä. Käyttäjät eivät ymmärtäneet avata oletusarvoisesti piilotettuja ohjetekstejä, eivätkä aina huomanneet edes näkyvillä olevia ohjetekstejä. Koska ohjetekstit olivat toiminnon toteuttajan kirjoittamia, ei niiden ymmärrettävyys ollut aina paras mahdollinen toimintoa ensikertaa käyttäviä käyttäjiä ajatellen. Lisäksi osa käyttöliittymässä käytetyistä käsitteistä jäi käyttäjille epäselviksi.

Palvelun ohjeita päätettiin parantaa luomalla palveluun erillinen ohjeisuus, jossa kuvataan miten käyttäjä voi parhaiten hyödyntää palvelun toimintoja. Lisäksi toimintojen ja lomakkeiden yhteyteen sijoitetaan ytimekkäitä opasteita, jotka auttavat kyseisen toiminnon käytössä. Opasteista viitataan ohjeisuuteen, jotta käyttäjä voi lukea toimintoon liittyvän kattavamman ohjeistuksen. Ohjeisuudesta vas-

taavasti viitataan toimintoihin, jotta käyttäjä pääsee ohjeet luettuaan helposti itse kokeilemaan toimintoa.

Ruokavalioprofilin määrittelyssä havaittiin ongelmia pikavalintojen käytössä ja ruoka-aineiden luokittelussa. Käyttäjät, joiden ruokavalioon ei liittynyt gluteenia tai laktoosia koskevia rajoitteita, kokivat pikavalintojen olevan liian hallitsevasti esillä. Harmilliseksi koettiin myös pikavalinnan valituiksi pakottamat ruoka-aineet, joita käyttäjä ei voinut muuttaa muuttamatta itse pikavalintaa. Lisäksi käyttäjillä oli vaikeuksia löytää tiettyjä ruoka-aineita useita kymmeniä ruoka-aineita sisältävistä kategorioista.

Ruokavalion pikavalintoja muutetaan siten, että ne eivät pakota ruoka-aineita kielletyiksi, vaan käyttäjä voi muuttaa valintoja ruoka-ainelistauksessa. Ennen ruokavalion tallentamista tarkastetaan vastaako ruokavalio vielä esimerkiksi gluteenitonta ruokavaliota, ja liitetään ruokavalioon vastaava tieto. Lisäksi pikavalintojen yhteyteen liitetään yksikäsitteinen lista niiden kieltämisestä ruoka-aineista, josta käyttäjä näkee helposti miten pikavalinta vaikuttaa ruokavalioon. Ruokavalion määrittelyyn lisätään myös pikavalinnat munattomalle, maidottomalle ja soijattomalle ruokavaliolle. Ruoka-aineiden löydettävyyttä kategorioittain järjestetystä listasta päätettiin parantaa tarkentamalla kategoriointia, jolloin yksittäiseen kategoriaan tulee vähemmän ruoka-aineita.

Testeissä havaittiin myös ohjelmointivirheistä johtuvia ongelmia, joista pienimmät aiheuttivat lähinnä ihmetystä, mutta vakavimpien seurauksena kaikkia tehtäviä ei voitu toteuttaa täysin suunnitellulla tavalla. Kaikki havaitut ohjelmointivirheet pyritään korjaamaan ennen palvelun virallista julkistamista.

Katselmoinneissa havaitut puutteet koskivat lähinnä käyttöliittymää, johon muutosten tekeminen on esimerkiksi järjestelmän tietomalliin verrattuna helppoa. Verkkopalvelun toteutuksen onnistumisen keskeisimpänä mittarina voitaneen kuitenkin pitää loppukäyttäjien tyytyväisyyttä, joten toteutuksen onnistumista voidaan arvioida perinpohjaisesti vasta palvelun julkaisun jälkeen saadun, pitkäaikaisempaa käyttöä koskevan kattavamman palautteen perusteella.

6.1 Jatkokehitys

Hankkeen aikana esille nousi jatkokehitysideoita, joita ei kuitenkaan ajan puutteessa ehditty toteuttaa alkuperäisen hankkeen puitteissa. Seuraavassa arvioidaan näiden ideoiden toteutuskelpoisuutta ja pohditaan, mitä lisäarvoa toiminnot toteutuessaan tarjoaisivat loppukäyttäjille.

Yhdeksi keskeisimmäksi kehityskohteeksi havaittiin tietosisällön versionhallinta, joka toteutuessaan mahdollistaisi aiempien versioiden palauttamisen sekä mahdollisuuden näyttää loppukäyttäjille tietosisällön tiettyä versiota sillä välin, kun uudempi versio odottaa ylläpitäjän hyväksyntää. Myös mahdollisuus lukita tietosisältö muokkausten ajaksi oli hyödyllinen, jotta välttyttäisiin yhtäaikaisten muokkausten aiheuttamilta ongelmilta. Koska eri sisällöntuottajat käsittelevät eri sisältöjä, riski yhtäaikaistulle muokkauksille on melko pieni, mutta olemassa. Toteutukseltaan nämä ominaisuudet ovat melko vaativia, ja palvelun käyttöönotto onnistuu ilman niitäkin.

Koska useat elintarviketuottajat toimittavat tuotteensa markkinoille tukkujen kautta, eivät he tiedä missä liikkeissä ja millä hinnalla heidän tuotteitaan myydään. Näiden tietojen löytyminen palvelusta säästäisi loppukäyttäjiä tarpeettomalta liikkeiden kiertämiseltä. Mikäli tätä informaatiota ei tulla saamaan myöskään tukuilta, voidaan palvelua kehittää siten, että loppukäyttäjät voisivat lisätä ostamiensa tuotteiden hinta- ja saatavuustiedot palveluun muiden loppukäyttäjien nähtäville. Vaikka käyttäjät voivat nykyisellään toteutuksella annotoida tuotteita liittämällä niihin avainsanoja, ei näiden vaapamuotoisten tietojen käsittely koneellisesti ole mahdollista, jolloin esimerkiksi tuotteen hintojen vertailu on tarpeettoman vaikeaa. Erilliset toiminnot hinta- ja saatavuustietojen keräämiseen lienevät näinollen loppukäyttäjille erittäin hyödyllisiä verrattuna toteutuksen vaativuuteen. Myös mahdollisuus saatavuustietojen näyttämiseen kartalla koettiin mielenkiintoiseksi paikkatiedon sovelluskohteeksi. Uhkana tälle toiminnolle voidaan pitää tietojen oikeellisuuden ja ajantasaisuuden ylläpitämisestä koituvaa työmäärää.

Nykyisellään käyttäjät eivät voi palvelun kautta raportoida ylläpitäjille havaitsemistaan virheistä tai puutteista tuotteiden ja ruokaohjeiden sisällössä. Kynnys sähköpostin lähettämiseen saattaa olla liian korkea, joten toiminto tätä varten olisi varmasti hyödyllinen, etenkin kun huomioidaan, että tiedon oikeellisuus on palvelun keskeisiä tavoitteita. Yksinkertainen mahdollisuus liputtaa tuote tai ruokaohje ja lisätä lyhyt kuvaus mahdollisesta virheestä on saavutettavaan hyötyyn nähden helppo toteuttaa. Ylläpitäjille pitää vastaavasti toteuttaa yhteenvedonäkymä tehdyistä virheilmoituksista, mutta myös sen toteuttaminen on melko suoraviivainen toimenpide.

Vaikka Django-verkkosovelluskehys tukee monikielisten verkkopalveluiden luomista, toteutettiin Aatu-verkkopalvelu vain suomenkielisenä. Kieliversiointi huomioitiin kuitenkin siten, että sen toteuttaminen myöhemmin on helppoa. Käytännössä tämä tarkoittaa sitä, että sivupohjissa käytetyt käyttöliittämättekstit on toteutettu kieliavaimilla, jotka sivua koottaessa korvataan vastaavalla tekstillä. Uuden kieliversion toteuttaminen edellyttää ainoastaan näiden käännöstekstien tuottamis-

ta. Suuremman haasteen aiheuttaakin tietosisällön kääntäminen eri kielille. Luonnollisesti uutta versiota toteutettaessa tulee huomioida muut kulttuurisidonnaiset seikat, kuten esimerkiksi käytetyt mittayksiköt.

Luku 7

Yhteenveto

Verkkopalvelun käyttäjän mieltymykset ja tavoitteet voivat merkittävästi erota muiden saman palvelun käyttäjien mieltymyksistä ja tavoitteista. Kaikkia palvelun käyttäjiä ei siis voida palvella yhtä hyvin näyttämällä kaikille samaa, staattista sisältöä. Verkkopalvelut vastaavat käyttäjäkuntansa monimuotoisuuteen mukauttamalla esittämäänsä sisältöä, avusteita ja toimintoja käyttäjästä luotavan profiilin perusteella.

Kuten kaikki automaattinen tietojenkäsittely, myös mukautusprosessi edellyttää, että tietokoneet ymmärtävät käsittelemästään datasta myös sen merkityksen. Tietojärjestelmissä tämä saavutetaan tyypillisimmin tuottamalla tietosisältö rakenteisessa muodossa ja liittämällä siihen metadataa. Verkkopalvelun tietomallia suunniteltaessa tulee varmistua, että se kykenee varastoimaan kaiken tarpeellisen tiedon ja metatiedon sellaisessa muodossa, että kaikki palvelun toiminnallisuuksien edellyttämät tietosisältöä koskevat toimenpiteet voidaan suorittaa.

Tietosisällön tuottaminen ja ylläpito on sitä työläämpää, mitä rakenteisemmassa muodossa tietosisältö halutaan kerätä. Koska mukautuvan verkkopalvelun tapauksessa tietosisällön rakenteisuus on välttämätöntä, ei tätä työvaihetta voida poistaa. Sen sijaan sitä voidaan tukea ja helpottaa erilaisilla apukeinoilla.

Tämä työ kirjoitettiin osana hanketta, jossa toteutettiin mukautuva verkkopalvelu erityisruokavaliota noudattaville ihmisille. Keskeisenä osana palvelua ovat näkyvät, joiden kautta käyttäjät voivat ylläpitää järjestelmän tietosisältöä välittämättä käyttöliittymän alla piilevistä teknisistä ratkaisuista. Näissä näkymissä hyödynnetään erilaisia sisällönhallintaa tukevia apukeinoja.

Sisällönhallinnasta vastaavien käyttäjien kognitiivista kuormaa voidaan keventää vaiheistamalla laajan tietosisältökokonaisuuden kerääminen useampaan vaiheeseen. Olennaista on, että tiedot kerätään käyttäjän käsitelmän mukaisissa loogisissa kokonaisuuksissa ja loogisessa järjestyksessä. Ulkoamuistamisen tarvetta voidaan

vähentää näyttämällä käyttäjälle yhteenvedot edellisissä vaiheissa tehdyistä merkinnöistä.

Kysyttäessä tietoa, jonka arvovaihtoehdot tiedetään, voidaan ulkoa muistamisen tarvetta vähentää tarjoamalla valmiit vaihtoehdot, joista käyttäjä voi valita sopivan. Pitkät vaihtoehtolistat voidaan korvata suosittelija-toiminnolla, joka päivittää tarjottavien vaihtoehtojen listaa käyttäjän syötteen perusteella.

Tiedonkeruulomaketta voidaan esitää aiemmin kerättyjen tietojen perusteella tai tarjota käyttäjälle pikavalintoja, joiden valinta automaattisesti täyttää määrätyt kentät määrätyillä arvoilla. Käyttäjä voi halutessaan tämän jälkeen muokata asetettuja arvoja sopivammiksi. Samankaltaisena toistuvien työvaiheiden aiheuttamaa työtä voidaan vähentää tarjoamalla mahdollisuus käyttää jo luotuja tietosisältökokonaisuuksia uuden sisällön pohjana, jolloin käyttäjän täytyy tehdä vain tarvittavat muutokset, eikä aloittaa koko prosessia alusta asti.

Virheiden syntymistä voidaan vähentää esittämällä tietosisältökokonaisuus ennen julkaisua esikatselunäkymässä sellaisessa muodossa, jossa se tullaan julkaisun jälkeen näyttämään. Virheiden korjaaminen voidaan suorittaa nopeasti muokkamalla vain sitä tietosisällön osaa, jossa virhe on, jolloin käyttäjän ei tarvitse käydä koko prosessia läpi uudelleen.

Liittämällä tietosisältöön julkaisutilaa koskevaa metatietoa voidaan valtuuttaa laaja käyttäjäkunta tuottamaan sisältöä, mutta silti keskitetysti hallita millaista sisältöä loppukäyttäjille näytetään. Poistamisen sijaan jo julkaistu sisältö voidaan väliaikaisesti piilottaa loppukäyttäjiltä, jolloin kertaalleen tuotettua sisältöä ei tarvitse myöhemmin tuottaa uudestaan.

Nykyaikaisen laajan verkkopalvelun rakenteisen tietosisällön tuottaminen on teknisesti monimutkainen ja työläs tehtävä. Elävä verkkopalvelu edellyttää, että sen sisältöä tuottamisen lisäksi myös ylläpidetään, eli kyseessä on jatkuva prosessi. Tuottamalla verkkopalveluun käyttöliittymä, joka piilottaa teknisen toteutuksen loppukäyttäjiltä ja tukee heitä sisällön ylläpitämisessä, voidaan sisällöntuotanto hajauttaa usealle käyttäjälle, jotka voivat keskittyä parhaan mahdollisen sisällön tuottamiseen teknisistä rajoitteista välittämättä.

Lähteet

- [Berners-Lee 1996] Berners-Lee, T. 1996. *WWW: Past, Present, and Future*. Computer, Volume 29, Issue 10 (Oct. 1996), pp. 69-77. IEEE Computer Society Press. ISSN: 0018-9162.
- [Boyle & Encarnacion 1994] Boyle, C. & Encarnacion, A.O. 1994. *MetaDoc: An Adaptive Hypertext Reading System*. User Modeling and User-Adapted Interaction 4, 1 (Jan. 1994), pp. 1-19.
- [Brusilovsky 1996] Brusilovsky, P. 1996. *Methods and Techniques of Adaptive Hypermedia*. User Modeling and User-Adapted Interaction 1996, volume 6, pp. 87-129.
- [Brusilovsky & Pesin 1998] Brusilovsky, P. & Pesin, L. 1998. *Adaptive Navigation Support in Educational Hypermedia: An Evaluation of the ISIS-Tutor*. Journal of Computing and Information Technology 6, 1 (1998), pp. 27-38.
- [Brusilovsky 2001] Brusilovsky, P. 2001. *Adaptive Hypermedia*. User Modeling and User-Adapted Interaction 2001, volume 11, pp. 87-110.
- [Brusilovsky & Maybury 2002] Brusilovsky, P. & Maybury, M.T. 2002. *From Adaptive Hypermedia to the Adaptive Web*. Communications of the ACM, May 2002. Vol. 45, No. 5.
- [Burrell, Treadwell & Gay 2000] Burrell, J., Treadwell, P. & Gay, G.K. 2000. *Designing for Context: Usability in a Ubiquitous Environment*. roceedings on the ACM conference on Universal Usability, 2000. pp. 80-84. ISBN: 1-58113-314-6
- [Bush 1945] Bush, V. 1945. *As We May Think*. The Atlantic Monthly, July 1945. Saatavilla: <http://www.theatlantic.com/doc/194507/bush> [Tarkistettu 17.6.2007]
- [Chawner & Lewis 2006] Chawner, B. & Lewis, P.H. 2006. *WikiWikiWebs: New Ways of Interacting in a Web Environment*. Information technology and libra-

- ries, 2006, vol. 25, no1, pp. 33-43. American Library Association, Chicago, IL. ISSN: 0730-9295.
- [Chiu 2001] Chiu, W. 2001. *Web site personalization*. IBM High Volume Web Sites, Software Group (AIM Division). Saatavilla: <http://www-128.ibm.com/developerworks/websphere/library/techarticles/hipods/personalize.html> [Tarkistettu 9.8.2007]
- [Cicconi 1999] Cicconi, S. 1999. *Hypertextuality*. Mediapolis. Berlino & New York: De Gruyter., 1999, pp. 21-43. Saatavilla: <http://www.cisnet.com/cisnet/writing/essays/hypertextuality.htm> [Tarkistettu 17.6.2007]
- [Dublin Core 2007] Dublin Core Metadata Initiative 2007. <http://dublincore.org/>
- [Di Iorio & Vitali 2005] Di Iorio, A. & Vitali, F. 2005. *Web authoring: a closed case?*. Proceedings of the 38th Hawaii International Conference on System Sciences, January 2005.
- [ECMA 1999] European Computer Manufacturers Association. 1999. *Standard ECMA-262: ECMAScript Language Specification, 3rd edition*. Saatavilla: <http://www.ecma-international.org/publications/standards/Ecma-262.htm> [Tarkistettu 24.3.2008]
- [Ebert 2006] Ebert, C. 2006. *Using wikis in software development*. IEEE Software, Vol. 26, No. 2, March-April 2006. ISSN 0740-7459.
- [Ebersbach & Glaser 2004] Ebersbach, A. & Glaser, M. 2004. *Towards Emancipatory Use of a Medium: The Wiki*. International Journal of Information Ethics, November 2004. ISSN 1614-1687.
- [FormatData 2000] FormatData 2002. *RecipeML DTD Version 0.5*. Saatavilla: <http://www.formatdata.com/recipeml/spec/recipeml-spec.html> [Tarkistettu 15.11.2007]
- [FormatData 2002] FormatData 2002. *RecipeML Working Draft 20 April 2002*. Saatavilla: <http://www.formatdata.com/recipeml/recipeml.dtd> [Tarkistettu 9.2.2008]
- [Goodwin & Vidgen 2002] Goodwin, S. & Vidgen, R. 2002. *Content, content, everywhere... time to stop and think? The process of Web content management*. Computing & Control Engineering Journal, Vol 13, No. 2, pp. 66-70. ISSN: 0956-3385.

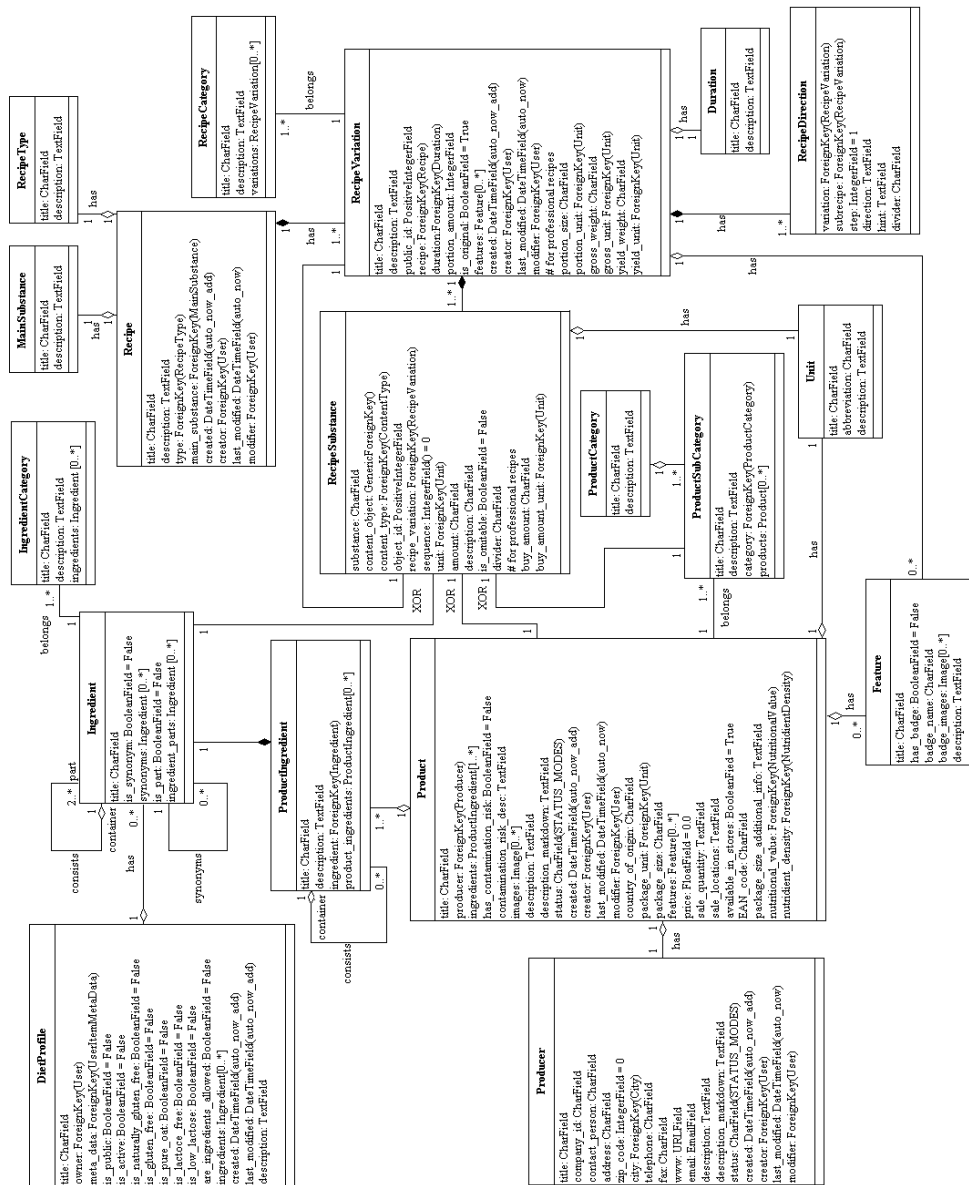
- [Haarmann 2001] Haarmann, P-L. 2001. *Immateriaalioikeuksien oppikirja*. 3. painos. Helsinki: Kauppakaari, Lakimiesliiton Kustannus.
- [Hirsh, Basu & Davison 2000] Hirsh, H., Basu, C. & Davison, D. 2000. *Learning to Personalize*. Communications of the ACM, August 2000. Vol. 43, No. 8.
- [Joerding 1999] Joerding, T. 1999. *A Temporary User Modeling Approach for Adaptive Shopping on the Web*. Proceedings of the 2nd Workshop on Adaptive Systems and User Modeling on the WWW. Saatavilla: <http://wwwis.win.tue.nl/asum99/joerding/joerding.html> [Tarkistettu 7.8.2007]
- [Kansanterveyslaitos 2008] *Fineli-koostumustietokanta*. Kansanterveyslaitoksen ravitsemusyksikön ylläpitämä elintarvikkeiden koostumustietopankki. Saatavilla: <http://www.fineli.fi/foodclass.php?classif=igclass> [Tarkistettu 26.3.2008]
- [Kaplan, Fenwick & Chen 1993] Kaplan, C., Fenwick, J. & Chen, J. 1993. *Adaptive Hypertext Navigation Based on User Goals and Context*. User Modeling and User-Adapted Interaction 3, 3 (1993), pp. 193-220.
- [Kilpi 2006] Kilpi, T. 2006. *Blogit ja bloggaaminen*. Gummerus Kirjapaino Oy, Jyväskylä, 2006. ISBN: 952-5655-03-2.
- [Kobsa, Koenemann & Pohl 2001] Kobsa, A., Koenemann, J. & Pohl, W. 2001. *Personalized Hypermedia Presentation Techniques for Improving Online Customer Relationships*. The Knowledge Engineering Review 16, 2001, pp. 111-155. Cambridge University Press.
- [Leuf & Cunningham 2001] Leuf, B. & Cunningham, W. 2001. *The Wiki way: quick collaboration on the Web*. Boston: Addison-Wesley.
- [Lindahl & Blount 2003] Lindahl, C. & Blount, E. 2003. *Weblogs: Simplifying Web Publishing*. Computer, vol. 36, No. 11, pp. 114-116. IEEE Computer Society. ISSN: 0018-9162.
- [Lowe & Hall 1999] Lowe, D. & Hall, W. 1999. *Hypermedia & the Web: An Engineering Approach*. John Wiley & Sons Ltd, Chichester, UK. ISBN: 0-471-98312-8.
- [McKeever 2003] McKeever, S. 2003. *Understanding Web content management systems: evolution, lifecycle and market*. Industrial Management & Data Systems, vol. 103, No. 9, pp. 686-692. Julkaisija MCB UP Ltd. ISSN: 0263-5577.
- [Mielonen & Hintikka 1998] Mielonen, S. & Hintikka, K.A. 1998. *Web-palveluiden käytettävyyys ja tuotanto*. Saatavilla: <http://www2.uiah.fi/mediastudio/survey4/> [Tarkistettu 7.12.2007]

- [Moody & Shanks 2003] Moody, D.L. & Shanks, G.G. 2003. *Improving the quality of data models: empirical validation of a quality management framework*. Information Systems 28 (2003), Elsevier.
- [Nelson 1965] Nelson, T. 1965. *Complex Information Processing: A File Structure for The Complex, The Changing and The Indeterminate*. ACM/CSC-ER, proceedings of the 1965 20th national conference, pp. 84-100.
- [Nielsen 1994] Nielsen, J. 1994. *Heuristics for User Interface Design*. Saatavilla: http://www.useit.com/papers/heuristic/heuristic_list.html [Tarkistettu 7.8.2007]
- [Nielsen 1999] Nielsen, J. 1999. *User Interface Directions for the Web*. Communications of the ACM. January 1999, vol. 42, No. 1, pp. 65-72.
- [Nielsen 2003] Nielsen, J. 2003. *Usability 101: Introduction to Usability*. Saatavilla: <http://www.useit.com/alertbox/20030825.html> [Tarkistettu 7.12.2007]
- [Nieminen 2002] Nieminen, S. 2002. *Adaptiivinen Hypermedia*. Hypermediajärjestelmät-seminaari, Helsingin Yliopisto. Saatavilla: <http://www.cs.helsinki.fi/u/erkio/hmsem02/pdf/nieminen.pdf> [Tarkistettu 18.6.2007]
- [Nunamaker 2001] Nunamaker Jr., J.F., Romano Jr., N.C. & Briggs, R.O. 2001. *A Framework for Collaboration and Knowledge Management*. Proceedings of the 34th Hawaii International Conference on System Sciences, IEEE. ISBN: 0-7695-0981-9.
- [Salminen 2005] Salminen, A. 2005. *Metatieto organisaatioiden sisällönhallinnassa*. Ilmestynyt julkaisussa Lehtinen, A., Salminen, A. & Nurmeksela, R. Metatieto suomalaisen lainsäädäntöprosessin tiedonhallinnassa. RASKE2-projektin II väliraportti (s. 4-13). Eduskunnan kanslian julkaisu 7/2005. Helsinki: Eduskunnan kanslia.
- [Tekijänoikeus] Opetusministeriö. 2008. *Tekijänoikeus*. Saatavilla: <http://www.minedu.fi/OPM/Tekijaenoikeus/> [Tarkistettu 7.4.2008]
- [Tekijänoikeuslaki] *Tekijänoikeuslaki* 8.7.1961/404. Saatavilla: <http://www.finlex.fi/fi/laki/ajantasa/1961/19610404> [Tarkistettu 7.4.2008]
- [Tervakari & Silius 2007] Tervakari, A. & Silius, K. 2007. *Verkkopalvelun sisällöntuotanto, luentomateriaali opintojakson syksyn 2006 suorituskerralle, osa 7*. TTY/Hypermedialaboratorio. Saatavilla:

- http://matriisi.ee.tut.fi/hmopetus/vpsist/2006/liitteet/7luento_vepsto2006.pdf
[Tarkistettu 21.6.2007]
- [W3C 2006] World Wide Web Consortium. 2006. *Extensible Markup Language 1.0 (Fourth Edition)*. Saatavilla: <http://www.w3.org/TR/2006/REC-xml-20060816/> [Tarkistettu 15.11.2007]
- [W3C 2007] World Wide Web Consortium. 2007. *Cascading Style Sheets Level 2 Revision 1 (CSS 2.1) Specification*. Saatavilla: <http://www.w3.org/TR/CSS21/> [Tarkistettu 14.4.2008]
- [Wang et al. 2006] Wang, L., Li, Q., Li, Y. & Meng, X. 2006 *Dish_Master: an Intelligent and Adaptive Manager for a Web-based Recipe Database System*. Proceedings of the Second International Conference on Semantics, Knowledge and Grid (SKG'06)
- [Wang & Li 2007] Wang, L. & Li, Q. 2007. *A Personalized Recipe Database System with User-Centered Adaptation and Tutoring Support*. Proceedings of SIGMOD2007 Ph.D. Workshop on Innovative Database Research 2007(IDAR2007), June 10, 2007, Beijing, China.
- [Wilde 2007] Wilde, E. 2007. *Declarative Web 2.0*. IEEE International Conference on Information Reuse and Integration 2007 (IRI2007), August 13-15, 2007, Las Vegas, USA. ISBN: 1-4244-1500-4.

Liite A

Järjestelmän tietomalli



Järjestelmän karsittu tietomalli UML-kaaviona