

VAASAN YLIOPISTO

TEKNILLINEN TIEDEKUNTA

OHJELMISTOTEKNIikka

Marianne Sjöberg

**AKTIVOIVAA OPETUSTA TUKEVA VERKKO-OPPIMISYMPÄRISTÖ
KURSSIMUOTOISILLE OPPILAITOKSILLE**

Diplomityö, joka on jätetty tarkastettavaksi diplomi-insinöörin tutkintoa varten
Vaasassa 12.9.2016

Työn valvoja

Jouni Lampinen

Työn ohjaaja

Jouni Lampinen

ALKULAUSE

Tämän diplomityön aihe, eli aktivoivaa opetusta tukevan verkko-oppimisympäristön kehittäminen, lähti liikkeelle todellisesta, edellisessä työssäni havaitsemastani tarpeesta. Olen aiemmalta koulutukseltani matematiikan ja fysiikan aineenopettaja, ja toiminut tässä ammatissa yhteensä kuusi vuotta sekä peruskoulussa että lukiossa. Viimeisinä opettajavuosina havahduin siihen, että ns. perinteisellä opetusmenetelmällä pitämäni oppitunnit rakentuivat pääosin luokan edessä pitämäni monologiin, jota opiskelijat joutuivat passiivisina seuraamaan. Tämä johti vääjäämättä siihen, etten ollut lainkaan tietoinen siitä, miten hyvin tai huonosti opiskelijat omaksuivat opettamani asiat. Pohdin samalla myös tämän opetusmenetelmän vaikutusta opiskelijoiden opiskelumotivaatioon. Ryhdyin etsimään keinoja asian muuttamiseksi ja huomasin, kuinka pienimuotoinenkin tietotekniikan mukaanotto opetukseen, ja sen avulla opiskelijoiden aktivointi nosti selvästi opiskelijoiden opiskelumotivaatiota ainetta kohtaan. Saatavilla ei kuitenkaan ollut riittäviä työkaluja kaikkiin opetukseni tarpeisiin, eikä minulla ollut osaamista sellaisten luomiseen. Tämä oli käynnistävä tekijä sille, että päätin lähteä jatkokouluttautumaan ja opiskelemaan ohjelmointia.

Aloitin ohjelmistotekniikan diplomi-insinööriopinnot Vaasan yliopistossa syksyllä 2014 oikeastaan ilman minkäänlaista pohjatietoa alaan liittyvistä aiheista. Ensimmäinen opiskeluvuosi oli hyvin raskas, sillä kursseilla opeteltaviin asioihin perehtyminen vaati hyvin paljon omatoimista lisätyötä ja toi mukanaan lukuisia epätoivon hetkiä. Vaasan yliopiston teknillinen tiedekunta on pieni yliopistoyksikkö, mutta ehkä juuri tästä syystä yleishenki siellä on lämmin ja ystävällinen. Opettajat kohtaavat opiskelijat yksilöinä eivätkä vain numerosarjoista koostuvana massana. Ainakin itse tunsin saavani henkilökohtaista opetusta aina tarpeen sitä vaatiessa, ja ilman tätä opiskelutaipaleeni olisi ollut entistä kivisempi. Erityiskiitokset haluan esittää Laura Lappalaiselle ja Hannu Niinimäelle. Kiitos laadukkaasta opetuksesta ja siitä, että teillä oli kiireidenkin keskellä aikaa antaa ylimääräistä, mutta aina ystävällistä lisäopetusta.

Opiskeluaika Vaasan yliopistossa oli kaikin puolin mukavaa ja kehityksen seurauksena erittäin palkitsevaa. Olkoon tämä diplomityö, ja siinä esittelemäni, itse alusta kehittämäni aktivoivaa opetusta tukeva verkko-oppimisympäristö, eräänlainen esimerkki periksiantamattoman työnteon tuloksesta.

Lopuksi haluan kiittää perhettäni. Äiti ja Markus, olette rakkaimmat ja tärkeimmät ihmiset minulle koko maailmassa.

Helsingissä 12.9.2016

Marianne Sjöberg

SISÄLLYSLUETTELO

LYHENTEET	5
TIIVISTELMÄ	6
ABSTRACT	7
1 JOHDANTO	8
1.1 Työn tausta	8
1.2 Työn tavoite, toteutus ja rajaus	9
1.3 Työn rakenne	10
2 OHJELMISTOKEHITYS VESIPUTOUSMALLIA MUKAILLEN	12
2.1 Määrittelyvaihe	14
2.2 Suunnitteluvaihe	16
2.2.1 Arkkitehtuurisuunnittelu	17
2.2.2 Tietokantasuunnittelu	19
2.2.3 Komponenttisuunnittelu	23
2.2.4 Käyttöliittymäsuunnittelu	23
2.3 Toteutusvaihe	26
2.4 Mallinnus ohjelmistokehityksessä	28
3 VERKKO-OPPIMISYMPÄRISTÖN KEHITYKSESSÄ HUOMIOITAVIA TEKIJÖITÄ	33
3.1 Verkko-oppimisympäristön rakenteen ja toiminnallisuuksien suunnittelu	33
3.2 Verkko-oppimisympäristön käyttöliittymä	35
3.3 Verkko-oppimisympäristöön liittyvä tekniikka	36
4 AKTIVOIVAA OPETUSTA TUKEVAN VERKKO-OPPIMISYMPÄRISTÖN VAATIMUKSET	39
4.1 Toiminnalliset vaatimukset	39
4.1.1 Etusivun toiminnot	40
4.1.2 Pääsivuston toiminnot	44

4.1.3	Kurssisivuston toiminnot	55
4.1.4	Viestiliikennetoiminnot	67
4.2	Ei-toiminnalliset vaatimukset	68
4.2.1	Käytettävyystvaatimukset	68
4.2.2	Kehitysvaatimukset	68
4.2.3	Turvallisuusvaatimukset	68
4.2.4	Käyttövarmuus- ja saatavuusvaatimukset	69
5	AKTIVOIVAA OPETUSTA TUKEVAN VERKKO-OPPIMISYMPÄRISTÖN SUUNNITTELUVAIHE	70
5.1	Arkkitehtuurisuunnittelu	70
5.2	Tietokantasuunnittelu	76
5.3	Komponenttisuunnittelu	77
5.4	Käyttöliittymäsuunnittelu	78
6	AKTIVOIVAA OPETUSTA TUKEVAN VERKKO-OPPIMISYMPÄRISTÖN TOTEUTUS	83
6.1	Käyttöliittymä	83
6.2	Komponentit	84
6.2.1	Etusivun komponentit	85
6.2.2	Pääsivuston komponentit	91
6.2.3	Kurssisivuston komponentit	100
7	TULOKSET	114
8	JOHTOPÄÄTÖKSET	115
8.1	Työn luotettavuuden arviointi	116
8.2	Jatkotutkimusaiheita	117
	LÄHTEET	118
	LIITTEET	

LYHENTEET

CSS	<i>Cascading Style Sheets</i> , verkkosivuston ulkoasun muotoilussa hyödynnettävä tyylikieli.
ER-malli	<i>Entity-Relationship model</i> , tietokannan rakenteen kuvaamiseen käytettävä malli.
HTML	<i>Hypertext Markup Language</i> , verkkosivujen esittämiseen kehitetty kuvauskieli.
HTTP	<i>Hypertext Transfer Protocol</i> , Internetissä tiedonsiirrossa käytettävä yhteyskäytäntö eli protokolla.
JavaScript	Verkkosivujen kehittämiseen soveltuva ohjelmointikieli.
jQuery	JavaScript-ohjelmia sisältävä kirjasto.
PHP	<i>PHP: Hypertext Preprocessor</i> , verkkosivujen luomiseen soveltuva ohjelmointikieli.
SQL	<i>Structured Query Language</i> , relaatiotietokantojen kyselykieli.
UML	<i>Unified Modeling Language</i> , graafinen mallinnuskieli, jota voidaan hyödyntää ohjelmistokehityksessä.
URL	<i>Uniform Resource Locator</i> , selaimelle kirjoitettava osoite, jolla voidaan avata verkkosivuja.

VAASAN YLIOPISTO
Teknillinen tiedekunta

Tekijä:	Marianne Sjöberg
Diplomityön nimi:	Aktivoivaa opetusta tukeva verkko-oppimis-ympäristö kurssimuotoisille oppilaitoksille
Valvoja:	Professori Jouni Lampinen
Ohjaaja:	Professori Jouni Lampinen
Tutkinto:	Diplomi-insinööri
Koulutusohjelma:	Tietotekniikan ko (DI)
Suunta:	Ohjelmistotekniikka
Opintojen aloitusvuosi:	2014
Diplomityön valmistumisvuosi:	2016

Sivumäärä: 122

TIIVISTELMÄ

Nykyaikainen oppimiskäsitys näkee oppimisen tiedon aktiivisena tuottamisena. Vastuu oppimisesta siirretään opiskelijalle, ja opettaja toimii enemmän yhteistyökumppanina kuin vain tiedon yksipuolisena jakajana. Tässä yhteydessä puhutaan usein niin sanotusta aktivoivasta opetuksesta. Lisäksi nykyaikaisessa koulumaailmassa tieto- ja viestintätekniikan rooli vahvistuu entisestään. Käytettävissä olevan teknologian avulla pyritään tarjoamaan uusia ja kiinnostavia opetukseen ja oppimiseen liittyviä kokemuksia ja ympäristöjä. Tämän diplomityön tutkimustehtävä, ja näin ollen työn tavoite, on kehittää aktivoivaa opetusta tukeva verkko-oppimisympäristö, joka suunnataan kurssimuotoisille oppilaitoksille. Tutkimustehtävä suoritetaan päätöksentekometodologisella tutkimusotteella, joka perustuu uuden konstruktion luomiseen deduktiivisesti analyttisellä mallintamisella.

Verkko-oppimisympäristö kehitetään, eli uusi konstruktio luodaan, ohjelmistokehityksessä käytettävää vesiputousmallia mukailten suorittamalla vesiputousmallin sisältämät määrittely-, suunnittelu- ja toteutusvaiheet. Mallin sisältämät testausvaiheet rajataan työn ulkopuolelle, mikä on perusteltavissa valitulla tutkimusotteella, jossa empirialla ei ole keskeistä osaa. Verkkoympäristön kehitystyö pohjustetaan kirjallisuuskatsauksella, jossa perehdytään yleisiin verkko-oppimisympäristön kehityksessä huomioitaviin tekijöihin painottaen niitä tekijöitä, joilla verkko-oppimisympäristö voidaan suunnata tukemaan aktivoivaa opetusta.

Työn tulos on uusi konstruktio, kurssimuotoisille oppilaitoksille suunnattu, aktivoivaa opetusta tukeva verkko-oppimisympäristö, jolle annetaan nimeksi Cuulis-oppimisympäristö. Ominaisuuksiensa johdosta Cuulis-oppimisympäristöstä voi olla suurta käytännön hyötyä nykyaikaisille digitalisoituville oppilaitoksille. Cuulis-oppimisympäristö pilotoidaan Espoon Tapiolan lukiossa eri vuositasojen kemian kursseilla. Sekä pilotoinnin yhteydessä saatujen käyttökokemusten että muiden uusien ideoiden perusteella Cuulis-oppimisympäristöä on tarkoitus myöhemmin kehittää. Suositeltavia jatkotutkimusaiheita ovat muun muassa Cuulis-oppimisympäristön systemaattiset ohjelmisto- ja käytettävyydestestaukset.

AVAINSANAT: aktivoiva opetus, verkko-oppimisympäristö, ohjelmistokehitys

UNIVERSITY OF VAASA**Faculty of technology**

Author:	Marianne Sjöberg
Topic of the Thesis:	Online-learning environment supporting activating teaching for course-based educational institutions
Supervisor:	Professor Jouni Lampinen
Instructor:	Professor Jouni Lampinen
Degree:	Master of Science (Technology)
Degree Programme:	Information Technology
Major:	Software Engineering
Year of Entering the University:	2014
Year of Completing the Thesis:	2016

Pages: 122

ABSTRACT

According to the contemporary approach learning is seen as a process of actively producing information. The responsibility of learning has been transferred to students while teachers are increasingly acting as partners rather than one-sided sources of information. The term 'active learning' is often referred to in this context. Furthermore, the role of information and communication technology is increasingly growing in the education field. The technology available is being used for providing new and interesting experiences and environments related to teaching and learning. The research task and thus the goal of this thesis is to develop an online-learning environment which supports activating teaching and which is designed to be used in course based educational institutions. The research task will be implemented according to a decision-making oriented research method based on deductively creating a new construction using analytical modeling.

The online-learning environment is developed, i.e. the construction is created, by executing the definition, planning and implementation phases included in the waterfall model used in software development. The testing phases included in the model are excluded from this thesis, which is justified by the chosen research method where empirical research does not have an important role. The development of this online-learning environment is based on a literature review of the basic elements of developing an online-learning environment, with the emphasis on the elements with which online-learning environment can be oriented towards supporting activating teaching.

The outcome of the work is a new construction – an online-learning environment supporting activating teaching for course-based educational institutions. The name of this environment is 'Cuulis learning environment'. Because of its features the Cuulis learning environment is potentially highly useful for contemporary, digitalizing educational institutions. The Cuulis learning environment will be tried out first in Tapiola Upper Secondary School, Espoo, in chemistry courses of different year levels. It is planned to develop the Cuulis learning environment further based on user experiences from the trying out phase as well as on new ideas. Recommended subject for further research is for example the systematic software and usability testing of the Cuulis learning environment.

KEYWORDS: activating teaching, online-learning environment, software development

1 JOHDANTO

1.1 Työn tausta

Perinteiset opetusmenetelmät kehitettiin aikana, jolloin tutkimuksessakin keskityttiin oppimistulosten mittaamiseen sen sijaan, että olisi tutkittu sitä, miten asioita opitaan. Tutkimuksessa pyrittiin kehittämään erilaisia testejä ja mittareita, joiden avulla opiskelijoita ja oppimistuloksia luokiteltiin eritasoisiksi. Myöhemmin (1980-luvun aikana) kun havaittiin, että tällaisilla tutkimustuloksilla ei kyetty selvittämään oppimisen perusmekanismeja, alettiin keskittyä enemmän itse oppimisprosessin tutkimiseen. Tämä alkoi vähitellen heijastua myös opetuskäytäntöihin siten, että oppimisen lopputuloksiin keskittyvän, ns. tuotospainotteisen opetuksen sijaan on alettu puhua nimenomaan oppimistapahtumaan keskittyvästä, ns. prosessipainotteisesta opetuksesta. Tietotekniikan kehitys ja siitä seurannut tiedon tulva on asettanut haasteita koulumaailmalle sekä lisännyt entisestään kiinnostusta prosessipainotteista opetusta kohtaan. (Lonka & Lonka 1991: 7.)

Aktivoiva opetus määritellään opetuksiksi, jossa oppiminen on keskeinen tavoite. Aktivoivassa opetuksessa vastuuta oppimisesta siirretään opiskelijalle, ja opettaja toimii enemmän yhteistyökumppanina ja työn ohjaajana kuin tiedon jakajana. Aktivoivat opetusmenetelmät perustuvat sekä prosessipainotteiseen ajatteluun että tiedon rakentelua painottavaan oppimiskäsitykseen, jossa oppimisen ajatellaan olevan tiedon aktiivista tuottamista. Todellista oppimista ei saavuteta asioiden muistiin tallentamisella, vaan opiskelijan aktiivisella panoksella, kun hän yrittää rakentaa oppimastaan merkityksellisiä kokonaisuuksia. (Lonka & Lonka 1991: 12.)

Tieto- ja viestintätekniiikan avulla voidaan tarjota erilaisia uusia ja kiinnostavia sekä opetukseen että oppimiseen liittyviä kokemuksia ja ympäristöjä (Nam & Smith-Jackson 2007: 23). Yksi vuoden 2015 eduskuntavaalien jälkeen muodostetun hallituksen (ns. Juha Sipilän hallitus) hallitusohjelman tavoitteista on digitalisoida suomalaista yhteiskuntaa.

Myös opetusta digitalisoidaan. Eräs opetukselle ja koulutukselle suunnattu tavoite on oppimisympäristöjen modernisointi, sekä digitalisaation ja uuden pedagogiikan mahdollisuuksien hyödyntäminen oppimisessa. (Valtioneuvoston kanslia 2015a: 17-26.) Yhdeksi kärkihankkeeksi asetettiin uusien oppimisympäristöjen ja digitaalisten materiaalien käyttöönotto kaikissa Suomen peruskouluissa. Kärkihankkeen tavoitteena on parantaa oppimistuloksia, vastata tulevaisuuden osaamistarpeisiin, uudistaa pedagogiikkaa ja tehdä oppimisesta innostavaa läpi elämän. (Valtioneuvoston kanslia 2015b: 26.) Opetuksen digitalisaatio koskee kuitenkin kaikkia oppiasteita esiopetuksesta korkeakouluihin.

Jo ennen Sipilän hallitusta monipuolisten oppimisympäristöjen ja teknologian tarve on tiedostettu suomalaisessa koulumaailmassa. Yksi uuden perusopetuksen opetussuunnitelman perusteiden (OPS 2016) tavoitteista on syventää oppimiskäsitystä sekä vahvistaa edellytyksiä tietoa luovaan, yhteisölliseen ja oppilaiden tarpeet huomioon ottavaan oppimiseen monipuolisissa oppimisympäristöissä. Lukion opetussuunnitelman perusteiden 2015 päivittämisen yhtenä tavoitteena mainitaan monipuolisten opiskeluympäristöjen ja opetusteknologian käytön tukeminen. Molemmat uudet opetussuunnitelman perusteet on tarkoitus ottaa käyttöön viimeistään 1.8.2016. (Opetushallitus 2013 & 2015.)

1.2 Työn tavoite, toteutus ja rajaus

Tämän diplomityön tutkimustehtävä, ja näin ollen työn tavoite, on kehittää aktivoivaa opetusta tukeva, kurssimuotoisille oppilaitoksille suunnattu verkko-oppimisympäristö. Pantzar & Väliharju (1996: 25–26) määrittelevät *oppimisympäristön* tavoitteellisen opiskelun mahdollistamaksi kokonaisuudeksi, joka muodostuu fyysisistä ja henkisistä puitteista sekä oppimateriaaleista. Fyysisillä puitteilla tarkoitetaan muun muassa opetuksessa/oppimisessa tarvittavia tiloja ja laitteita. Yksi esimerkki henkisistä puitteista on opettajan tai muun oppimista edistävän henkilön osaaminen. *Verkko-oppimisympäristöllä* tarkoitetaan verkossa toimivaa, Internet-pohjaista oppimisympäristöä (Mäcklin 2012: 7).

Tutkimustehtävä suoritetaan päätöksentekometodologisella tutkimusotteella. Päätöksentekometodologinen tutkimusote on uusia konstruktioita tuottava deduktiivinen metodologia. Tutkimusotteen taustalla oleva tieteellinen ideaali mukailee logiikkaa ja matematiikkaa, ja tutkimustulokset, eli uudet konstruktiot, ovat ratkaisuja havaittuihin todellisiin ongelmiin. Päätöksentekometodologisissa tutkimuksissa uusi konstruktio luodaan siis teorian pohjalta analyttisellä mallinnuksella, eikä empirialla ole tässä keskeistä roolia. Empiriaa hyödynnetään korkeintaan sovellusesimerkkien muodossa, eikä luodun konstruktion testaaminen ole näin ollen välttämätöntä. (Neilimo & Näsi 1980: 66-67.) Päätöksentekometodologinen tutkimusote on valittu tähän diplomityöhön nimenomaan sen uuden, käytännön ongelmia ratkaisevan konstruktion luomiseen pyrkivän tavoitteellisuuden vuoksi, millä voidaan suorittaa työlle asetettu tutkimustehtävä. Tuleva uusi konstruktio on siis kurssimuotoisille oppilaitoksille suunnattu verkko-oppimisympäristö, joka tukee aktivoivaa opetusta.

Verkko-oppimisympäristön kehittäminen, eli uuden konstruktion luomisessa käytettävä analyttinen mallinnus, pohjustetaan tekemällä kirjallisuuskatsaus, jossa perehdytään sekä yleisiin verkko-oppimisympäristön kehityksessä huomioitaviin tekijöihin että niihin tekijöihin, joilla verkko-oppimisympäristöstä saadaan aktivoivaa opetusta tukeva. Verkko-oppimisympäristö kehitetään ohjelmistokehityksessä käytettävää ns. vesiputousmallia mukaillen suorittamalla mallin sisältämät määrittely-, suunnittelu- ja toteutusvaiheet. Testausvaiheet rajataan tämän työn ulkopuolelle, mikä on perusteltavissa valitulla tutkimusotteella.

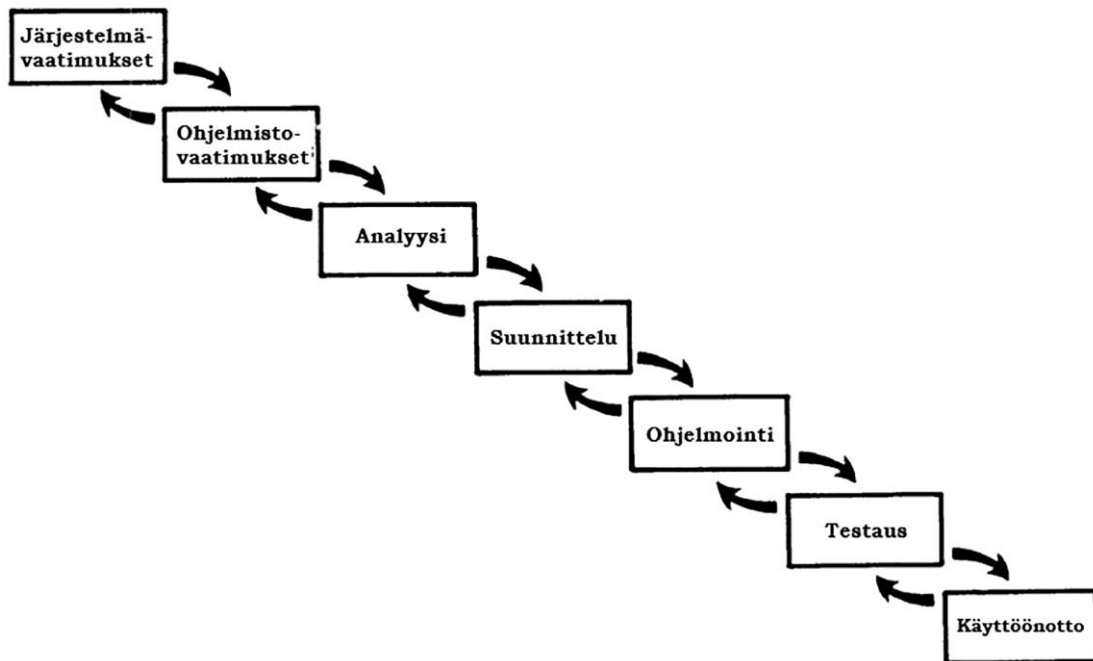
1.3 Työn rakenne

Diplomityö on jaettu johdannon lisäksi seitsemään päälukuun. Toisessa luvussa esitellään ohjelmistokehityksessä käytettävä vesiputousmalli ja sitä mukaileva, tässä työssä uuden verkko-oppimisympäristön kehittämisessä käytettävä versio. Kolmannessa luvussa käsitellään sekä yleisiä että erityisesti aktivoivaa opetusta tukevan verkko-oppimisympäristön kehityksessä huomioitavia tekijöitä. Neljännessä luvussa esitellään kehitettävän verkko-oppimisympäristön määrittelyvaiheessa muodostetut vaatimukset. Viidennessä luvussa

kuvataan verkko-oppimisympäristön kehityksen suunnitteluvaihe, ja kuudennessa luvussa sen toteutusvaihe. Seitsemännessä luvussa esitellään työn tulokset. Lopuksi kahdeksannessa luvussa esitellään johtopäätökset, jossa muun muassa pohditaan kehitetyn verkko-oppimisympäristön merkitystä, analysoidaan työn luotettavuutta ja annetaan jatkotutkimusideoita. Diplomityö on pyritty laatimaan siten, että se on ymmärrettävissä kaikille aktivoivasta opetuksesta ja verkko-oppimisympäristöistä kiinnostuneille lukijan teknologisesta tietämyksestä ja taustasta riippumatta.

2 OHJELMISTOKEHITYS VESIPUTOUSMALLIA MUKAILLEN

Pressman (2010: 4) määrittelee *ohjelmiston* kokonaisuudeksi, joka muodostuu 1) ohjeista (tietokoneohjelmista), jotka ajettaessa tarjoavat halutut ominaisuudet, toiminnot ja suorituskyvyn, 2) tietorakenteista, jotka käsittelevät informaatiota asiaankuuluvasti, sekä 3) ohjelmien toimintaa ja käyttöä kuvailevasta tiedosta. *Järjestelmällä* tarkoitetaan puolestaan ohjelmiston ja laitteiston muodostamaa kokonaisuutta (Haikala & Märijärvi 2004: 79). Seuraavassa ohjelmistokehitystä tarkastellaan nimenomaan ohjelmiston kehityksen kannalta.



Kuva 1. Yksi ensimmäisistä vesiputouksmallin versioista (Royce 2010: 330).

Ohjelmistoprosessi sisältää erilaisia toisiinsa liittyviä toimintoja, joiden suorittaminen tuottaa ohjelmiston. *Ohjelmistoprosessimalli* kuvaa yksinkertaistetulla tavalla ohjelmistoprosessin tietystä näkökulmasta (Sommerville 2011: 28-29). Winston W. Royce esitteli vuonna 1970 julkaisemassaan artikkelissaan ”Managing the Development of Large Soft-

ware Systems” ohjelmistoprosessimallin, jota on myöhemmin alettu kutsua ns. *vesiputousmalliksi*. Malli jakaa ohjelmistokehityksen lineaarisesti eri vaiheisiin siten, että edellinen vaihe suoritetaan aina loppuun ennen seuraavan aloittamista. Mallissa voidaan siirtyä myös taaksepäin, mikäli sellaiseen havaitaan tarvetta, mutta useimmiten kuitenkin vain korkeintaan yhden vaiheen verran. (Royce 1970: 328-330.) Roycen esittelemää mallia pidetään vesiputousmallin ensimmäisenä määrittelynä, mutta jo aiemmin sekä H. D. Beningtonin (vuonna 1956) että W.A. Hosierin (vuonna 1961) julkaisemat artikkelit käsitelivät ohjelmistoprosessimalleja, jotka muistuttivat paljon Roycen mallia (Boehm 1987: 296-297). Vesiputousmalli on joka tapauksessa yksi vanhimmista ohjelmistokehityksessä käytetyistä ohjelmistoprosessimalleista ja siitä on olemassa useita muunnelmia. Yhteistä vesiputousmallin eri versioille on yleensä se, että ne sisältävät ainakin vaatimusten muodostamisesta koostuvan määrittelyvaiheen, sekä suunnittelu- ja toteutusvaiheet (Haikala & Märijärvi 2004: 37). Kuvassa 1 on Roycen esittelemä vesiputousmallin versio.

Vesiputousmalliin liittyy lineaarisuutensa takia useita heikkouksia, minkä jo Royce (1970: 329) artikkelissaan totesi. Jos eri vaiheissa tehdyt virheet tai muut puutteet havaitaan vasta myöhemmissä vaiheissa, niin tarvittavien muutosten tekeminen myöhästyttää ohjelmiston valmistumista ja lisää kehityskustannuksia. Lisäksi ohjelmistosta saadaan julkaistu versio vasta mallin loppuvaiheessa, mikä saattaa huolestuttaa sekä ohjelmiston kehittäjiä että asiakasta, joka on tilannut ohjelmiston. (Royce 1970: 329.)

Heikkouksistaan huolimatta vesiputousmalli tuo esille ohjelmistokehitysprosessiin kuuluvat tärkeät vaiheet ja niiden suhteet toisiinsa (Munassar & Govardhan 2010: 96). Vesiputousmalli kuvastaa myös hyvin tavanomaista ongelmanratkaisuprosessia, joka koostuu ratkaistavan ongelman analysoinnista sekä ratkaisun suunnittelusta, toteutuksesta ja testauksesta (Haikala & Märijärvi 2004: 41). Mallia voidaan soveltaa käytännön ohjelmistokehityksessä muun muassa siten, että eri vaiheita suoritetaan osittain päällekkäin ja vaiheiden yhteydessä valmistetaan erilaisia prototyyppisiä, joiden avulla voidaan tehdä testauksia, katselmointeja tms. (Munassar & Govardhan 2010: 97).

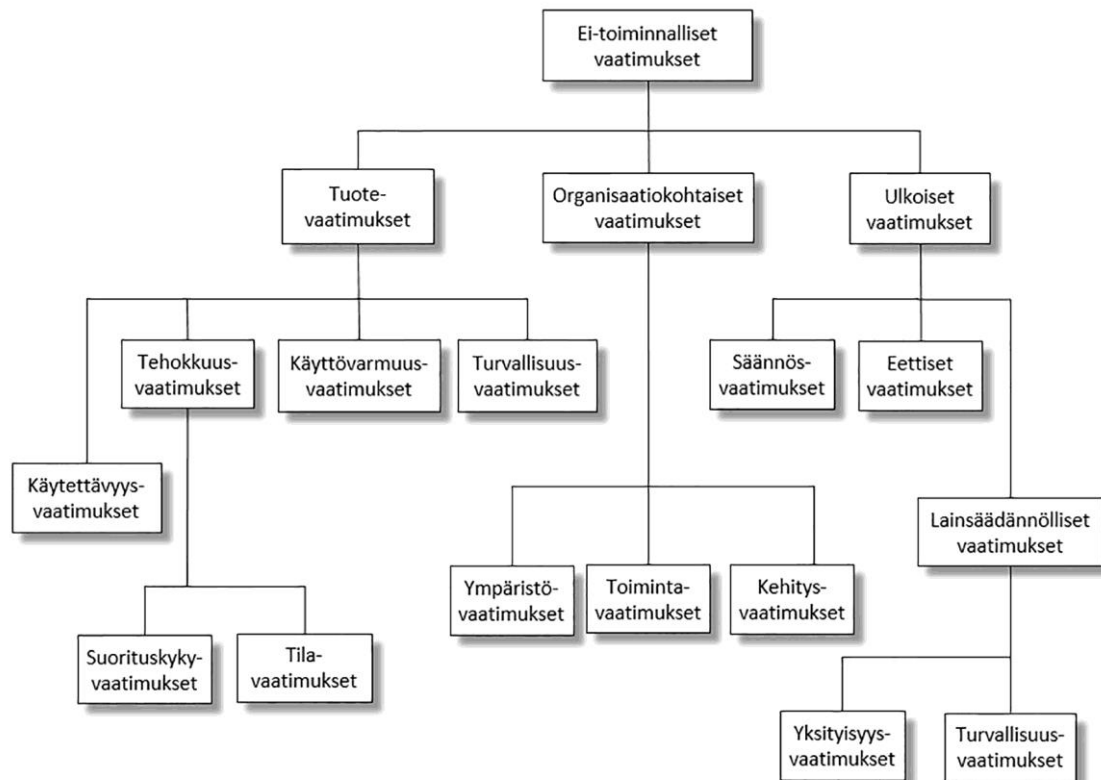
Ohjelmistoprosessimallit ovat usein vain yleisiä kuvauksia ohjelmistoprosessin sisältämistä vaiheista ja tehtävistä. Käytännössä näitä malleja voidaan laajentaa ja mukauttaa kuhunkin ohjelmistokehitysprosessiin sopivaksi. (Sommerville 2011: 28–29.) Tässä diplomityössä kehitettävän verkko-oppimisympäristön kehitys suoritetaan vesiputousmallia mukaillen siten, että mallista suoritetaan määrittely-, suunnittelu- ja toteutusvaiheet. Seuraavassa kuvataan tarkemmin näiden vaiheiden sisältö ja niihin liittyvät tehtävät, sekä esitellään ohjelmistokehityksessä hyödynnettävä mallinnus.

2.1 Määrittelyvaihe

Määrittelyvaiheessa selvitetään kehitettävään ohjelmistoon liittyvät vaatimukset (Haikala & Märijärvi 2004: 38). Haikala & Mikkonen (2011: 61) määrittelevät *vaatimuksen* joksikin, mitä tuotteella pystyy tekemään, tai (laatu-)ominaisuudeksi, joka tuotteella tulee olla. Vaatimukset voidaan jakaa toiminnallisiin ja ei-toiminnallisiin vaatimuksiin. *Toiminnallisilla vaatimuksilla* tarkoitetaan niitä *palveluita*, joita ohjelmiston tulee tarjota, sitä, miten ohjelmiston tulee *reagoida* tiettyihin syötteisiin ja sitä, miten sen tulee *käyttäytyä* tiettyissä tilanteissa. *Ei-toiminnalliset vaatimukset* voidaan luokitella tuote-, organisaatiokohtaisiin ja ulkoisiin vaatimuksiin (kuva 2). Tuotevaatimukset määrittelevät tai rajoittavat ohjelmiston käyttäytymistä. Tällaisia ovat esimerkiksi ohjelmiston käytettävyyteen liittyvät vaatimukset. Organisaatiokohtaiset vaatimukset perustuvat sekä ohjelmiston kehittäjän että asiakasorganisaation sääntöihin ja toimintatapoihin, kuten kehitysprosessiin liittyviin standardeihin. Ulkoiset vaatimukset perustuvat nimensä mukaisesti ohjelmiston ja koko kehitysprosessin ulkoisiin tekijöihin, kuten lakisäädöksiin ja eettisiin tekijöihin. Ei-toiminnalliset vaatimukset liittyvät usein ohjelmistoon kokonaisuutena, eivätkä niinkään yksittäisiin ohjelmiston ominaisuuksiin tai palveluihin. Käytännössä jako toiminnallisten ja ei-toiminnallisten vaatimusten välillä ei kuitenkaan ole niin selvää. Esimerkiksi jokin ei-toiminnallinen vaatimus, kuten tietoturvaan liittyvä vaatimus, voi luoda uusia toiminnallisia vaatimuksia. (Sommerville 2011: 84-88.)

Vaatimusten dokumentointiin liittyvän abstraktiotason perusteella puhutaan usein käyttäjä- ja ohjelmistovaatimuksista. (Sommerville 2011: 83; Haikala & Märijärvi 2004: 38).

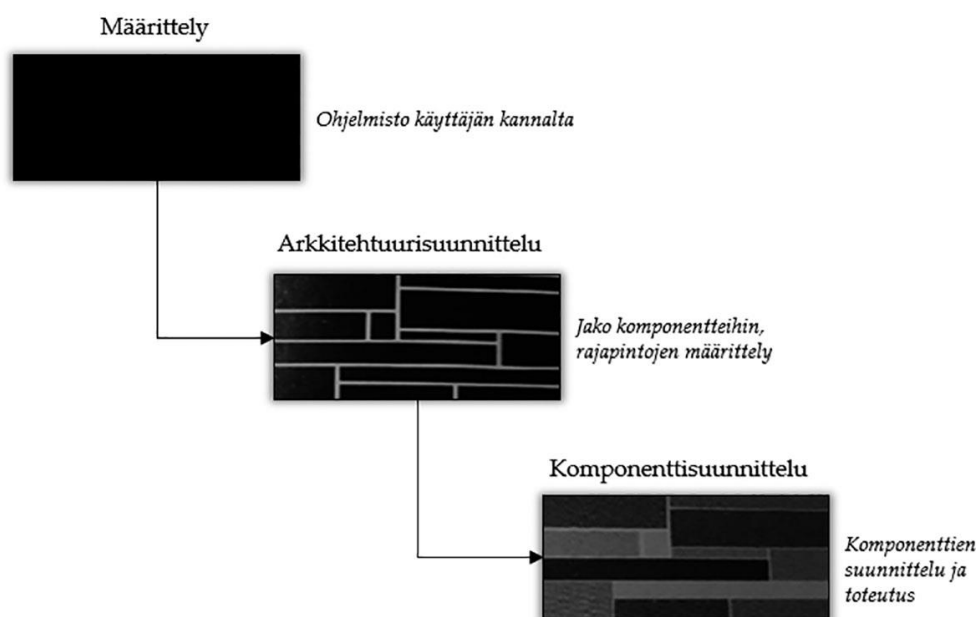
Käyttäjävaatimukset ovat käyttäjien tarpeisiin perustuvia kuvauksia siitä, mitä palveluita ohjelmiston odotetaan tarjoavan käyttäjille, ja ohjelmiston toimintaa rajoittavista tekijöistä. Käyttäjävaatimukset osoitetaan ohjelmiston käyttäjille, jotka eivät yleensä ole kiinnostuneet siitä, miten ohjelmisto toteutetaan tai muista yksityiskohdista. Näin ollen käyttäjävaatimukset ovat yleensä hyvin abstrakteja, usein luonnollisella kielellä sekä sopivilla kaavioilla ja taulukoilla esitettyjä kuvauksia. *Ohjelmistovaatimukset* osoitetaan ohjelmiston kehityksestä vastaaville, joten ne sisältävät yksityiskohtaisia kuvauksia ohjelmiston tarjoamista palveluista ja sen toiminnallisuuteen liittyvistä rajoitteista. Ohjelmistovaatimusten yhteydessä voidaan suunnitella myös ohjelmiston alustava arkkitehtuuri. Ohjelmistovaatimukset voidaan kirjoittaa luonnollisella kielellä, minkä lisäksi voidaan käyttää erilaisia graafisia tai matemaattisia malleja. Sekä käyttäjä- että ohjelmistovaatimusten tulee olla selkeitä, yksiselitteisiä, helppoja ymmärtää, testattavissa olevia, täydellisiä ja johdonmukaisia. (Sommerville 2011: 91–110.)



Kuva 2. Ei-toiminnallisten vaatimusten luokittelu, vapaasti suomennettu lähteestä (Sommerville 2011: 88).

2.2 Suunnitteluvaihe

Määrittelyvaiheessa kuvataan siis sitä, *mitä* kehitettävä ohjelmisto tekee. Suunnitteluvaiheessa puolestaan esitetään se, *miten* nämä toteutetaan. (Haikala & Märijärvi 2004: 40.) Suunnitteluvaiheessa kuvataan muun muassa kehitettävän ohjelmiston rakenne, sen sisältämät komponentit, komponenttien väliset rajapinnat sekä ohjelmiston käyttämä tietoaineisto. Vaiheen sisältämät osavaiheet riippuvat keskeisesti siitä, millaista ohjelmistoa ollaan kehittämässä. Yleensä vaihe kuitenkin sisältää ainakin arkkitehtuuri- ja komponenttisuunnittelun. (Sommerville 2011: 39-40.) Kuvassa 3 hahmotetaan yksinkertaistetusti ohjelmistosuunnittelun etenemistä.



Kuva 3. Ohjelmistosuunnittelun eteneminen yksinkertaistetusti (Haikala & Mikkonen 2011: 177).

Jos ohjelmiston toimintaan liittyy vuorovaikutus ihmisten kanssa, vaihe sisältää myös käyttöliittymäsuunnittelua (Pressman 2010: 312). Suunnitteluvaiheessa on huomioitava myös tietokannan suunnittelu, jos ohjelmiston tietoaineisto tallennetaan tietokantaan.

Suunnitteluvaiheen tuloksena syntyvien suunnitelmien yksityiskohtaisuus ja muu esitystapa riippuu kehitettävästä ohjelmistosta. Kriittiset ohjelmistot suunnitellaan yleensä hyvinkin yksityiskohtaisesti ennen kuin mitään toteutetaan. (Sommerville 2011: 40.)

2.2.1 Arkkitehtuurisuunnittelu

Ohjelmistosuunnittelun sisältämällä keinoilla ja menetelmillä pyritään siis ylittämään määrittelyvaiheessa muodostettujen vaatimusten ja käytettävissä olevan toteutusteknologian välistä kuilua. Keskeisin keino tässä on arkkitehtuurisuunnittelu, jonka tarkoituksena on muuntaa ohjelmiston kuvaus ongelmasta ratkaisuun. Arkkitehtuurisuunnittelussa kehitettävä ohjelmisto jaetaan pienempiin komponentteihin, joista jokaisella voi olla oma arkkitehtuurinsa. Lopulta päädytään niin konkreettisen tason ratkaisuihin, jotka voidaan toteuttaa ohjelmointikielen rakentein. (Haikala & Mikkonen 2011: 177.) Usein kehitettävä ohjelmisto on jollain tasolla yhteydessä myös muihin ulkopuolisiin järjestelmiin, kuten tietokantaan. Arkkitehtuurisuunnitelmassa voidaankin määritellä myös nämä ohjelmiston ulkoiset rajapinnat. (Munassar & Govardhan 2010: 96.)

Ohjelmiston *arkkitehtuurille* on olemassa erilaisia määritelmiä. Yksinkertaisimmillaan arkkitehtuuri kattaa ohjelmiston osat, ns. komponentit, ja niiden väliset suhteet. *Komponentti* on kokonaisuus, joka sisältää tiettyyn ohjelmiston osaan liittyvät toiminnot ja ominaisuudet. Kansainvälisen tekniikan alan järjestön IEEE:n (*Institute of Electrical and Electronics Engineers*) arkkitehtuurien kuvaamista koskevan standardin mukaan ohjelmistoarkkitehtuuri on ohjelmiston perusorganisaatio, joka sisältää ohjelmiston eri komponentit, niiden suhteet toisiinsa ja ympäristöön, sekä kehitysprosessia ohjaavat periaatteet. (Haikala & Mikkonen 2011: 178-179.)

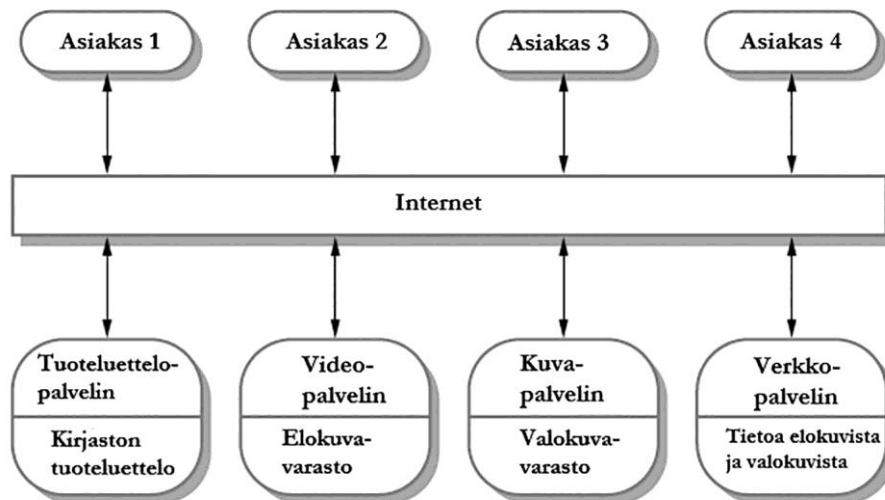
Arkkitehtuurisuunnittelu on siis ohjelmiston kokonaisrakenteen suunnittelua. Sen tavoitteena on ymmärtää, miten ohjelmisto tulee organisoida, jotta sille asetetut vaatimukset voidaan toteuttaa. (Sommerville 2011: 148.) Koska ohjelmistot ovat usein monimutkaisia kokonaisuuksia, niin arkkitehtuurisuunnittelussa keskeistä on juuri monimutkaisuu-den hallinta. Arkkitehtuurisuunnitteluun liittyviä periaatteita ovat muun muassa yksinker-

taisuus ja suoraviivaisuus, osittaminen, lokaalisuus sekä yhdenmukainen toteutusfilosofia, joilla voidaan saavuttaa selkeitä ja ymmärrettäviä ratkaisuja. *Yksinkertaisuudella* ja *suoraviivaisuudella* tarkoitetaan sitä, että toteutusvaihtoehtoja suunniteltaessa on pyrittävä löytämään aina mahdollisimman yksinkertaisin ja suoraviivaisin ratkaisu. *Osittamisperiaate* liittyy siihen, että ainoa keino ohjelmiston monimutkaisuuden hallitsemiseksi on ohjelmiston osittaminen hallittavimpiin kokonaisuuksiin, ns. komponentteihin. *Lokaalisuudella* tarkoitetaan puolestaan sitä, että ohjelmiston osittaminen suunnitellaan siten, että suunnittelupäätökset kapseloidaan osien sisään siten, että muutosten tekeminen kohdistuu yhteen komponenttiin kerrallaan. Suunnittelussa tulisikin tähdätä siihen, että ohjelmiston yksittäisiä osia tai muutaman osan muodostamia kokonaisuuksia voidaan toteuttaa ja testata erikseen. Onnistunut *arkkitehtuurifilosofia* on koko suunnitteluvaiheen avaintekijä, ja tästä syystä suunnittelussa ei kannata edetä, ennen kuin hyvä kokonaisratkaisu on löytynyt. Yksittäisten komponenttien ja niiden välisten rajapintojen ja riippuvuuksien hallintaan voidaan siirtyä vasta, kun arkkitehtuurissa käytettävästä perusfilosofiasta on sovittu. (Haikala & Mikkonen 2011: 180-182.)

Arkkitehtuurisuunnittelu on luova prosessi, ja sen tuotoksena syntyvän arkkitehtuurimallin laajuus ja esitystapa riippuu siitä, mihin mallia käytetään ja kenelle se osoitetaan. Abstraktimmat mallit voivat olla hyödyllisempiä ohjelmiston käyttäjille, jolloin niissä ei ole turhia yksityiskohtia. Jos taas malli on tarkoitettu ohjelmiston kehityksestä vastaaville, eli sitä käytetään esimerkiksi ohjelmiston muuhun suunnitteluun, toteutukseen ja dokumentointiin, sen tulee olla yksityiskohtaisempi. (Sommerville 2011: 148-151.)

Arkkitehtuurisuunnittelussa voidaan hyödyntää erilaisia valmiita suunnittelumalleja ja arkkitehtuurityylejä. Arkkitehtuurityylien tarkoitus on helpottaa kokonaisuuden hahmottamista, kun taas suunnittelumallit auttavat dokumentoimaan yksittäisiä suunnitteluratkaisuja. (Haikala & Mikkonen 2011: 187-189.) Arkkitehtuurityylit ovat yleisiä, hyväksi todettuja abstrakteja kuvauksia siitä, miten ohjelmisto tulisi organisoida sekä sisäisesti että suhteessa muihin järjestelmiin. Esimerkkejä tällaisista arkkitehtuurityyleistä on muun muassa kerrosarkkitehtuuri, MVC-arkkitehtuuri ja asiakas-palvelin-arkkitehtuuri. (Sommerville 2011: 155-156.) Tässä diplomityössä kehitettävän verkko-oppimisympäristön

suunnittelussa hyödynnetään asiakas-palvelin-arkkitehtuuria. *Asiakas-palvelin-arkkitehtuuri* on hajautettu malli, joka esittää sen, miten tieto ja sen käsittely on hajautettu eri osaluokkiin. Asiakas-palvelin-arkkitehtuuriin perustuva järjestelmä koostuu tietyistä palveluista tarjoavista palvelimista ja näitä palveluita pyytävistä asiakkaista sekä verkosta, joka sallii asiakkaan pääsyn näille palvelimille. *Palvelin* on tietokone tai muu laite, joka sisältää tietyn tyyppisten palveluiden tarjoamiseen tarkoitettua palvelinohjelmaa. *Asiakas* on myös tietokone tai muu laite, ja se sisältää ns. asiakasohjelman, joka pyytää palvelimelta erilaisia palveluita. Kuvassa 4 on esimerkki elokuvakirjastosta, jonka toiminta perustuu asiakas-palvelin-arkkitehtuuriin. Elokuvakirjastoa voi käyttää yhtäaikaaisesti useampi käyttäjä siten, että he pyytävät Internetin välittämänä asiakaslaitteellaan erilaisia palveluita aina siltä palvelimelta, joka hallinnoi kyseisen palvelun tarjoamiseen liittyvää aineistoa. (Sommerville 2011: 160-167.)



Kuva 4. Esimerkki elokuvakirjaston asiakas-palvelin-arkkitehtuurista, vapaasti suomennettu lähteestä (Sommerville 2011: 162).

2.2.2 Tietokantasuunnittelu

Tietokanta on karkeasti määriteltynä loogisesti yhteenkuuluva ja tallennettava tietokokoelma. Tämän lisäksi tietokannalta vaaditaan yleensä myös muita ominaisuuksia, kuten

useamman käyttäjän hallittu yhteiskäyttö, tietojen suojaus ja se, ettei tiedon rakenne riipu sitä käsittelevistä välineistä. *Relaatiotietokannat* ovat laajimmin käytettyjä tietokantoja, ja niiden toiminta perustuu matematiikan joukko-oppiin ja relaatio-käsitteeseen. Relaatiotietokantoja käsitellään lähes poikkeuksetta SQL-kyselykielellä (Structured Query Language). (Rantala 2006: 252-255.)

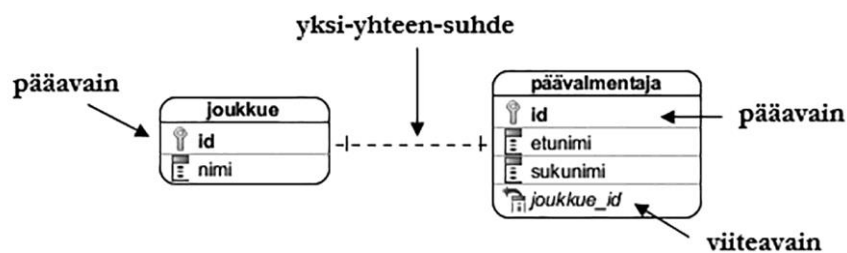
Tietokantasuunnittelu sisältää ohjelmiston käyttämän tietoaineiston rakenteen suunnittelua ja sitä, miten tämä tulisi esittää tietokannassa. Vaiheessa suoritettavat toiminnot riippuvat keskeisesti siitä, käytetäänkö jo olemassa olevaa tietokantaa vai luodaanko kokonaan uusi. (Sommerville 2011: 40.) Tietokantasuunnittelun pohjana on määrittelyvaiheessa kerätty tieto siitä, minkälaista tietoaineistoa ohjelman tulee varastoida ja käsitellä. *ER-mallin* (The Entity-Relationship model) avulla tietokannan rakenne voidaan kuvata siten, että sekä ohjelmiston kehittäjät että mahdollisesti myös sen loppukäyttäjät saavat siitä yhtenäisen käsityksen. (Connolly & Begg 2005: 342.)

ER-mallin laatimisessa on käytössä useita eri merkintätapoja, mutta sen sisältämistä elementeistä vallitsee kuitenkin yhteisymmärrys. Näitä elementtejä ovat muun muassa kohteet, ominaisuudet ja suhteet. *Kohde* on jokin tunnistettava ja yksilöitävissä oleva, ohjelmistoon ja sen toimintaan liittyvä kokonaisuus. Kullakin kohteella on tiettyjä *ominaisuuksia*. Kukin kohteen ominaisuus saa tietyn arvon tietystä arvojoukosta, mikä kuvaa kohdetta ja siitä tietokantaan tallennettavaa tietoa. Ne ominaisuudet, jotka yksilöivät kunkin kohteen ilmentymän, kutsutaan avaimiksi. Tällaisella avainominaisuudella tulee olla aina arvo, ja sen tulee olla jokaiselle kohteen ilmentymälle erilainen. Kohteella voi olla useampia avaimia, joista on aina valittava ns. pääavain. *Suhteella* kuvataan kohteiden välisiä riippuvuuksia. Suhteet voivat olla joko yksi-yhteen-, monta-yhteen-, tai monta-moneen-suhteita riippuen siitä kuinka moneen toisen kohteen ilmentymään kyseisen kohteen ilmentymä voi olla tällaisessa riippuvuussuhteessa. (Connolly & Begg 2005: 342-354.)

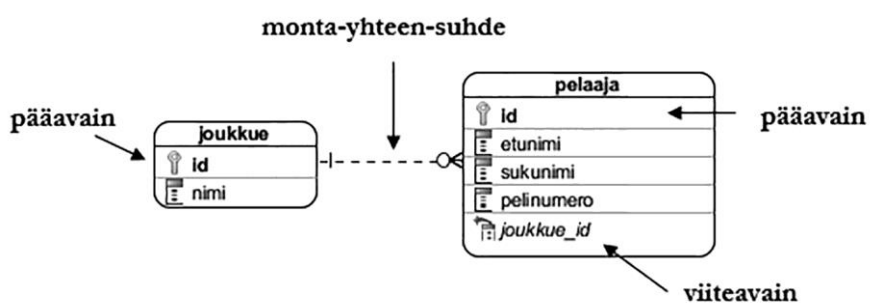
ER-mallin avulla saadaan yksinkertaisesti muodostettua tietokannan sisältämät taulut, eli relaatiot. Jokaista kohdetta varten luodaan oma relaatio, joka sisältää kaikki kohteen ominaisuudet. Mallin sisältämät monta-yhteen-suhteet käsitellään siten, että sen kohteen pääavain, joka on suhteen ”yksi-puolella”, lisätään suhteen ”monta-puolella” olevan kohteen

ns. viiteavaimeksi. Yksi-yhteen-suhteen käsittely voidaan puolestaan tehdä useammalla eri tavalla. Yksi vaihtoehto on yhdistää suhteessa olevat kohteet yhdeksi relaatioksi. Toinen vaihtoehto on lisätä toisen kohteen pääavain toisen kohteen viiteavaimeksi. Montamoneen-suhde käsitellään siten, että suhteesta muodostetaan oma relaatio siten, että siihen lisätään viiteavaimiksi kaikkien suhteeseen osallistuvien kohteiden pääavaimet. Pääavaimen siirtäminen toisen kohteen viiteavaimeksi vaatii aina kyseisen avaimen uudelleennimeämisen. (Connolly & Begg 2005: 463-470.) ER-mallit sisältävät myös useita muita elementtejä kuin tässä mainitut. Näihin ja muihin tietokannan relaatioiden luomisessa huomioitaviin asioihin voi tutustua tarkemmin esimerkiksi perehtymällä Connolly & Beggin (2015) teokseen ”Database Systems, A Practical Approach to Design, Implementation, and Management”.

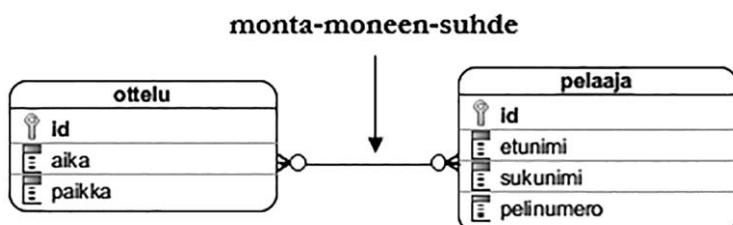
Seuraavassa hahmotetaan edellä kuvattuja käsitteitä jalkapalloturnauksen tulospalvelujärjestelmään liittyvän esimerkin avulla. Tällaisen järjestelmän tietokannan kohteita voisi olla esimerkiksi ”joukkue”, ”päävalmentaja”, ”pelaaja” ja ”ottelu”. Kohteen ”pelaaja” ominaisuuksia ovat esimerkiksi etu- ja sukunimi sekä pelinumero. Ominaisuuden pelinumero arvojoukko koostuu kaikista positiivisista kokonaisluvuista, ja ominaisuuksien etu- ja sukunimi arvojoukko erilaisista merkkijonoista. Koska pelaajilla voi olla samoja nimiä ja periaatteessa myös samoja pelinumeroita, niin pääavaimeksi voidaan valita esimerkiksi automaattisesti kasvava numeromuotoinen id-tunniste. Eli aina, kun tietokantaan tallennetaan uusi pelaaja, niin tämän pelaajan id on automaattisesti aina yhtä numeroa suurempi kuin edellisen. Kohteiden ”joukkue” ja ”päävalmentaja” suhde on yksi-yhteen, jos oletetaan, että kullakin joukkueella voi olla vain yksi päävalmentaja, ja kukin päävalmentaja ”ehtii” olla vain yhden joukkueen päävalmentaja (kuva 5). Kohteiden ”pelaaja” ja ”joukkue” välinen suhde on puolestaan monta-yhteen, sillä kussakin joukkueessa voi pelata useampi kuin yksi pelaaja, mutta kukin pelaaja voi kuulua vain yhteen joukkueeseen (kuva 6). Kohteiden ”ottelu” ja ”pelaaja” välinen suhde on monta-moneen, sillä kukin pelaaja voi olla turnauksessa mukana useammassa kuin yhdessä ottelussa, ja kuhunkin otteluun voi osallistua useampi kuin yksi pelaaja (kuva 7). Tämän suhteen tietokannassa esittämistä varten on luotava siis uusi relaatio, johon lisätään viiteavaimiksi molempien kohteiden pääavaimet, ja tämä esitetään kuvassa 8.



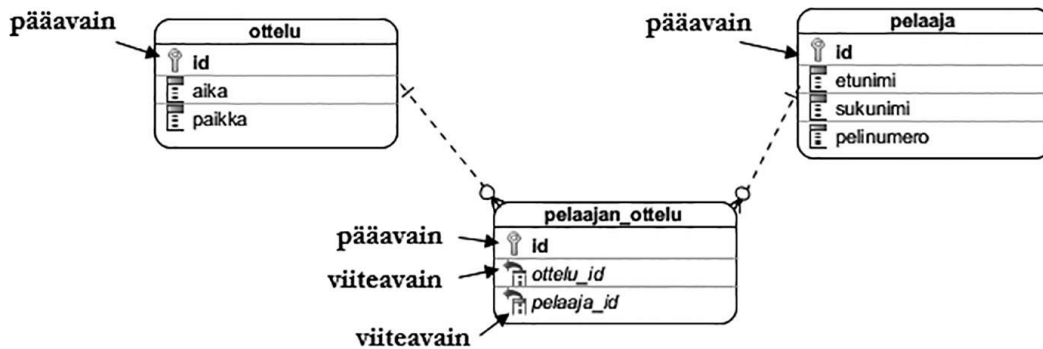
Kuva 5. Esimerkki tietokannan taulujen välisestä yksi-yhteen-suhteesta.



Kuva 6. Esimerkki tietokannan taulujen välisestä mona-yhteen-suhteesta.



Kuva 7. Esimerkki tietokannan taulujen välisestä mona-moneen-suhteesta.



Kuva 8. Esimerkki tietokannan taulujen välisen monta-moneen-suhteen korvauksesta uuden taulun avulla.

2.2.3 Komponenttisuunnittelu

Komponenttisuunnittelussa tarkastellaan kutakin arkkitehtuurisuunnitelman sisältämää komponenttia ja suunnitellaan, miten se toteutetaan. Komponenttisuunnitelmat voivat sisältää yksityiskohtaisia kuvauksia siitä, miten kukin komponentti tullaan toteuttamaan. Toisaalta ne voivat sisältää vain maininnan komponenttien odotetusta toiminnallisuudesta ja yksityiskohtaisempi suunnittelu tehdään vasta kunkin komponentin toteutusvaiheessa. (Sommerville 2011: 40.) Yksityiskohtaisessa, komponentteihin liittyvässä suunnittelussa on sama keskeinen pyrkimys kuin arkkitehtuurisuunnittelussakin, eli vähentää ja selkeyttää riippuvuuksia. Abstraktiotaso on kuitenkin lähempänä toteutusta. (Haikala & Mikkonen 2011: 183–184.)

2.2.4 Käyttöliittymäsuunnittelu

Käyttöliittymän tarkemman määritelmän mukaan se on 1) rajapinta, joka mahdollistaa informaation välittämisen ihmisen ja tietokoneen laitteiston tai ohjelmistokomponentin välillä, 2) ohjelmiston ja laitteiston muodostama kokonaisuus, jonka avulla käyttäjä voi olla vuorovaikutuksessa tietokoneen kanssa (ISO/IEC/IEEE 24765 2010: 390). Käyttöliittymäsuunnitteluun liittyvän prosessin luonne on samanlainen kuin muuhun ohjelmistokehitykseen liittyvä ongelmanratkaisuprosessi, eli ratkaistava ongelma *analysoidaan*

ennen ratkaisun suunnittelua ja toteutusta (Pressman 2010: 320). Käyttöliittymäsuunnittelua edeltävässä analyysivaiheessa voidaan hyödyntää esimerkiksi Benyonin (2010: 28-38) esittelemää PACT-analyysia, jossa pohditaan seuraavia asioita:

- 1) millaisia ovat ohjelmistoa käyttävät ihmiset (People)
- 2) millaisia toimintoja (Activities) ohjelmistossa suoritetaan
- 3) millainen on konteksti (Context), jossa ohjelmistoa käytetään
- 4) millaista teknologiaa (Technologies) ohjelmiston käyttö vaatii

Käyttöliittymäsuunnittelu perustuu määrittelyvaiheessa muodostettuihin ohjelmiston vaatimuksiin, ja sen tavoitteena on määritellä käyttöliittymän eri osat ja toiminnallisuudet siten, että käyttäjä voi suorittaa kaikki toiminnot siten, että käytettävyydelle asetetut tavoitteet täyttyvät (Pressman 2010: 312). Graafisen käyttöliittymän suunnittelu koostuu elementtien sommittelusta ja asettelusta kuvaruudulla. Samalla suunnitellaan navigointi, sekä valitaan värit, kuvat, symbolit, tyyli, tekstit ja kieli, joita käytetään yhdenmukaisesti koko käyttöliittymässä. (Pressman 2010: 329; Sampola 2008: 21-22.) Suunnitteluprosessin aikana tulevasta käyttöliittymästä voidaan tehdä erilaisia luonnoksia ja prototyyppejä joko perinteisellä ”kynä ja paperi”-menetelmällä tai käyttäen apuna erilaisia ohjelmia ja muita työkaluja. Näiden perusteella käyttöliittymää voidaan arvioida ja kehittää edelleen. (Benyon: 2010: 54-55.)

Käyttöliittymän suunnittelussa hyödynnettäviä periaatteita on kehitetty vuosien saatossa useita (Benyon: 2010: 89). Kenties yksi tunnetuin käyttöliittymäsuunnitteluun tarkoitettu periaatekokoelma on seuraava Nielsenin (1994: 30) kymmenen periaatteen lista:

1. *Näkyvyys:*
Käyttäjien on oltava koko ajan tietoisia ohjelmiston tilasta ja heille tulee antaa riittävästi palautetta tehdyistä toiminnoista.
2. *Yhteensopivuus todellisen maailman kanssa:*
Käyttöliittymän kieli, eli sen sisältämät sanat ja grafiikat, tulee olla käyttäjän kannalta selkeitä ja ymmärrettäviä.

3. *Käyttäjien hallinta ja vapaus:*

Käyttäjille on annettava mahdollisuus palata alkutilaan ei-toivotun toiminnon tai valinnan jälkeen.

4. *Johdonmukaisuus ja standardit:*

Käyttöliittymän sisältämät sanat, toiminnot, kuvakkeet jne. tulee olla kauttaaltaan johdonmukaisia, jotta käyttäjän ei tarvitse pohtia, tarkoittavatko ne samoja asioita vai ei.

5. *Virheiden ehkäisy:*

Erilaiset virhetilanteet tulee pyrkiä ehkäisemään muun muassa pyytämällä käyttäjiltä varmistuksia toimintoihin, joita ei myöhemmin voida kumota.

6. *Muistikuormituksen minimoiminen:*

Käyttöliittymä tulee suunnitella siten, ettei käyttäjien tarvitse muistaa asioita siirtäessään näkymästä toiseen.

7. *Käytön tehokkuus ja joustavuus:*

Tottuneille käyttäjille tulee tarjota oikopolkuja eri toimintoihin, jotta nämä voivat käyttää ohjelmistoa tehokkaasti.

8. *Estetiikka ja minimalistinen suunnittelu:*

Käyttöliittymän ei tule sisältää turhia ja merkityksettömiä asioita ja tietoa.

9. *Virhetilanteiden käsittely:*

Käyttäjille tulee antaa selkeät ja tarkat ilmoitukset virhetilanteista.

10. *Opastus ja ohjeet:*

Käyttäjille tulee tarjota selkeää ja helposti saatavilla olevaa opastusta ja ohjeita.

Nielsenin käytettävyyssäännöt ovat melko yleisiä, mutta toisaalta ne ovat kattavia ja melko helppoja muistaa, joten ne on mahdollista ottaa huomioon koko suunnittelun ajan (Riihiaho 1998: 4).

Käyttöliittymän suunnitteluun on olemassa useita eri lähestymistapoja. *Luovassa lähestymistavassa* (craft approach) käyttöliittymän suunnittelijan taidot ja kokemus ovat suunnitteluprosessissa merkittävässä roolissa. Onnistunut käyttöliittymä riippuu siis keskeisesti suunnittelijan luovuudesta ja intuitiivisesta päättelystä. (Nam & Smith-Jackson 2007: 26.)

2.3 Toteutusvaihe

Toteutusvaiheessa ohjelmistosta luodaan toimiva versio. Arkkitehtuurisuunnitelman mukaiset komponentit ohjelmoidaan jollakin ohjelmointikielellä joko alusta asti tai muokkaamalla tarkoitukseen soveltuvia valmiita ohjelmistokomponentteja. Toteutusvaihe voi sisältää myös komponentin yksityiskohtaisemman suunnittelun, jos ohjelmisto on arkkitehtuurisuunnittelussa jaettu riittävän pieniin komponentteihin. Kunkin komponentin valmistuttua se testataan, eli suoritetaan ns. *yksikkötestausta*, jolla varmistetaan, että komponentti on määrittystensä mukainen. Toteutusvaiheen lopuksi valmiit komponentit integroidaan toimivaksi ohjelmistoksi, jolle testausvaiheessa suoritetaan systemaattinen ohjelmistotestaus. (Sommerville 2011: 193; Haikala & Mikkonen 2011: 184; Haikala & Märijärvi 2004: 41.) Seuraavassa esitellään tässä työssä kehitettävän verkko-oppimisympäristön toteutusvaiheessa käytettävät menetelmät.

PHP (PHP: Hypertext Preprocessor) on avoin, eli kaikille ilmainen ohjelmisto, joka on kehitetty nimenomaan dynaamisten verkkosivustojen rakentamiseen. Termillä PHP voidaan viitata sekä PHP-kieleen että niihin teknisiin ratkaisuihin, jotka mahdollistavat PHP-kielisten ohjelmien suorittamisen. PHP-kieli on palvelimella tulkittava ohjelmointikieli, eli sillä tehdyt ohjelmat suoritetaan palvelimella siellä olevan PHP-tulkin avulla. Tämän jälkeen selaimelle lähetetään valmis *HTML*-kielinen (Hypertext Markup Language) tiedosto. (Rantala 2005: 9-14.)

JavaScript on ohjelmointikieli, jolla voidaan elävöittää verkkosivuja ja luoda sellaista dynaamista vuorovaikutusta käyttäjän kanssa, johon ei päästä palvelimella tulkittavilla kielillä. JavaScript soveltuu parhaiten lyhyiden ja yksinkertaisten ohjelmien tekoon, ja ne voidaan liittää suoraan HTML-tiedostoon. JavaScript-ohjelma suoritetaan selaimessa, johon on upotettu JavaScript-tulkki. (Peltomäki & Nykänen 2006: 89-97.) JavaScript-ohjelmien toteutuksessa voidaan hyödyntää jQueryä, joka on erilaisia JavaScript-ohjelmia sisältävä kirjasto (The jQuery Foundation 2016).

CSS (Cascading Style Sheets) on verkkosivun ulkoasun vaikuttamiseen tarkoitettu tyylikieli. CSS-kielillä tehdyt tyylimäärittelyt voi tehdä verkkosivuun liittyvän tiedoston alussa, tiedoston sisällä elementtikohtaisesti, tai erillisessä CSS-tiedostossa. Useammat selaimet tukevat CSS-tyylejä, mutta tyylien määrittely ja toteutus riippuu selaimesta. Näin ollen CSS-tyylien käyttäminen vaatii huolellista testaamista eri selaimilla. (Peltomäki & Nykänen 2006: 260–261.)

Responsiivisuus voidaan toteuttaa pelkällä CSS-kielillä. *Mediakyselyt* (media queries) ovat CSS:n menetelmiä, joilla verkkosivu voidaan skaalata kullekin laitteelle sopivaksi. Seuraavassa on esimerkki yhdestä mediakyselystä:

```
@media screen and (min-width: 1024px) {
/* CSS-kieli tilanteisiin, joissa leveys on vähintään 1024 pikseliä */
}
```

Mediakyselyissä voidaan tutkia sekä laitteen kokoa (esim. min-device-width) tai näkymän kokoa (esim. min-width). Tietokoneissa näkymä riippuu selainikkunan koosta, jota käyttäjä voi muokata. Mobiililaitteissa tilanne on toinen, sillä siinä näkymä voikin olla suurempi kuin laitteen näyttö. Esimerkiksi älypuhelimien näytön leveys voi olla 320 pikseliä, mutta selaimen leveys voi olla 980 pikseliä. Näin käyttäjä voi kutistaa ja suurentaa sivua haluamallaan tavalla. HTML-kielen meta-elementin avulla voidaan kuitenkin hallita selaimen näkymää eri laitteissa. (Marcotte 2011: 64-81.) Seuraava HTML-kieli tekee selaimen näkymän leveydestä saman kuin mikä laitteen näytön koko on (width=device-width) ilman oletusarvoista suurennusta (initial-scale=1).

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

2.4 Mallinnus ohjelmistokehityksessä

Mallinnus on yksi keskeinen osa ohjelmistokehitystä. Kyseessä on luova prosessi, jossa kehitettävästä ohjelmistosta muodostetaan tietystä näkökulmasta abstrakti malli. *Ulkoisesta näkökulmasta* tehty malli kuvaa ohjelmiston kontekstin tai ympäristön. *Vuorovaikutusnäkökulmasta* tehty malli kuvaa joko ohjelmiston vuorovaikutuksen ympäristönsä kanssa, tai ohjelmiston komponenttien välisen vuorovaikutuksen. *Rakenteellisesta näkökulmasta* tehty malli kuvaa ohjelmiston organisaation tai sen tietoaineiston rakenteen, jota ohjelmisto prosessoi. *Käyttäytymisnäkökulmasta* tehty malli kuvaa ohjelmiston dynaamisen käyttäytymisen, ja sen miten se reagoi erilaisiin tapahtumiin. (Sommerville 2011: 119.)

Malleja voidaan käyttää ohjelmistokehityksen eri vaiheissa. Toisaalta malli voidaan suunnata ohjelmiston käyttäjille siten, että sen avulla käyttäjät saavat käsityksen ohjelmiston toiminnoista, ominaisuuksista jne. Yhteistä kaikentyypisille malleille on se, että niillä on aina selkeä tarkoitus ja ne kuvaavat ohjelmistoa yksinkertaistetusti siten, että ne sisältävät ainoastaan ohjelmiston merkityksellisimmät piirteet. (Sommerville 2011: 119.)

Mallit kuvataan mallinnuskielen avulla. Mallinnuskieli koostuu *notaatiosta* eli merkinöistä ja säännöistä, jotka kertovat, kuinka mallia käytetään. Lisäksi malleja kuvataan usein visuaalisesti siten, että suurin osa mallin tietosisällöstä esitetään graafisilla symboleilla ja yhteyksillä. Visuaalisia kuvauksilla voidaan esittää monimutkaisia yhteyksiä ja helpottaa käytännön työtä. Sanonta ”kuva kertoo enemmän kuin tuhat sanaa” pätee myös mallinnuksessa, mutta joidenkin asioiden esittämiseen luonnollinen kieli voi olla paras vaihtoehto. (Eriksson & Penker 2000: 1-6.)

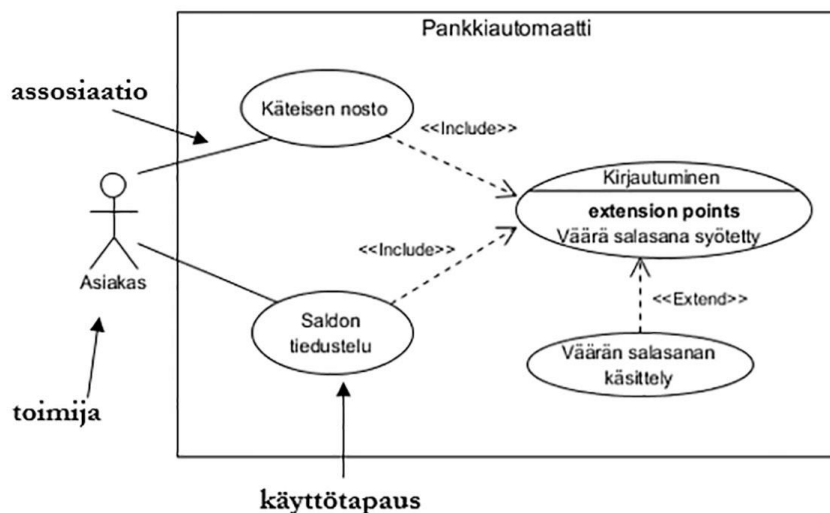
Malleja voidaan laatia hyvinkin joustavasti eikä notaation yksityiskohtiin tarvitse takerua. Mallin yksityiskohtaisuus ja täsmällisyys riippuu siitä, mihin mallia on tarkoitus käyttää. Jos malleja käytetään dokumentointiin, niiden ei tarvitse olla täydellisiä, kunhan

ne ovat kuvaavat ohjelmistoa täsmällisesti. Jos malleista luodaan suorittavaan muotoon käännettäviä lähdekielisiä tiedostoja, niiden tulee olla sekä täydellisiä että täsmällisiä. (Sommerville 2011: 120-121.)

UML (Unified Modelling Language) on 1990-luvulla kehitetty yhtenäistetty mallinnuskieli, joka on sekä virallisesti että käytännössä muodostunut mallinnuskielten standardiksi. UML:ää käytetään ohjelmistokehityksessä laaja-alaisesti erilaisten ohjelmistojen mallintamiseen. Sitä voidaan käyttää kehitysprosessin eri vaiheissa aina vaatimusten dokumentoinnista lopullisen tuotteen testaamiseen. (Eriksson & Penker 2000: 2-7.) Seuraavassa esitellään UML-kaavioista käyttötapauskaavio ja aktiviteettikaavio, joita käytetään tässä diplomityössä uuden verkko-oppimisympäristön kehityksessä.

Käyttötapauskaavio on vuorovaikutusnäkökulmasta laadittu malli, jolla voidaan mallintaa ohjelmiston toiminnallisia vaatimuksia. Käyttötapauskaavion tärkeimmät osat ovat käyttötapaukset, toimijat ja mallinnettava ohjelmisto. *Käyttötapaus* on ohjelmiston toiminto, ja jokainen käyttötapaus kuvastaa jotain, mitä ulkoinen toimija haluaa ohjelmiston tekävän. *Mallinnettavaa ohjelmistoa* pidetään ns. ”mustana laatikkona”, joka tarjoaa käyttötapausten kuvaamat toiminnot ulkoisille toimijoille. *Toimija* on joko ihminen, joka käyttää ohjelmistoa, tai toinen järjestelmä tai laite, joka on vuorovaikutuksessa mallinnettavan ohjelmiston kanssa. (Eriksson & Penker 2000: 8-39.)

Kuvassa 9 on esimerkki pankkiautomaattiin liittyvästä käyttötapauskaaviosta. Ohjelmisto kuvataan laatikkona ja ohjelmiston nimi on laatikon yläpuolella tai sisällä. Toimija kuvataan tikku-ukkona, jonka alapuolella on toimijan nimi, joka kuvaa sen roolia ohjelmistossa. Jokaisella toimijalla on oltava yhteys yhteen tai useampaan käyttötapaukseen. Käyttötapaus kuvataan ellipsinä, jonka sisällä tai alapuolella on käyttötapauksen nimi. Käyttötapaus nimetään sen suorittaman toiminnon mukaan ja on usein pidempi kuin yksi sana. Käyttötapaus sijoitetaan yleensä ohjelmiston rajojen sisäpuolelle. Käyttötapaukset kytketään toimijoihin *yhteyksillä*, joita kutsutaan myös *assosiaatioiksi*. Assosiaatiot kuvaavat sitä, minkä toimijoiden kanssa käyttötapaus on vuorovaikutuksessa. (Eriksson & Penker 2000: 39-47.)



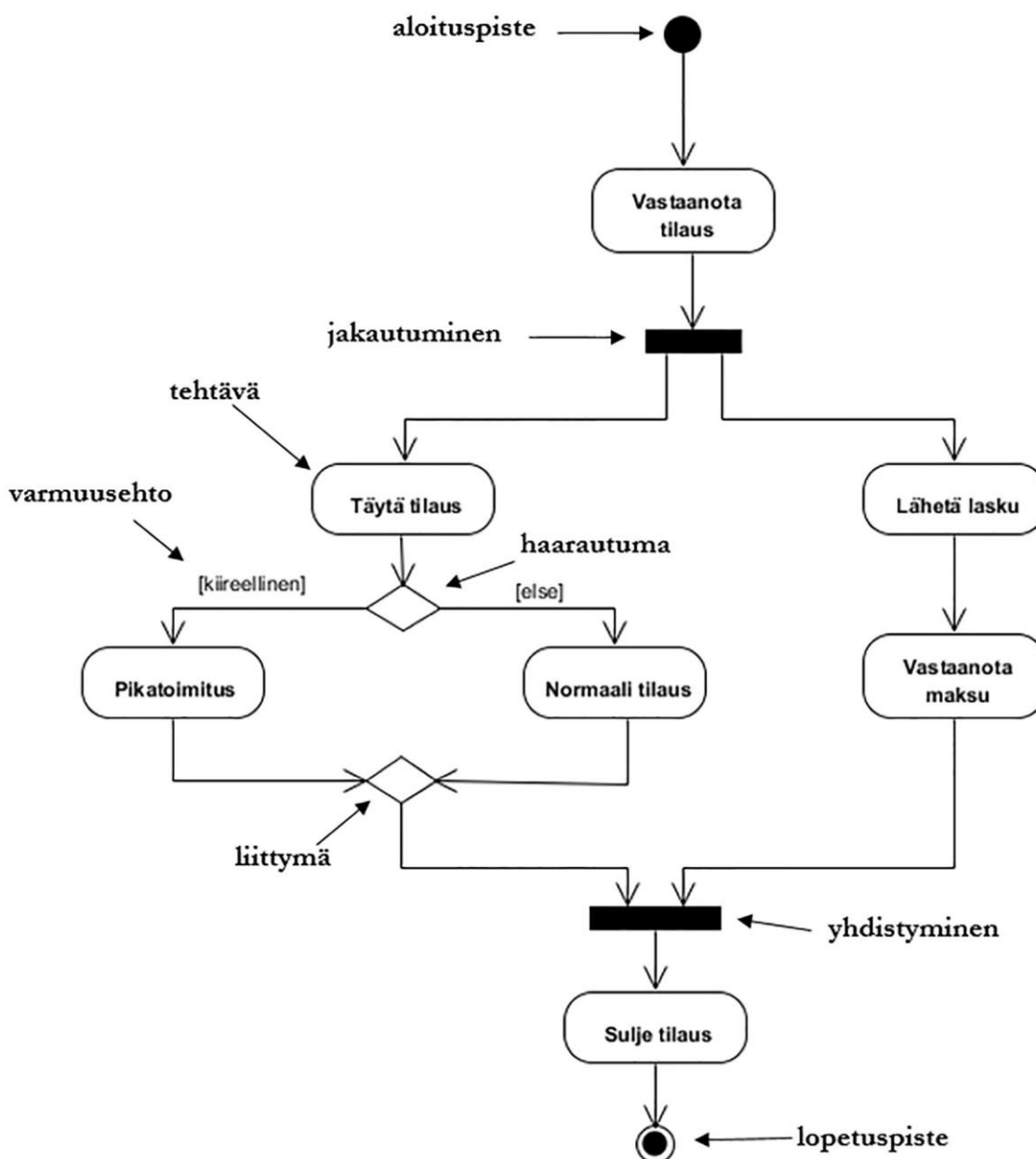
Kuva 9. Esimerkki käyttötapauskaaviosta.

Käyttötapauksilla voi olla monenlaisia keskinäisiä suhteita. *Sisältymissuhdetta* (include) voidaan käyttää silloin, kun jokin yhteinen toiminta on osana monessa eri käyttötapauksessa ja halutaan välttää tämän toiminnan kuvauksen kopioimista. *Laajennussuhteen* (extend) avulla laajentava käyttötapaus voi lisätä toiminnallisuutta peruskäyttötapaukseen. Tässä yhteydessä peruskäyttötapauksessa on ns. *laajennuspisteitä* (extension points), ja laajentava käyttötapaus voi lisätä toiminnallisuutta vain näihin laajennuspisteisiin. (Fowler & Scott 2002: 40-41.)

Käyttötapauskaavion sisältämiä käyttötapauksia on usein hyvä kuvata sanallisesti tarkemmin. Käyttötapausten kuvaustapa voi vaihdella paljonkin eikä UML:ssä ole määritetty sille mitään standardia. Kuvauksessa käytetään ohjelmiston käyttäjien ymmärtämään kieltä ja termistöä, ja sen tulisi kattaa ainakin seuraavat asiat:

- Käyttötapausten *tarkoitus*
- Käyttötapausten *käyttäjät*
- Käyttötapausten *kulku*
- Käyttötapausten *poikkeukset*
- Käyttötapausten *lopputulos*

Näiden lisäksi käyttötapauksen kuvaukseen voidaan sisällyttää myös muita osia, kuten esimerkiksi *alkuehdot*, joiden on toteuduttava ennen kuin käyttötapaus voi alkaa. Koska UML ei tue ei-toiminnallisten vaatimusten dokumentointia, niin yksi vaihtoehto tähän on kirjata ne käyttötapauksen sanallisen kuvauksen *Muut vaatimukset*-kohtaan. (Fowler & Scott 2002: 37; Haikala & Mikkonen 2011: 79-80; Eriksson & Penker 2000: 49-50.)



Kuva 10. Esimerkki aktiviteettikaaviosta, muokattu lähteestä (Fowler & Scott 2002: 114)

Aktiviteettikaavio on käyttäytymisnäkökulmasta tehty malli, jolla voidaan kuvata ohjelmiston toimintaan liittyvien tapahtumien kulkua aikajärjestyksessä (Eriksson & Penker 2000: 19). Kuvassa 10 on esimerkki tilausprosessiin liittyvästä aktiviteettikaaviosta. *Aloituspiste* kuvataan täytettynä ympyränä ja *lopetuspiste* ympyränä täytetyn ympyrän ympärillä. *Tehtävät* kuvataan suorakaiteena, jolla on pyöristetyt kulmat. Tehtävät kuvataan sanallisesti. Tehtävien väliset siirtymät kuvataan nuolilla, joihin voidaan liittää muun muassa varmuusehtoja. *Varmuusehto* on totuusarvolauseke ja se esitetään hakasulkujen sisällä. Jos siirtymässä on varmuusehto, siirtymä laukaistaan, kun ehto tulee todeksi. Jos siirtymään ei ole liitetty mitään, se laukeaa heti, kun se tehtävä on suoritettu, josta kyseinen siirtymä lähtee. (Fowler & Scott 2002: 114-116; Eriksson & Penker 2000: 110-129.)

Aktiviteettikaavio kuvaa siis tehtävien tapahtumajärjestystä, ja siinä tuetaan sekä ehdollista että rinnakkaista toimintaa. Ehdollista toimintaa kuvataan haarautumilla ja liittymillä. *Haarautumassa* on yksi tuleva siirtymä ja monta varmuusehdoilla varustettua lähtevää siirtymää. Varmuusehtojen tulee olla toisensa poissulkevia siten, että vain yksi lähtevistä siirtymistä voidaan valita. *Liittymässä* on yksi monta tulevaa ja yksi lähtevä siirtymä, ja sillä kuvataan haarautuman aiheuttaman ehdollisen toiminnan päättymistä. Haarautumat ja liittymät kuvataan usein vinoneliöillä, mikä tekee eri vaihtoehtoiset haarat ja yhteenliittymiset selvemmiksi. Rinnakkaista toimintaa kuvataan jakautumisilla ja yhdistymisillä. *Jakautumisessa* on yksi tuleva siirtymä ja monta lähtevää siirtymää siten, että kun tuleva siirtymä käynnistyy, kaikki lähtevät siirtymät suoritetaan rinnakkain. Rinnakkainen toiminta vaatii synkronointia, joka voidaan esittää *yhdistymisellä*. Yhdistymisessä on monta tulevaa ja yksi lähtevä siirtymä siten, että lähtevä siirtymä tapahtuu vasta, kun kaikki tuleviin siirtymiin liittyvät tehtävät on suoritettu loppuun. (Fowler & Scott 2002: 114-116; Eriksson & Penker 2000: 129.)

3 VERKKO-OPPIMISYMPÄRISTÖN KEHITYKSESSÄ HUOMIOITAVIA TEKIJÖITÄ

Verkko-oppimisympäristön avulla sekä opiskelijoille että opettajille, ja muille koulutuksesta vastaaville, luodaan tieto- ja viestintäteknikkaa hyödyntäen yhteinen työskentelypaikka, jossa voidaan toteuttaa opiskelua. Verkko-oppimisympäristö on informaatiojärjestelmä, jonka tavoitteena on sekä tukea oppimista että monipuolistaa opetusta. Se sisältää tähän tarkoitukseen soveltuvia toiminnallisuuksia ja oppimateriaaleja, motivoi opiskelijaa ja arvioi opiskelijan osaamista (Räsänen 2002: 3; Mäcklin 2012: 4-11.) Seuraavassa esitellään tarkemmin keskeisimpiä, erityisesti aktivoivaa opetusta tukevan verkko-oppimisympäristön kehityksessä huomioitavia tekijöitä.

3.1 Verkko-oppimisympäristön rakenteen ja toiminnallisuuksien suunnittelu

Verkko-oppimisympäristön suunnittelu koostuu sen rakenteen ja toiminnallisten ratkaisujen suunnittelusta. Suunnittelun taustalla on aina oppimiskäsitys. Nykyaikaisen oppimiskäsityksen mukaan opiskelija on *aktiivinen toimija*, joka valikoi itse opittavia asioita, konstruoi niitä aktiivisesti ja päättää itse omat tarpeensa, kiinnostukset ja näkemykset oppimansa mukaan. Lisäksi nykyaikaisessa oppimiskäsityksessä korostuvat *yhteistoiminnallisuus* ja *vuorovaikutus*. (Mäcklin 2012: 9-10.) Opetusta, joka perustuu tähän oppimiskäsitykseen, kutsutaan usein Lonkan & Lonkan (1991: 12) määrittelemäksi aktivoivaksi opetuksi.

Lonka & Lonka (1991: 28-43) esittelevät aktivoivaa opetusta käsittelevässä teoksessaan useita aktivoivia opetusmenetelmiä, joita on kokeiltu usean vuoden ajan muun muassa Helsingin yliopistossa ja Helsingin normaalikoulussa. Eräitä esimerkkejä näistä opetusmenetelmistä ovat aktivoiva luento, porinaryhmät ja virittävät kysymykset. Aktivoivassa luennossa erilaisia tekniikoita hyödyntämällä luento-opetusta elävöitetään ja tehostetaan oppimista. Porinaryhmillä tarkoitetaan opetuksen yhteydessä muodostettavia ryhmiä,

joissa opiskelijat keskustelevat jostain yhteisestä aiheesta. Virittävät kysymykset esitetään opiskelijoille ennen luentoa, jolloin he voivat valmistautua niihin. Kysymykset käsitellään luennon aikana joko suullisesti tai kirjallisesti.

Nykyaikainen oppimiskäsitys ja aktivoivat opetusmenetelmät näkyvät verkko-oppimisympäristössä muun muassa siten, sen avulla opiskelija voi itse valita missä järjestyksessä hän suorittaa asioita, mihin hän syventyy ja mitkä ohittaa tuttuina asioina (Mäcklin 2012: 9-10). Opiskelija joutuu arvioimaan omaa oppimistaan ja ottamaan vastuuta oppimisestaan. Hän saa oppimiseen liittyvää palautetta ja tukea opettajilta, muilta opiskelijoilta ja muilta opetukseen osallistuvilta henkilöiltä. (Sampola 2008: 29.) Aktivoiviin opetusmenetelmiin, tai muihin pedagogisiin ratkaisuihin, liittyviä toiminnallisuuksia lisäämällä verkko-oppimisympäristöön käytettävissä oleva teknologia ja pedagogiset linjaukset saadaan sulavasti integroitua keskenään.

Verkko-oppimisympäristön ideaalirakenne muodostuu tuottamisesta, vuorovaikutuksesta, arvioinnista ja hallinnasta. *Tuottamisella* tarkoitetaan mahdollisuutta tuottaa oppimateriaaleja oppimisympäristöön. Verkko-oppimisympäristön materiaali koostuu tyypillisesti opettajan, opiskelijoiden ja kolmannen osapuolen tuottamasta materiaalista, sekä linkeistä muualle verkossa olevaan materiaaliin. Materiaalien avulla opiskelijat voivat muun muassa syventää tietojaan tarpeensa mukaan. *Vuorovaikutuksella* tarkoitetaan oppimisprosessin eri osapuolten välisiä vuorovaikutusmahdollisuuksia. Verkko-oppimisympäristö voi sisältää esimerkiksi alustan, jossa keskustellaan opintojaksoon liittyvistä asioista tai esitetään kysymyksiä opettajalle ja muille opiskelijoille. Näin voidaan tukea opiskelijan tiedonrakentamisprosessia. *Arviointia* käytetään muun muassa opiskelijoiden oppimisen seuraamiseen sekä palautekanavana. *Hallinta* liittyy hallinnollisiin välineisiin, joilla hallinnoidaan, ylläpidetään ja organisoidaan verkko-oppimisympäristöä, sen toimintaa ja käyttäjiä. (Mäcklin 2012: 9-18; Sampola 2008: 31-33; Räsänen 2002: 4.) Lisäksi verkko-oppimisympäristössä eri toimijoille annetaan tyypillisesti erilaisia rooleja, ja rooleihin liittyviä käyttöoikeuksia. Rooleja voivat olla esimerkiksi oppimisympäristön ylläpitäjä, materiaalin tuottaja, opettaja, opiskelija, asiantuntija ja avustaja. (Sampola 2008: 31.)

Verkko-oppimisympäristön suunnittelussa syntyneet ratkaisut vaikuttavat keskeisesti myös siihen, soveltuuko verkko-oppimisympäristö ensisijaisesti monimuotokoulutukseen, perinteiseen luokkaopetukseen vai molempiin. Monimuotokoulutuksella tarkoitetaan lähiopetusta, joka vuorottelee etäopiskelun, itsenäisen opiskelun ja työssäoppimisen kanssa. (Räsänen 2002: 3-4.)

3.2 Verkko-oppimisympäristön käyttöliittymä

Verkko-oppimisympäristön suunnittelussa erityisen tärkeää on myös käyttöliittymän ja sen käytettävyyden huomiointi (Mäcklin 2012: 17). *Käyttöliittymä* mahdollistaa käyttäjän ja verkko-oppimisympäristön välisen vuorovaikutuksen (Sampola 2008: 21). Verkko-oppimisympäristöllä on korkea *käytettävyys*, jos sitä on sekä helppoa että tehokasta käyttää, se on helppo muistaa, siinä on vain harvoja virheitä ja se on subjektiivisesti miellyttävä (Silius & Tervakari 2003: 2). Käyttöliittymäsuunnittelu vaikuttaa keskeisesti koko verkko-oppimisympäristön onnistumiseen ja se tuleekin integroida osaksi oppimisympäristön muuta suunnitteluprosessia (Nam & Smith-Jackson 2007: 24).

Verkko-oppimisympäristön käyttöliittymän suunnittelun tavoitteita ovat muun muassa selkeys, johdonmukaisuus ja ulkonäön miellyttävyys. Turhaa informaatiota tulee välttää. Motivaatio vaikuttaa oppimiseen ja aktivoi opiskeluun liittyvää toimintaa. Verkko-oppimisympäristössä opiskelumotivaatio voi perustua pakkoon tai palkitseviin koearvosanoihin, mutta se voi herätä myös tiedonjanosta ja suuntautua ongelmanratkaisuun. Verkko-oppimisympäristössä muuhun kuin ulkoisiin palkkioihin liittyvää opiskelumotivaatiota voi herättää mielenkiinto oppimisympäristössä olevan asian sisältöä ja käyttömahdollisuuksia kohtaan. Opiskelumotivaatioon voidaan vaikuttaa myös jollain verkko-oppimisympäristön käyttöliittymän sisältämällä ns. ulkoisella tekijällä, kuten värien ja kuvien kiehtovuudella. (Sampola 2008: 20-22; Sadik 2004: 27-28.)

Mikäli verkko-oppimisympäristöä on tarkoitus käyttää erilaisilla ja erikokoisilla laitteilla (tietokoneilla, älypuhelimilla, tableteilla jne.), käyttöliittymäsuunnittelun yhteydessä on

myös huomioitava nk. *responsiivisuus*. Tällä tarkoitetaan sitä, että verkko-oppimisympäristön layout on suunniteltu siten, että sen sisältö näyttää hyvältä kaikissa laitteissa näytön koosta riippumatta. (Marcotte 2011: 107.) Kuvassa 11 esitetään visuaalisesti se, miten responsiivisesti toteutetun verkkosivun sisältö mukautuu käytettävän laitteen mukaisesti.



Kuva 11. Responsiivisesti toteutetun verkkosivun sisältö mukautuu kuhunkin laitteeseen sopivaksi (Faisal 2012).

Verkko-oppimisympäristön kehitysprosessin yhteydessä on tärkeää luoda myös jonkinlainen arviointimenetelmä, jonka avulla verkko-oppimisympäristön käytettävyyttä voidaan kehittää. Menetelmän tulisi olla erityisesti verkko-oppimisympäristöjen käytettävyyssarviointiin suunniteltu, jolloin huomio on muun muassa siinä, kuinka hyvin verkko-oppimisympäristö tukee opiskelijan oppimisprosessia. Käytettävyyden arviointi onkin keskeinen osa verkko-oppimisympäristön kokonaisarviointia. (Nam & Smith-Jackson 2007: 24; Silius & Tervakari 2003: 2.)

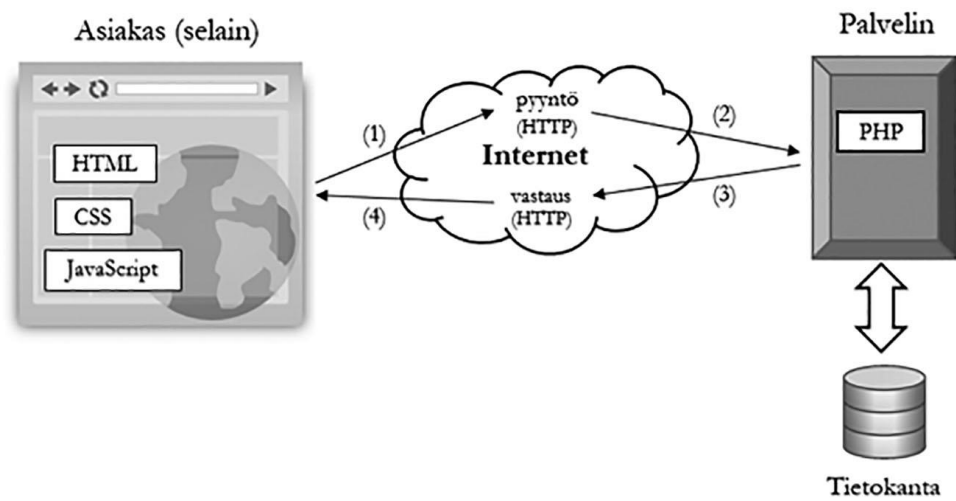
3.3 Verkko-oppimisympäristöön liittyvä tekniikka

Verkko-oppimisympäristö toteutetaan tietoverkkojen avulla, joten sitä voidaan pitää verkkosivustona. *Verkkosivusto* määritellään yksittäisen henkilön tai organisaation tuottamaksi tiettyä aihetta käsitteleväksi verkkosivujen joukoksi. *Verkkosivu* on Internetissä olevaan tiedostoon perustuva tietokokonaisuus, joka voidaan esittää käyttäjän laitteella.

(Sanastokeskus TSK ry.) Seuraavassa kuvataan yleisesti verkkosivustoihin liittyvä tekniikka, joka tulee huomioida verkko-oppimisympäristön kehityksessä.

Verkkosivut luodaan käyttäen HTML-sivunkuvauskieltä, ja ne jaetaan toimintaperiaatteensa mukaan staattisiin, eli muuttumattomiin, ja dynaamisiin, eli muuttuviin, verkkosivuihin. *Staattisen verkkosivun* lukijan mahdollisuudet vaikuttaa sivun sisältöön ovat rajoittuneet lähinnä sivulle upotettujen hyperlinkkien käyttämiseen, joten vuorovaikutusmahdollisuudet sivun kanssa ovat hyvin vähäiset. Usein kuitenkin tarvitaan käyttäjän ja verkkosivun välistä välitöntä vuorovaikutusta. Verkkosivun sisältöä tai verkkosivustoon liittyvää tietoa on esimerkiksi voitava muuttaa käyttäjän kyseisellä hetkellä tekemien valintojen perusteella. Tämän tyyppiseen tarpeeseen on kehitetty tekniikoita, joiden avulla verkkosivuista voidaan tehdä osittain tai kokonaan *dynaamisia*. Tällaisia tekniikoita ovat muun muassa selaimessa suoritettavat asiakastekniikat, kuten JavaScript ja CSS, sekä palvelimella suoritettavat palvelintekniikat, kuten PHP. (Rantala 2005: 3-6.)

Verkkosivustojen toiminta perustuu kappaleessa 2 esitettyyn asiakas-palvelin-malliin ja niihin liittyvä tietoaaineisto tallennetaan tyypillisesti tietokantaan. Internetissä asiakasohjelmiana toimii tyypillisesti Internet-selain, kuten Internet Explorer tai Mozilla Firefox. Kun käyttäjä aktivoi selaimen osoitekenttään kirjoittamansa, tiettyyn verkkosivuun viit-tavan URL-osoitteen, selain lähettää palvelimelle palvelupyynnön. (Rantala 2005: 2; Microsoft 2011: 2-3.) URL (Uniform Resource Locator) on merkkijono, joka viittaa tiedoston tai muun resurssin, kuten kuvan tai videon, sijaintiin verkossa (Berners-Lee, Fielding & Masinter 2005: 4-6). Palvelin vastaa palvelupyyntöön lähettämällä selaimelle URL-osoitteen määrittämän tiedoston, jonka sisältämän HTML-kielen avulla Internet-selain tulkitsee tiedoston sisällön ja näyttää verkkosivun käyttäjälle. Sekä palvelupyyntö että vastaus lähetetään käyttämällä URL-osoitteen määrittämää *yhteykäytäntöä* eli *protokollaa*, joka tyypillisesti on HTTP (Hypertext Transfer Protocol). (Rantala 2005: 2; Microsoft 2011: 2-3.) Kuvassa 12 havainnollistetaan dynaamiseen verkkosivuun liittyvää toimintaa ja tekniikkaa asiakas-palvelin-mallissa.



Kuva 12. Dynaaminen verkkosivu asiakas-palvelin-mallissa.

4 AKTIVOIVAA OPETUSTA TUKEVAN VERKKO-OPPIMISYMPÄRISTÖN VAATIMUKSET

Seuraavassa esitellävät uuden verkko-oppimisympäristön toiminnalliset ja ei-toiminnalliset ohjelmistovaatimukset on muodostanut tämän työn tekijä analyttisellä mallintamisella oppimisympäristön kehitysprosessin määrittelyvaiheessa, joka on osa tätä diplomityötä. Vaatimusten muodostamisessa on hyödynnetty pääosin kappaleessa 3 esiteltyjä verkko-oppimisympäristön kehityksessä huomioitavia tekijöitä. Lisäksi vaatimusten muodostamisessa on huomioitu ne tekijät, jotka vaaditaan, jotta verkko-oppimisympäristö soveltuu käytettäväksi kurssimuotoisissa oppilaitoksissa.

Uuden verkko-oppimisympäristön keskeisimpänä tehtävänä on tukea aktivoivaa opetusta. Se suunnataan kaikille sellaisille oppilaitoksille, joissa opetus tapahtuu kurssimuotoisesti. Verkko-oppimisympäristön nimeksi annetaan Cuulis-oppimisympäristö, ja siitä tehdään aluksi vain suomenkielinen versio. Cuulis-oppimisympäristön vaatimukset esitellään seuraavassa sekä luonnollisella kielellä että sopivia UML-kaavioita ja taulukoita käyttäen mahdollisimman täsmällisesti ja johdonmukaisesti siten, että niiden perusteella oppimisympäristön kehityksen myöhemmät suunnittelu- ja toteutusvaiheet voidaan suorittaa huolellisesti ja suoraviivaisesti. Lisäksi vaatimukset muotoillaan siten, että jokainen niistä on myöhemmin testattavissa. Samalla tarkoituksena on tarjota selkeä ja kattava kuva Cuulis-oppimisympäristön toiminnoista, ominaisuuksista ja rajoitteista sekä oppimisympäristön käyttäjille että kaikille siitä kiinnostuneille. Vaatimusten esittelyn yhteydessä annetaan myös alustava kuvaus Cuulis-oppimisympäristön arkkitehtuurista, millä muun muassa helpotetaan eri toimintojen hahmottamista. Lisäksi sitä voidaan käyttää arkkitehtuurisuunnittelun pohjana.

4.1 Toiminnalliset vaatimukset

Cuulis-oppimisympäristö toimii selainpohjaisesti Internetissä siten, että sen *etusivulle* on kaikilla vapaa pääsy. Sen varsinainen käyttöönotto vaatii kuitenkin rekisteröitymisen.

Etusivulla rekisteröitynyt käyttäjä voi kirjautua sisään Cuulis-oppimisympäristöön. Kirjautumisen jälkeen käyttäjälle avautuu *pääsivusto*, joka sisältää erilaisia yleisiä toimintoja liittyen muun muassa käyttäjiin, kursseihin ja oppilaitoksiin. Opetukseen liittyvät kurssi-kohtaiset toiminnot sijaitsevat erillisellä *kurssisivustolla*.

Cuulis-oppimisympäristön rekisteröityneitä käyttäjiä ovat opettajat, opiskelijat ja ylläpitäjät. Oppimisympäristön kokonaisvaltaisesta ylläpidosta ja hallinnasta vastaa yleinen ylläpitäjä. Oppilaitoskohtaisesta ylläpidosta vastaa oppimisympäristöön rekisteröitynyt opettaja tai joku muu oppilaitoksen valtuuttama henkilö. Kunkin oppilaitoksen oppilaitoskohtaisia ylläpitäjiä voi olla useita. Cuulis-oppimisympäristö sisältää siis viisi käyttäjäroolia:

- Opettaja
- Opiskelija
- Yleinen ylläpitäjä
- Oppilaitoskohtainen ylläpitäjä
- Opettaja ja oppilaitoskohtainen ylläpitäjä

Seuraavassa esitellään kaikki Cuulis-oppimisympäristön etusivun, pääsivuston ja kurssisivuston toiminnot, sekä kuvataan kootusti näihin sivustoihin liittyvät viestiliikennetoiminnot.

4.1.1 Etusivun toiminnot

Rekisteröityminen Cuulis-oppimisympäristöön tapahtuu siis oppimisympäristön etusivulla. Rekisteröityessään käyttäjä syöttää seuraavat tiedot:

- etunimi
- sukunimi
- sähköpostiosoite
- rooli
- ensisijainen oppilaitos

Jokainen näistä tiedoista on pakollinen. Käyttäjän roolia ja ensisijaista oppilaitosta ei voi myöhemmin muuttaa. Jos käyttäjä rekisteröityessään syöttää näihin virheellisiä tietoja, täytyy käyttäjä poistaa oppimisympäristöstä, minkä jälkeen hän voi suorittaa rekisteröitymisen uudestaan. Käyttäjän sähköpostiosoite toimii oppimisympäristön käyttäjätunnuksena, jotta sen muistaminen olisi helpompaa.

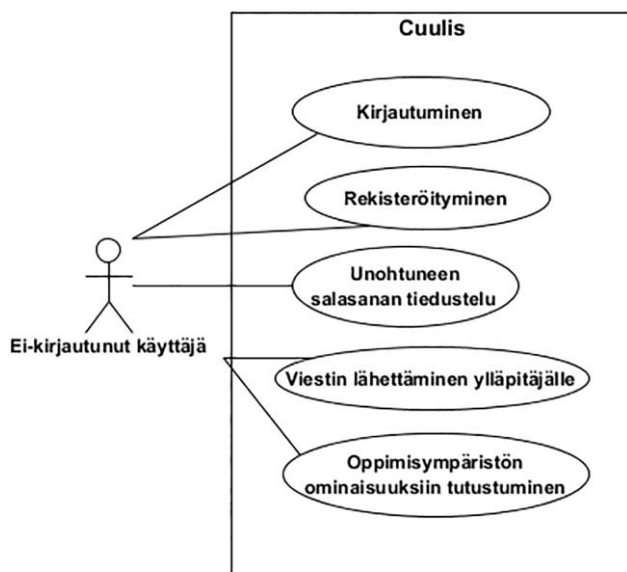
Opiskelijat rekisteröityvät oppimisympäristöön roolissa ”Opiskelija” ja opettajat roolissa ”Opettaja”. Jos käyttäjän tarkoituksena on toimia ainoastaan oppilaitoskohtaisena ylläpitäjänä, hän rekisteröityy roolissa ”Muu”. Käyttäjä, jonka tarkoituksena on toimia sekä opettajana että oppilaitoskohtaisena ylläpitäjänä, rekisteröityy roolissa ”Opettaja”. Tämän jälkeen joko yleinen ylläpitäjä tai kyseisen oppilaitoksen toinen ylläpitäjä lisää hänet oppilaitoksen ylläpitäjäksi, jolloin käyttäjän rooliksi vaihtuu ”Opettaja ja oppilaitoskohtainen ylläpitäjä”. Käyttäjä, jonka rooli on ”Yleinen ylläpitäjä” rekisteröidään oppimisympäristöön manuaalisesti.

Onnistuneen rekisteröitymisen jälkeen roolissa ”Opiskelija” rekisteröityneelle käyttäjälle lähetetään sähköpostiin vahvistuslinkki, jonka kautta käyttäjä pääsee antamaan käyttäjätunnuksensa liittyvän salasanan, jolloin hän vahvistaa lopullisesti rekisteröitymisensä. Koska roolissa ”Opettaja” ja ”Muu” rekisteröityneillä käyttäjillä on Cuulis-oppimisympäristössä vahvempia oikeuksia eri toimintojen suhteen, näiden käyttäjien rekisteröityminen vaatii sen oppilaitoksen ylläpitäjän hyväksynnän, jonka he valitsivat rekisteröitymisen yhteydessä ensisijaiseksi oppilaitoksekseen. Kun ylläpitäjä on hyväksynyt käyttäjän rekisteröitymisen, käyttäjälle lähetetään sähköpostiin salasanan syöttämiseen vievä rekisteröitymisen vahvistuslinkki. Kun roolissa ”Muu” rekisteröitynyt käyttäjä on vahvistanut oman rekisteröitymisensä, joko yleisen tai oppilaitoskohtaisen ylläpitäjän tulee vielä lisätä hänet jonkin oppilaitoksen ylläpitäjäksi. Vasta tämän jälkeen käyttäjä saa Cuulis-oppimisympäristöön käyttöoikeudet, ja hänen rooliksi vaihtuu ”Oppilaitoskohtainen ylläpitäjä”. Jos ylläpitäjä hylkää käyttäjän rekisteröitymisen, käyttäjä saa tiedon tästä sähköpostitse ja käyttäjän rekisteröinti poistetaan.

Käyttäjän rekisteröitymisen yhteydessä antamat tiedot kootaan *käyttäjäprofiiliin*. Muille käyttäjille käyttäjäprofiilissa näkyy rekisteröitymisen yhteydessä annettavista tiedoista

käyttäjän nimi, rooli, ja ensisijainen oppilaitos. Sähköpostiosoite ei siis ole kaikille näkyvä, mutta käyttäjät voivat lähettää toisilleen viestejä tähän sähköpostiosoitteeseen oppimisympäristön sisällä. Ylläpitäjät näkevät kuitenkin käyttäjäprofiilissa kaikki käyttäjään liittyvät tiedot.

Rekisteröityneet käyttäjät *kirjautuvat sisään* oppimisympäristöön etusivulla antamalla käyttäjätunnuksena toimivan sähköpostiosoitteen sekä salasanan. Mikäli käyttäjä on unohtanut salasanansa, hän voi etusivun kautta *tilata tunnuksen uudelleenaktivointilinkin*, jonka avulla hän voi vaihtaa salasanansa uuteen. Uudelleenaktivointilinkin tilaaminen vaatii kuitenkin käyttäjätunnuksen muistamisen, sillä kyseinen linkki lähetetään tähän käyttäjätunnuksena toimivaan sähköpostiosoitteeseen.



Kuva 13. Käyttötapauskaavio Cuulis-oppimisympäristön etusivun toiminnoista.

Lisäksi etusivulla kaikki käyttäjät voivat *lukea tietoa* Cuulis-oppimisympäristön ominaisuuksista ja *lähettää viestin* Cuulis-oppimisympäristön yleiselle ylläpitäjälle. Viestin lähetyksen yhteydessä käyttäjä voi halutessaan pyytää vastausta viestiin joko sähköpostitse tai puhelimitse. Toimintoa voivat käyttää esimerkiksi ne rekisteröityneet käyttäjät, jotka ovat unohtaneet kirjautumisessa vaaditun käyttäjätunnuksen. Kaikki etusivun sisältämät

toiminnot esitetään kuvan 13 käyttötapauskaaviolla. Lisäksi rekisteröitymiseen ja unohtuneen salasanan tiedusteluun liittyvistä käyttötapauksista annetaan tarkemmat kuvaukset taulukoiden 1 ja 2 avulla.

Taulukko 1. Rekisteröityminen-käyttötapauksen tarkempi kuvaus.

TARKOITUS:	Rekisteröityminen.
KÄYTTÄJÄ:	Kuka tahansa oppimisympäristön etusivulla oleva käyttäjä.
KULKU:	<ol style="list-style-type: none"> 1) Käyttäjä valitsee toiminnon ”Rekisteröityminen”. 2) Rekisteröintilomake avautuu. 3) Käyttäjä syöttää etunimen, sukunimen, sähköpostiosoitteen (toimii käyttäjätunnuksena), roolin (”Opettaja”/”Opiskelija”/”Muu”) ja ensisijaisen oppilaitoksen.
POIKKEUKSET:	Rekisteröinti ei onnistu: <ol style="list-style-type: none"> a) Kaikkia kenttiä ei ole täytetty. b) Sähköpostiosoite ei ole oikeanmuotoinen. c) Sähköpostiosoite on jo rekisteröity.
LOPPUTULOS:	Käyttäjä on rekisteröity oppimisympäristöön. Ja <ol style="list-style-type: none"> a) Käyttäjä kirjautunut roolissa ”Opiskelija”: Käyttäjä saa sähköpostiinsa vahvistuslinkin, jonka kautta käyttäjä vahvistaa rekisteröitymisensä syöttämällä haluamansa salasanan. b) Käyttäjä kirjautunut roolissa ”Opettaja”/”Muu”: Käyttäjä saa sähköpostiinsa vahvistusilmoituksen, jossa käyttäjälle ilmoitetaan, että rekisteröityminen odottaa ensisijaisen oppilaitoksen ylläpitäjän hyväksyntää.

Taulukko 2. Unohtuneen salasanan tiedustelu-käyttötapauksen tarkempi kuvaus.

TARKOITUS:	Unohtuneen salasanan tiedustelu.
KÄYTTÄJÄ:	Salasanan unohtanut rekisteröitynyt käyttäjä.
ALKUEHDOT:	Käyttäjä on rekisteröitynyt.
KULKU:	<ol style="list-style-type: none"> 1) Käyttäjä valitsee toiminnon ”Unohtuneen salasanan tiedustelu”. 2) Tunnuksen uudelleenaktiivointilomake avautuu, johon käyttäjä syöttää käyttäjätunnuksena toimivan sähköpostiosoitteensa. 3) Käyttäjä saa sähköpostiinsa tunnuksen uudelleenaktiivointilinkin, jonka kautta pääsee antamaan uuden salasanan.
POIKKEUKSET:	Salasanan vaihto ei onnistu: <ol style="list-style-type: none"> a) Käyttäjä ei muista käyttäjätunnuksena toimivaa sähköpostiosoitettaan. b) Käyttäjän antamaa sähköpostiosoitetta ei ole rekisteröity.
LOPPUTULOS:	Käyttäjän salasana on vaihdettu.

Edellä kuvattujen toimintojen perusteella Cuulis-oppimisympäristön etusivu koostuu siis seuraavista osioista:

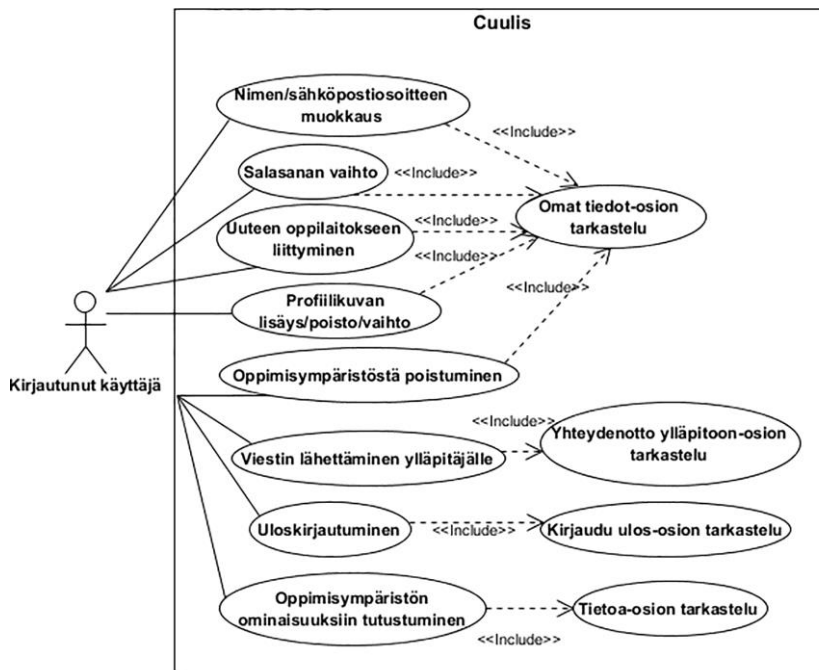
- Kirjautuminen
- Rekisteröityminen
- Unohtuneen salasanan tiedustelu
- Tietoa
- Yhteydenotto ylläpitoon

4.1.2 Pääsivuston toiminnot

Onnistuneen sisäänkirjautumisen jälkeen käyttäjälle avautuu pääsivusto, joka sisältää erilaisia toimintoja riippuen käyttäjän roolista. Jokaisella sisäänkirjautuneella käyttäjällä on kuitenkin pääsivustolla roolista riippumatta yhteisiä toimintoja, jotka jakautuvat osioihin:

- Omat tiedot
- Yhteydenotto ylläpitoon (ei ”Yleinen ylläpitäjä”-käyttäjällä)
- Tietoa
- Kirjaudu ulos

Näiden osioiden sisältämät toiminnot esitetään kuvan 14 käyttötapauskaaviolla. *Omat tiedot*-osiossa käyttäjä voi tarkastella omaa käyttäjäprofiiliaan ja halutessaan muokata nimeään ja sähköpostiosoitettaan, vaihtaa salasanaan, liittyä uusiin oppilaitoksiin, lisätä, vaihtaa tai poistaa profiilikuvansa, sekä poistaa itsensä oppimisympäristöstä. Käyttäjän, jonka rooli on ”Opettaja” tai ”Opettaja ja oppilaitoskohtainen ylläpitäjä”, liittyminen uuteen oppilaitokseen edellyttää kyseisen oppilaitoskohtaisen ylläpitäjän hyväksymisen. Käyttäjä, jonka rooli on ”Oppilaitoskohtainen ylläpitäjä”, ei voi liittyä rekisteröitymisensä jälkeen uusiin oppilaitoksiin, sillä hänen tehtävänä on toimia ainoastaan kyseisen oppilaitoksen ylläpitäjänä.



Kuva 14. Käyttötapauskaavio Cuulis-oppimisympäristön pääsivuston yhteisistä toiminnoista.

Pääsivuston *Tietoa*-osio on sama kuin etusivun samanniminen osio, eli siinä käyttäjä voi tutustua Cuulis-oppimisympäristön ominaisuuksiin. *Yhteydenotto ylläpitoon*-osiossa ”Opiskelija”- ja ”Opettaja”-käyttäjät voivat lähettää viestin sen oppilaitoksen ylläpitäjälle, jonka he valitsivat rekisteröinnin yhteydessä ensisijaiseksi oppilaitoksekseen. ”Opettaja ja oppilaitoskohtainen ylläpitäjä”- ja ”Oppilaitoskohtainen ylläpitäjä”-käyttäjät voivat puolestaan tässä osiossa lähettää viestin Cuulis-oppimisympäristön yleiselle ylläpitäjälle. ”Yleinen ylläpitäjä”-käyttäjällä tätä osiota ja sen sisältämää toimintoa ei luonnollisestikaan ole. *Kirjautu ulos*-osion kautta käyttäjä voi kirjautua ulos oppimisympäristöstä.

Muut Cuulis-oppimisympäristön pääsivuston toiminnot riippuvat keskeisesti käyttäjän roolista. ”Yleinen ylläpitäjä”-käyttäjä vastaa Cuulis-oppimisympäristön kokonaisvaltaisesta ylläpidosta liittyen oppimisympäristön käyttäjiin, kursseihin ja oppilaitoksiin sekä käyttäjien oppimisympäristöön lisäämiin tiedostoihin. ”Yleinen ylläpitäjä”-käyttäjän omat pääsivuston toiminnot jakautuvat siis seuraaviin osioihin:

- Käyttäjät
- Kurssit
- Oppilaitokset
- Tiedostot

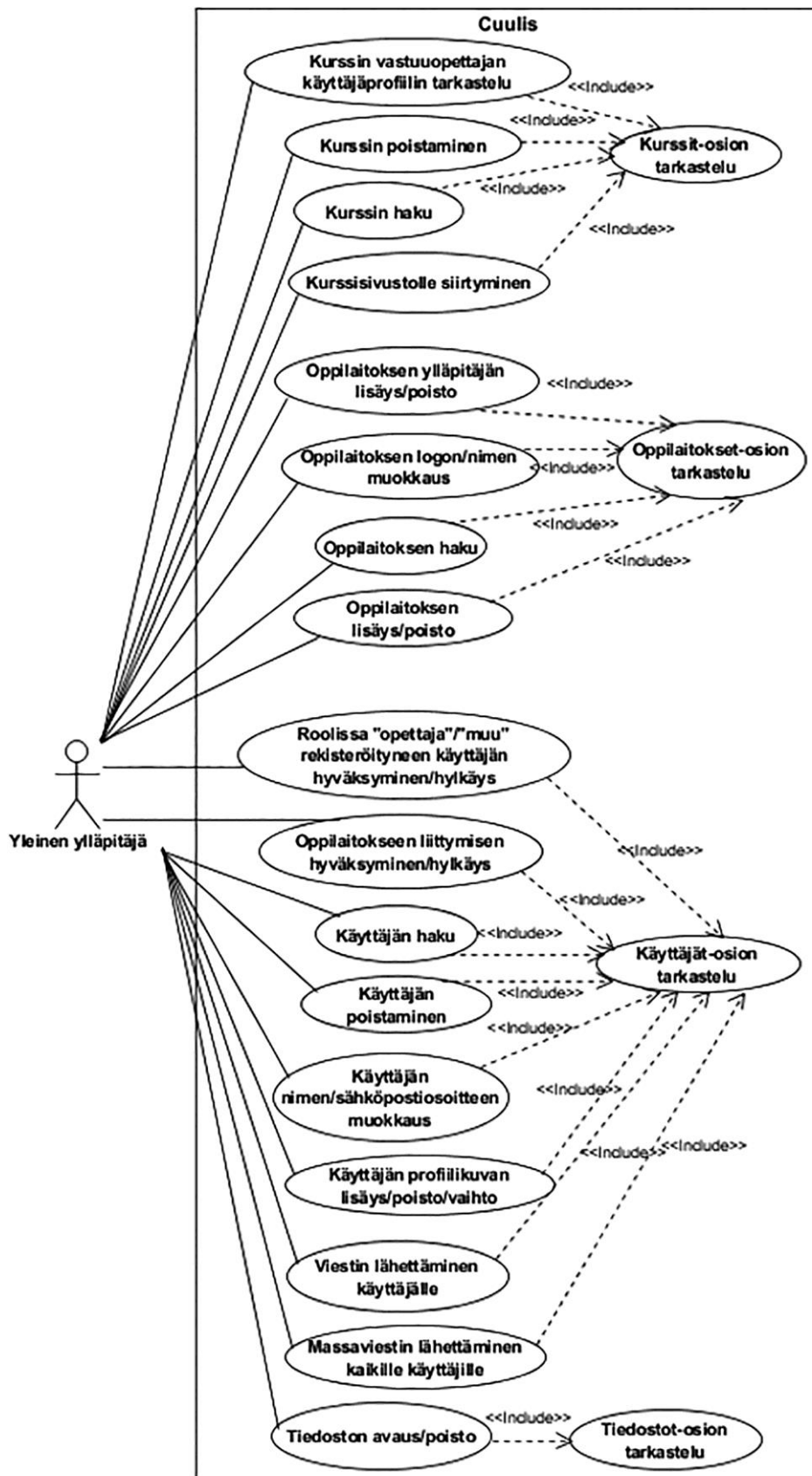
Kuvan 15 käyttötapauskaaviolla esitetään ”Yleinen ylläpitäjä”-käyttäjän toiminnot näiden osioiden osalta. *Käyttäjät*-osiossa ”Yleinen ylläpitäjä”-käyttäjä voi tarkastella kaikkien Cuulis-oppimisympäristöön liittyneiden käyttäjien käyttäjäprofileja, muokata niiden sisältämiä tietoja roolia ja oppilaitoksia lukuunottamatta, lähettää viestejä sekä yksittäisille käyttäjille että yhtäaikaisesti kaikille käyttäjille ja poistaa käyttäjiä oppimisympäristöstä. Lisäksi ”Yleinen ylläpitäjä”-käyttäjä voi kaikkiin oppimisympäristöön lisättyjen oppilaitosten osalta hyväksyä tai hylätä roolissa ”Opettaja” tai ”Muu” rekisteröityneen käyttäjän rekisteröitymisen ja ”Opettaja”-tai ”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjän liittymiseen uuteen oppilaitokseen.

Kurssit-osiossa ”Yleinen ylläpitäjä”-käyttäjä voi tarkastella kaikkia oppimisympäristöön lisättyjä kursseja ja niiden vastuuopettajan käyttäjäprofileita, poistaa kursseja ja siirtyä vapaasti haluamansa kurssin kurssisivustolle.

”Yleinen ylläpitäjä”-käyttäjä vastaa uusien oppilaitosten lisäämisestä oppimisympäristöön, ja tämä tapahtuu *Oppilaitokset*-osiossa. Lisätessään uutta oppilaitosta käyttäjä syöttää oppilaitoksesta seuraavat tiedot:

- nimi
- logo
- oppilaitoskohtaiset ylläpitäjät

Oppilaitokset-osiossa ”Yleinen ylläpitäjä”-käyttäjä voi myös muokata näitä oppilaitokseen liittyviä tietoja sekä poistaa oppilaitoksia. *Tiedostot*-osiossa ”Yleinen ylläpitäjä”-käyttäjä voi puolestaan tarkastella ja poistaa Cuulis-oppimisympäristön käyttäjien lisäämiä tiedostoja.



Kuva 15. "Yleinen ylläpitäjä"-käyttäjän yleiset toiminnot Cuulis-oppimisympäristössä.

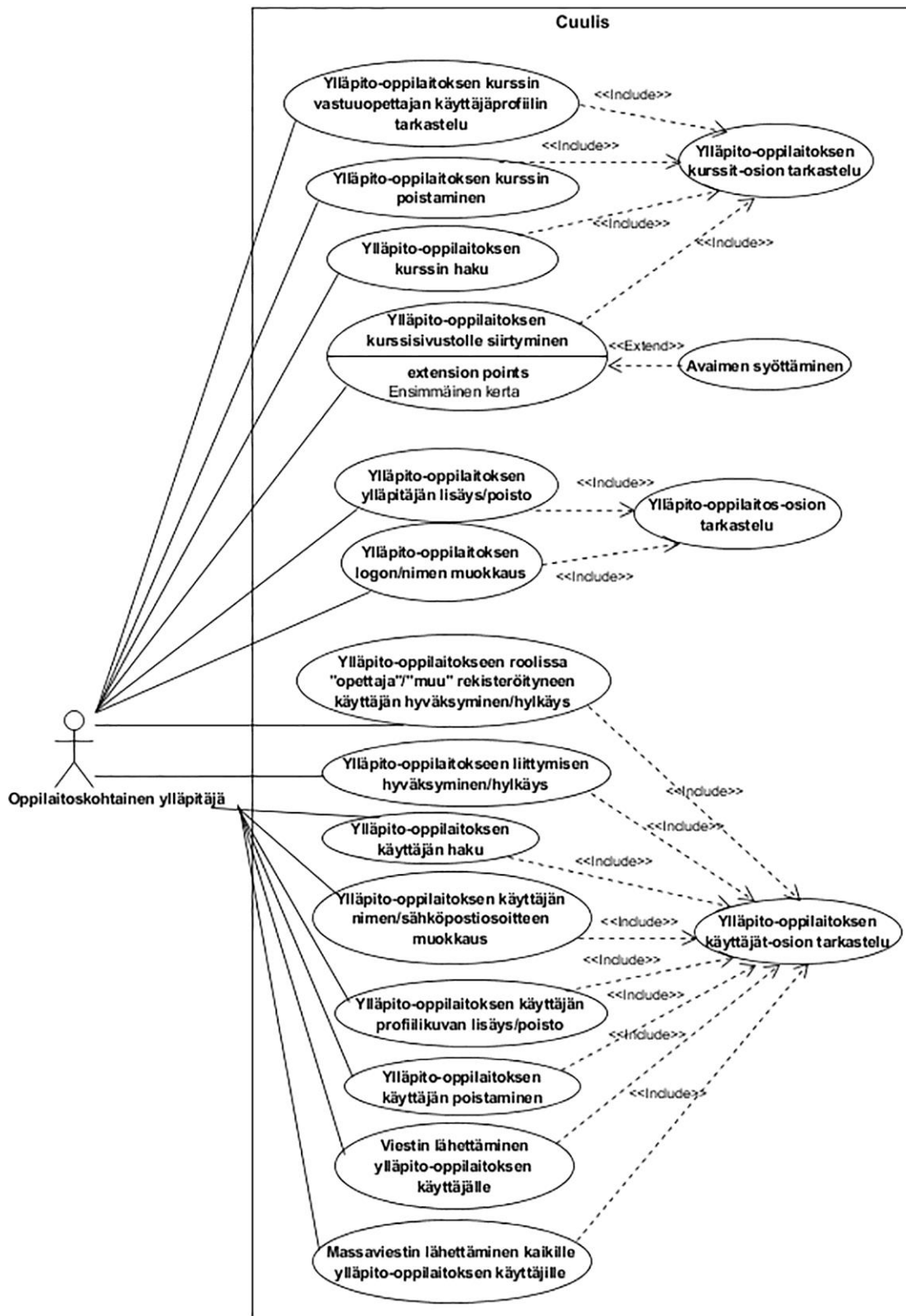
”Oppilaitoskohtainen ylläpitäjä”-käyttäjällä on pääsivustolla vastaavanlaisia toimintoja kuin ”Yleinen ylläpitäjä”-käyttäjällä, mutta vain ylläpitämänsä oppilaitoksen osalta. ”Oppilaitoskohtainen ylläpitäjä”-käyttäjän omat pääsivuston toiminnot jakautuvat näin ollen seuraaviin osioihin:

- Ylläpito-oppilaitoksen käyttäjät
- Ylläpito-oppilaitoksen kurssit
- Ylläpito-oppilaitos

Ylläpito-oppilaitoksen käyttäjät-osiossa ”Oppilaitoskohtainen ylläpitäjä”-käyttäjä voi tarkastella ylläpitämänsä oppilaitoksen käyttäjien käyttäjäprofiileja, muokata niiden sisältämiä tietoja roolia ja oppilaitoksia lukuunottamatta, poistaa heitä oppimisympäristöstä, ja lähettää heille sekä yksilöllisiä viestejä että massaviestejä. Lisäksi tässä osiossa hän voi hyväksyä tai hylätä niiden roolissa ”Opettaja” tai ”Muu” rekisteröityneiden käyttäjien rekisteröitymisen, jotka ovat rekisteröityessään valinneet kyseisen oppilaitoksen ensisijaiseksi oppilaitoksekseen, ja ”Opettaja”- ja ”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjien liittymisen kyseiseen oppilaitokseen.

Ylläpito-oppilaitoksen kurssit-osiossa ”Oppilaitoskohtainen ylläpitäjä”-käyttäjä voi tarkastella kaikkia ylläpitämäänsä oppilaitokseen liittyviä kursseja ja niiden vastuuolettajien käyttäjäprofiileita, sekä poistaa kursseja. Kurssisivustolle siirtyminen vaatii ”Oppilaitoskohtainen ylläpitäjä”-käyttäjältä kuitenkin ensimmäisellä kerralla kurssiin liittyvän nk. avaimen syöttämisen. *Ylläpito-oppilaitos*-osiossa ”Oppilaitoskohtainen ylläpitäjä”-käyttäjä voi puolestaan muokata ylläpitämäänsä oppilaitoksen nimeä ja logoa, sekä lisätä ja poistaa oppilaitoksen ylläpitäjiä.

Kuvan 16 käyttötapauskaaviolla esitellään tarkemmin kaikki ”Oppilaitoskohtainen ylläpitäjä”-käyttäjän yleiset toiminnot. ”Yleinen ylläpitäjä”- ja ”Oppilaitoskohtainen ylläpitäjä”-käyttäjän yhteinen Oppilaitoskohtaisen ylläpitäjän lisääminen-käyttötapaus selostetaan tarkemmin taulukossa 3.



Kuva 16. Käyttötapauskaavio "Oppilaitoskohtainen ylläpitäjä"-käyttäjän pääsivuston toiminnoista.

Taulukko 3. Oppilaitoskohtaisen ylläpitäjän lisääminen-käyttötapausten tarkempi kuvaus.

TARKOITUS:	Oppilaitoskohtaisen ylläpitäjän lisääminen.
KÄYTTÄJÄ:	Kirjautunut käyttäjä, jonka rooli on ”Yleinen ylläpitäjä” tai ”Oppilaitoskohtainen ylläpitäjä”.
ALKUEHDOT:	Haluttu uusi ylläpitäjä on rekisteröitynyt oppimisympäristöön ja liittynyt kyseiseen oppilaitokseen.
KULKU:	1) Käyttäjä valitsee toiminnon ”Oppilaitoskohtaisen ylläpitäjän lisääminen”. 2) Käyttäjä syöttää haluamansa uuden ylläpitäjän rekisteröimän sähköpostiosoitteen.
POIKKEUKSET:	Ylläpitäjän lisääminen ei onnistu: a) Käyttäjän syöttämää sähköpostiosoitetta ei ole rekisteröity. b) Käyttäjän valitsema uusi ylläpitäjä ei ole liittynyt kyseiseen oppilaitokseen.
LOPPUTULOS:	Valittu uusi ylläpitäjä lisätään oppilaitoskohtaiseksi ylläpitäjäksi ja tieto tästä lähetetään hänelle sähköpostiin.

”Opettaja”- ja ”Opiskelija”-käyttäjien omat pääsivuston toiminnot jaetaan puolestaan seuraaviin osioihin:

- Omat kurssit
- Kaikki kurssit

Omat kurssit-osiossa ”Opettaja”-käyttäjä voi lisätä uusia kursseja Cuulis-oppimisympäristöön. Lisätessään uutta kurssia käyttäjä syöttää kurssiin liittyen seuraavat tiedot:

- nimi
- koodi
- onko kurssilla muita opettajia
- avain
- oppilaitos
- lukuvuosi
- aloitus- ja päättymispäivämäärä

Cuulis-oppimisympäristössä kursseilla voi siis olla useita opettajia, mikä mahdollistaa samanaikaisopetuksen. Kurssin oppimisympäristöön lisäävä käyttäjä merkitään automaattisesti sen vastuuohtettajaksi. Mikäli kurssilla on muita opettajia, niiden lopullinen lisääminen (ja myöhemmin poistaminen) tapahtuu kurssisivuston sisältämässä kurssin muokkaustoiminnossa. Kurssin vastuuohtettajaa ei voida myöhemmin muuttaa. Kurssin avain on ikään kuin kurssin kurssisivuston salasana, jolla hallinnoidaan sitä, että kurssisivustolle pääsevät vain ne Cuulis-oppimisympäristön käyttäjät, jotka ovat sinne oikeutettuja. Näin ollen kurssin lisäävän käyttäjän on ilmoitettava kurssin avain kaikille kurssisivustolle oikeutetuille. Avain on myöhemmin nähtävillä ”Opettaja”-käyttäjille kurssisivustolla, mikäli he unohtavat sen.

Käyttäjän Omat kurssit-osiossa on kaikki ne kurssit, joissa käyttäjä on kurssin osallistujana. Näiden kurssien kurssisivustolle käyttäjällä on vapaa pääsy. ”Opettaja”-käyttäjän Omat kurssit-osiossa ovat siis kaikki ne kurssit, joissa hän on vastuuohtettajana, sekä sellaiset kurssit, joihin hänet on lisätty ns. ”muuksi opettajaksi”. ”Opiskelija”-käyttäjän Omat kurssit-osiossa on puolestaan ne kurssit, joihin hän on liittynyt. Omat kurssit-osiossa ”Opiskelija”-käyttäjä voi poistua haluamaltaan kurssilta. ”Opettaja”-käyttäjä voi puolestaan poistaa tässä osiossa niitä kursseja, jotka hän on itse lisännyt, eli joissa hän on vastuuohtettaja. Tällöin kurssi poistuu kokonaan Cuulis-oppimisympäristöstä. Niitä kursseja, joissa ”Opettaja”-käyttäjä on muu kuin vastuuohtettaja, hän ei voi poistaa, mutta hän voi poistua kurssilta ”Opiskelija”-käyttäjän tapaan. Lisäksi Omat kurssit-osiossa on mahdollista tarkastella kunkin kurssin vastuuohtettajan käyttäjäprofiilia.

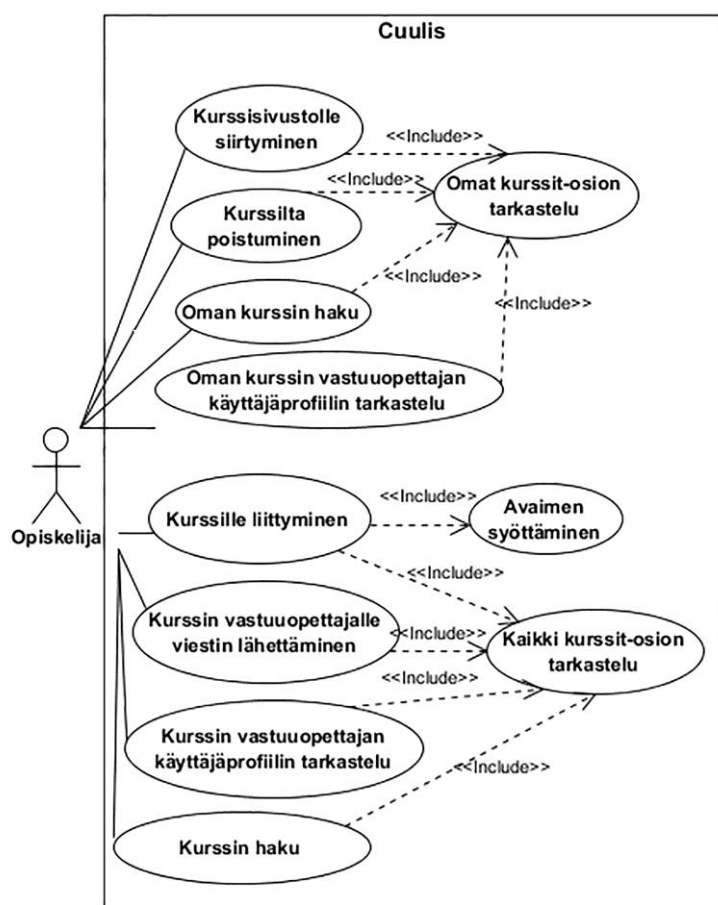
Kaikki kurssit-osiossa sekä ”Opettaja”- että ”Opiskelija”-käyttäjät voivat tarkastella kaikkia sellaisten oppilaitosten kursseja, joihin he ovat liittyneet. Lisäksi he voivat tarkastella kurssien vastuuohtettajien käyttäjäprofiileja ja lähettää heille viestejä, esimerkiksi tiedustellakseen kurssin avainta. Kaikki kurssit-osiossa ”Opettaja”-käyttäjä voi siirtyä haluamansa kurssin kurssisivustolle, mikä vaatii ensimmäisellä kerralla avaimen syöttämisen. Oikean avaimen syöttämisen jälkeen ”Opettaja”-käyttäjä voi siirtyä kyseiselle kurssisivustolle vapaasti, mutta mikäli käyttäjä ei ole kurssin opettaja, niin häntä ei lisätä kurssin osallistujaksi eikä kurssi näin ollen näy käyttäjän Omat kurssit-osiossa. ”Opiskelija”-käyttäjä voi *Kaikki kurssit*-osiossa liittyä haluamalleen kurssille, mikä vaatii avaimen

syöttämisen. Oikean avaimen syöttämisen jälkeen ”Opiskelija”-käyttäjä lisätään kurssin osallistujaksi ja kurssi on myöhemmin nähtävissä hänen Omat kurssit-osiossa.

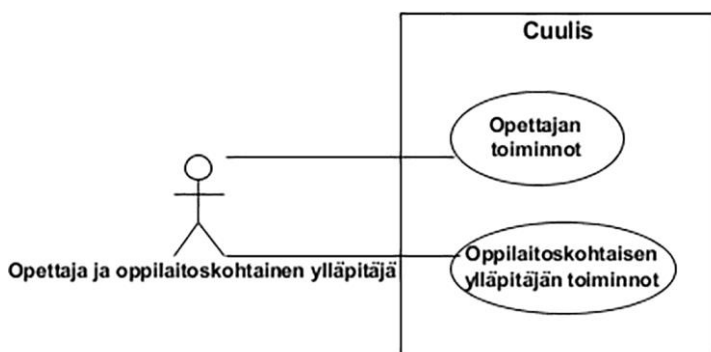
”Opettaja”- ja ”Opiskelija”-käyttäjän pääsivuston toiminnot esitetään kuvien 17 ja 18 käyttötapauskaavioilla. ”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjällä on ”Opettaja”- ja ”Oppilaitoskohtainen ylläpitäjä”-käyttäjän yhdistetyt pääsivuston toiminnot (kuva 19). Näin ollen ”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjällä on kaikki edellä kuvatut ”Opettaja”- ja ”Oppilaitoskohtainen ylläpitäjä”-käyttäjän toiminnot.



Kuva 17. Käyttötapauskaavio ”Opettaja”-käyttäjän pääsivuston toiminnoista Cuulis-oppimisympäristössä.



Kuva 18. Käyttötapauskaavio ”Opiskelija”-käyttäjän pääsivuston toiminnoista Cuulis-oppimisympäristössä.



Kuva 19. Käyttötapauskaavio ”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjän pääsivuston toiminnoista Cuulis-oppimisympäristössä.

Eri käyttäjärooleilla on siis erilaisia toimintoja Cuulis-oppimisympäristön pääsivustolla, joten pääsivusto jakautuu käyttäjäroolista riippuen eri tavoin eri osiin. Seuraavassa esitetään tarkennetusti kaikkien käyttäjäroolien pääsivuston sisältämät osiot.

”Yleinen ylläpitäjä”-käyttäjä:

- Omat tiedot
- Tietoa
- Kirjaudu ulos
- Käyttäjät
- Kurssit
- Oppilaitokset
- Tiedostot

”Oppilaitoskohtainen ylläpitäjä”-käyttäjä:

- Omat tiedot
- Tietoa
- Yhteydenotto yleiseen ylläpitäjään
- Kirjaudu ulos
- Ylläpito-oppilaitoksen käyttäjät
- Ylläpito-oppilaitoksen kurssit
- Ylläpito-oppilaitos

”Opettaja”-käyttäjä:

- Omat tiedot
- Tietoa
- Yhteydenotto ensisijaisen oppilaitoksen ylläpitäjään
- Kirjaudu ulos
- Omat kurssit (ne, joissa käyttäjä vastuuopettaja tai muu opettaja)
- Kaikki kurssit (niissä oppilaitoksissa, johon käyttäjä on liittynyt)

”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjä:

- Omat tiedot
- Tietoa
- Yhteydenotto yleiseen ylläpitäjään
- Kirjaudu ulos
- Omat kurssit (ne, joissa käyttäjä vastuuopettaja tai muu opettaja)
- Kaikki kurssit (niissä oppilaitoksissa, johon käyttäjä on liittynyt)
- Ylläpito-oppilaitoksen käyttäjät
- Ylläpito-oppilaitoksen kurssit
- Ylläpito-oppilaitos

”Opiskelija”-käyttäjä:

- Omat tiedot
- Tietoa oppimisympäristöstä
- Yhteydenotto ensisijaisen oppilaitoksen ylläpitäjään
- Kirjaudu ulos
- Omat kurssit (ne kurssit, joiden kurssisivustolle käyttäjä on liittynyt Kaikki kurssit-osiosta)
- Kaikki kurssit (niissä oppilaitoksissa, johon käyttäjä on liittynyt)

Kuten edellä esitetyistä käyttötapauskaavioista (kuvat 15-18) ilmenee, kaikkien käyttäjien, kurssien ja oppilaitosten tarkastelun yhteydessä on ns. hakutoiminto, jolla helpotetaan halutun kohteen löytämistä.

4.1.3 Kurssisivuston toiminnot

Cuulis-oppimisympäristön kurssisivusto sisältää varsinaiset opetukseen liittyvät toiminnot. Kurssisivustolle siirtyminen vaatii siis muilta kuin kurssin lisänneeltä käyttäjältä ja ”Yleinen ylläpitäjä”-käyttäjältä ensimmäisellä kerralla oikean avaimen syöttämisen.

Kurssisivustolle päästyään kaikki paitsi ”Opiskelija”-käyttäjät saavat ”Opettaja”-käyttäjän roolin. Cuulis-oppimisympäristön kurssisivuston toiminnot jakautuvat seuraaviin osiin:

- Aloitus
- Osallistujat
- Materiaalipankki
- Viestin lähetys
- Kurssin tietojen muokkaus (vain ”Opettaja”-käyttäjällä)
- Projektit
- Luentosessio
- Äänestykset
- Keskustelu

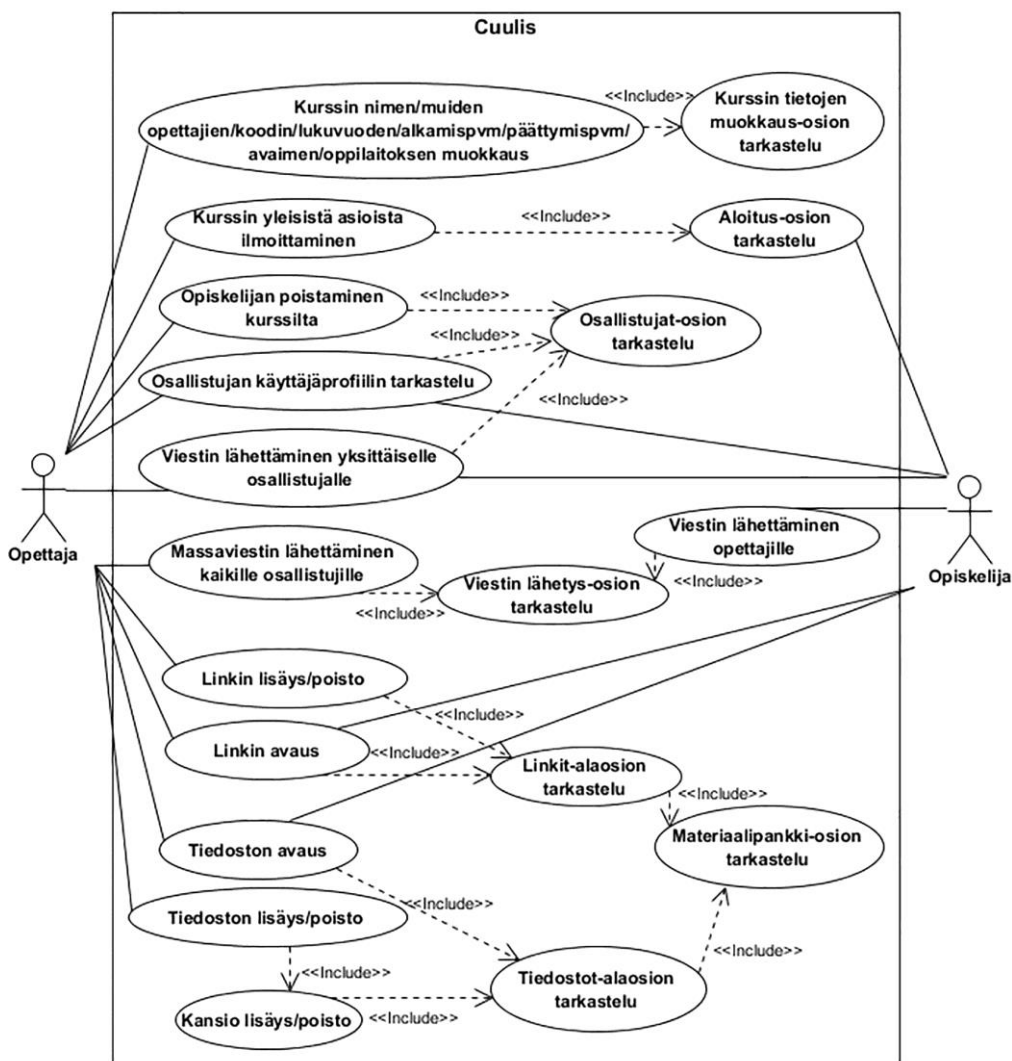
Aloitus-osio on kurssisivuston ns. etusivu, johon ”Opettaja”-käyttäjä voi kirjoittaa kurssiin liittyviä yleisiä ilmoitusasioita. Lisäksi tässä osiossa ”Opettaja”-käyttäjälle on näkyvissä kurssin avain.

Osallistujat-osio sisältää luettelon kurssin vastuuopettajasta, mahdollisista muista opettajista ja opiskelijoista. Tässä osiossa käyttäjät voivat tarkastella kurssin osallistujien käyttäjäprofiileja ja lähettää heille viestejä. Lisäksi ”Opettaja”-käyttäjä voi tässä osiossa poistaa kurssille liittyneitä opiskelijoita.

Viestin lähetys-osiossa ”Opettaja”-käyttäjä voi lähettää viestejä yhtäaikaaisesti kaikille kurssin osallistujille, eli opiskelijoille sekä mahdollisille muille opettajille. Mikäli ”Opettaja”-käyttäjä ei ole kurssin opettaja, viesti lähtee opiskelijoiden lisäksi kaikille kurssin opettajille. ”Opiskelija”-käyttäjän tässä osiossa lähettämä viesti lähtee vain kurssin opettajille.

Kurssin tietojen muokkaus-osio on esillä vain ”Opettaja”-käyttäjille ja tässä osiossa he voivat muokata kurssitiedoista kaikkia kurssin lisäämisen yhteydessä syötettyjä tietoja.

Lisäksi osiossa voi lisätä ja poistaa kurssin muita opettajia, jos kurssin lisäämisen yhteydessä valittiin, että kurssilla on muitakin opettajia. Tätä valintaa ei Kurssin tietojen muokaus-osiossa voi kuitenkaan enää muuttaa.

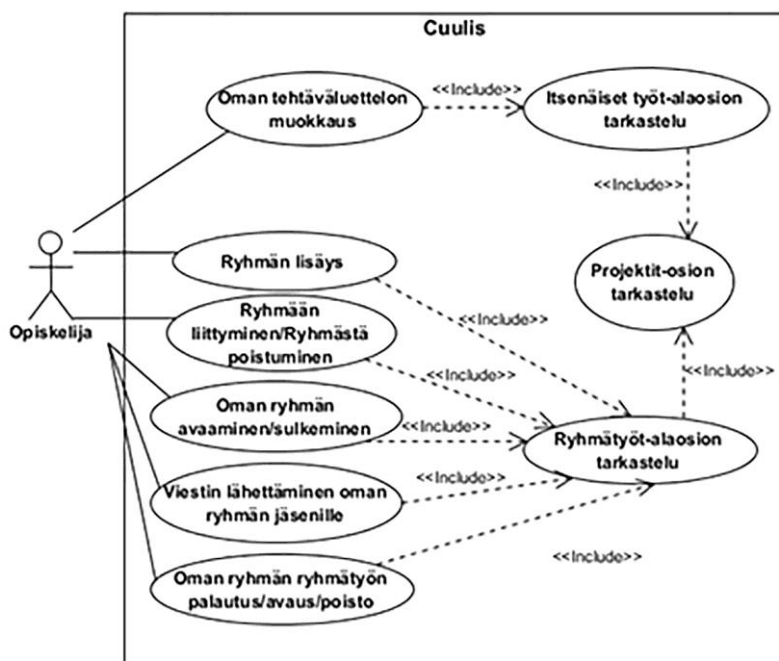


Kuva 20. Käyttötapauskaavio Aloitus-, Osallistujat-, Viestin lähetyksen-, Kurssin tietojen muokkaus- ja Materiaalipankki-osioihin liittyvistä toiminnoista.

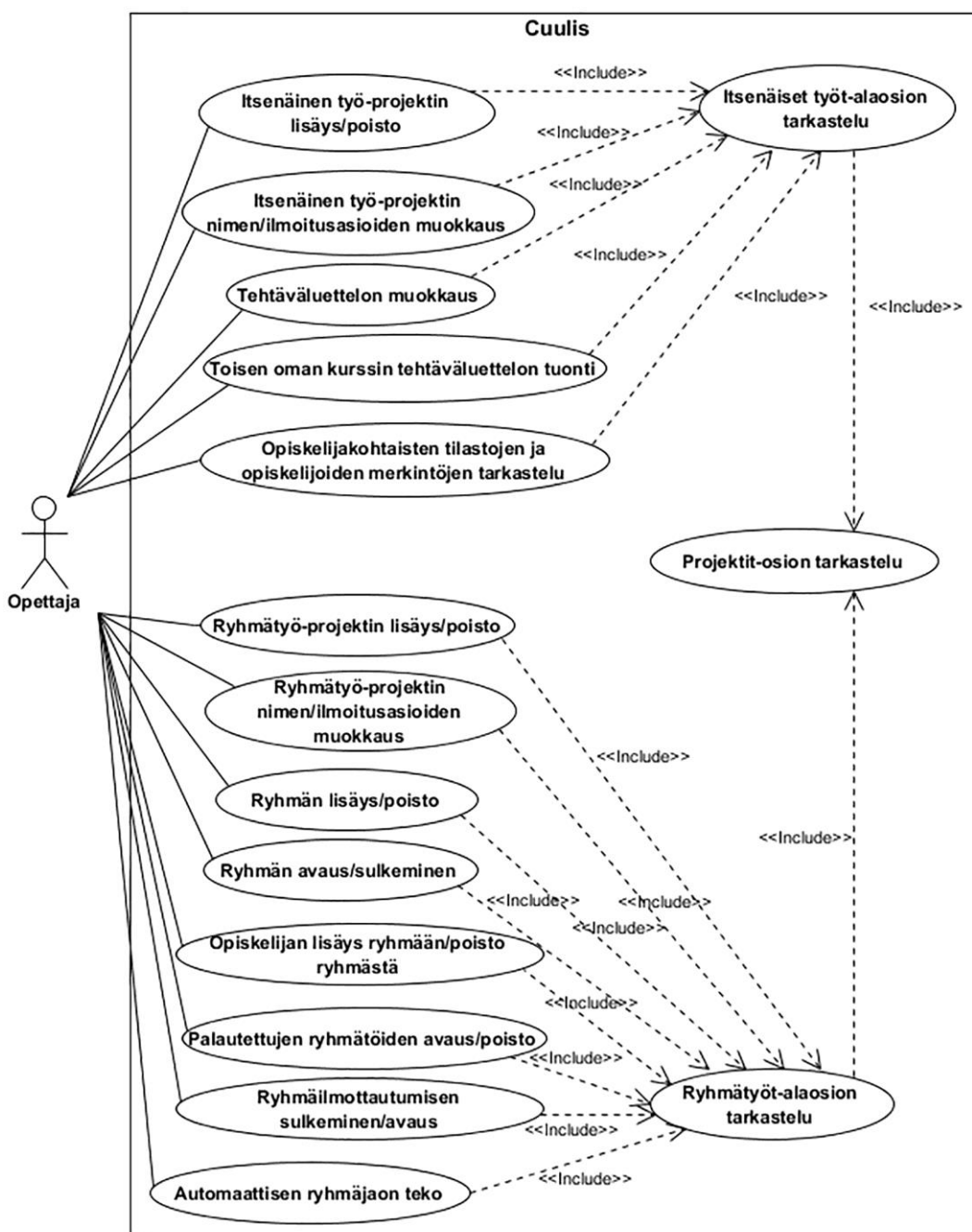
Materiaalipankki-osio sisältää *Tiedostot*- ja *Linkit*-alaosiot. *Tiedostot*-alaosiossa on ”Opettaja”-käyttäjän lisäämiä tiedostoja. Tiedostoja lisätään siten, että samaan asiakoko-

naisuuteen liittyvät tiedostot kootaan omaan kansioon (esim. ”Tuntimuistiinpanot”-kansio). Myöhemmin ”Opettaja”-käyttäjä voi poistaa sekä yksittäisiä tiedostoja että kokonaisen kansion. Linkit-alaosiossa on puolestaan ”Opettaja”-käyttäjän lisäämiä linkkejä, joita ”Opettaja”-käyttäjä voi myös poistaa. Sekä tiedostojen että linkkien lisäämisen yhteydessä ”Opettaja”-käyttäjä voi syöttää lyhyen kuvauksen niihin liittyen. Kuvan 20 käyttötapauskaaviolla hahmotetaan näitä Aloitus-, Osallistujat-, Viestin lähetyksen, Kurssin tietojen muokkauksen ja Materiaalipankki-osioiden sisältämiä kurssisivuston toimintoja sekä ”Opettaja”- että ”Opiskelija”-käyttäjän osalta.

Projektit-osio sisältää kaksi alaosiota. *Itsenäiset työt*-alaosio sisältää itsenäisiin töihin liittyviä projekteja ja *Ryhmätyöt*-alaosio ryhmätöihin liittyviä projekteja, joita ”Opettaja”-käyttäjä voi lisätä ja poistaa. Sekä Itsenäinen työ- että Ryhmätyö-projekteilla on lisäksi oma ilmoitusosionsa, johon ”Opettaja”-käyttäjä voi kirjoittaa kyseiseen projektiin liittyviä ilmoitusasioita. Molempia projektityyppejä voidaan lisätä rajaton määrä. ”Opiskelija”-käyttäjän toiminnot Projektit-osiossa esitetään kuvan 21 käyttötapauskaaviolla, ja ”Opettaja”-käyttäjän toiminnot kuvan 22 käyttötapauskaaviolla.



Kuva 21. Käyttötapauskaavio Projektit-osion toiminnoista ”Opiskelija”-käyttäjän osalta.



Kuva 22. Käyttötapauskaavio Projektit-osion toiminnoista ”Opettaja”-käyttäjän osalta.

Uutta Itsenäinen työ-projektia lisätessään ”Opettaja”-käyttäjä syöttää vain projektin nimen, jota ”Opettaja”-käyttäjä voi muokata myöhemmin. Itsenäinen työ-projekti sisältää automaattisesti kurssin suorittamiseen liittyvän tehtäväluettelon, jota ”Opettaja”-käyttäjä voi muokata lisäämällä siihen tehtäviä ja jaotteleamalla niitä haluamisensa otsikoiden alle.

”Opettaja”-käyttäjä voi myös tuoda toisen oman kurssin yhteydessä luomansa tehtäväluettelon osaksi kyseistä tehtäväluetteloa. ”Opiskelija”-käyttäjät voivat merkitä omaan tehtäväluetteloonsa kunkin tehtävän kohdalle osasivatko he tehtävän ja haluavatko he, että tehtävä käydään yhdessä tunnilla läpi. Lisäksi he voivat lisätä kommentteja kunkin tehtävän kohdalle. ”Opiskelija”-käyttäjän tehtäväluettelon yhteydessä on merkintöjen perusteella reaaliaikaisesti päivittyvä sektoridiagrammi, jonka avulla käyttäjä hahmottaa etenemistään tehtävissä. Sektoridiagrammissa kuvastetaan osattujen ja yritettyjen, mutta ei osattujen, sekä tekemättä olevien tehtävien osuus kaikista tehtävistä.

”Opettaja”-käyttäjä voi tarkastella seuraavia tehtäväluettelossa oleviin tehtäviin liittyviä tietoja:

- kuinka monta ja ketkä ovat osanneet tehtävän
- kuinka monta ja ketkä ovat yrittäneet tehdä tehtävää, mutta eivät ole osanneet sitä
- kuinka monta ja ketkä ovat toivoneet, että tehtävä käytäisiin yhdessä läpi
- kuinka moni ja ketkä ovat kommentoineet tehtävää, ja millä tavoin

”Opettaja”-käyttäjä voi myös tarkastella opiskelijakohtaisesti, kuinka monta prosenttia kaikista tehtävistä opiskelija on tehnyt, osannut ja yrittänyt, muttei osannut. Lisäksi ”Opettaja”-käyttäjä voi tarkastella yksityiskohtaisemmin kunkin opiskelijan tehtäväluettelon sisältöä ja siihen liittyvää sektoridiagrammia.

Ryhmätyöt-alaosio sisältää toiminnon, jolla voidaan hallita opiskelijoiden jakamista/jakautumista ryhmiin kurssin ryhmätöihin liittyen. Lisätessään Ryhmätyö-projektia ”Opettaja”-käyttäjä syöttää siihen liittyen seuraavat tiedot:

- nimi
- ryhmien maksimimäärä
- ryhmäkohtainen minimimäärä
- ryhmäkohtainen maksimimäärä
- sallitaanko opiskelijoille mahdollisuus palauttaa sivulle projektiin liittyvä tiedosto

Myöhemmin ”Opettaja”-käyttäjä voi muokata näistä tiedoista vain nimeä. Heti uuden Ryhmätyö-projektin lisäämisen jälkeen ryhmiä on tarjolla vain yksi. Käyttäjät voivat lisätä uusia ryhmiä niin kauan, kunnes ryhmäilmoittautuminen on auki ja ryhmien maksimimäärä ei ole tullut täyteen. ”Opiskelija”-käyttäjä voi liittyä haluamaansa ryhmään niin kauan kun ryhmäilmoittautuminen on auki, ryhmän maksimimäärä ei ole täyttynyt, eikä hän ole vielä missään ryhmässä. Ennen ryhmäilmoittautumisen sulkemista ”Opiskelija”-käyttäjä voi sulkea oman ryhmän, jos ryhmäkohtainen minimimäärä on ylittynyt, sekä aukaista suljetun ryhmän. Lisäksi hän voi lähettää viestin ryhmän muille jäsenille sekä poistua kyseisestä ryhmästä. Mikäli ryhmätyötiedostojen palautus on sallittu, ”Opiskelija”-käyttäjä voi palauttaa ryhmänsä ryhmätyötiedoston ryhmäilmoittautumisen sulkemisen jälkeen, sekä avata ja poistaa oman ryhmänsä tiedoston. Vain yksi ryhmäkohtainen tiedosto voidaan palauttaa.

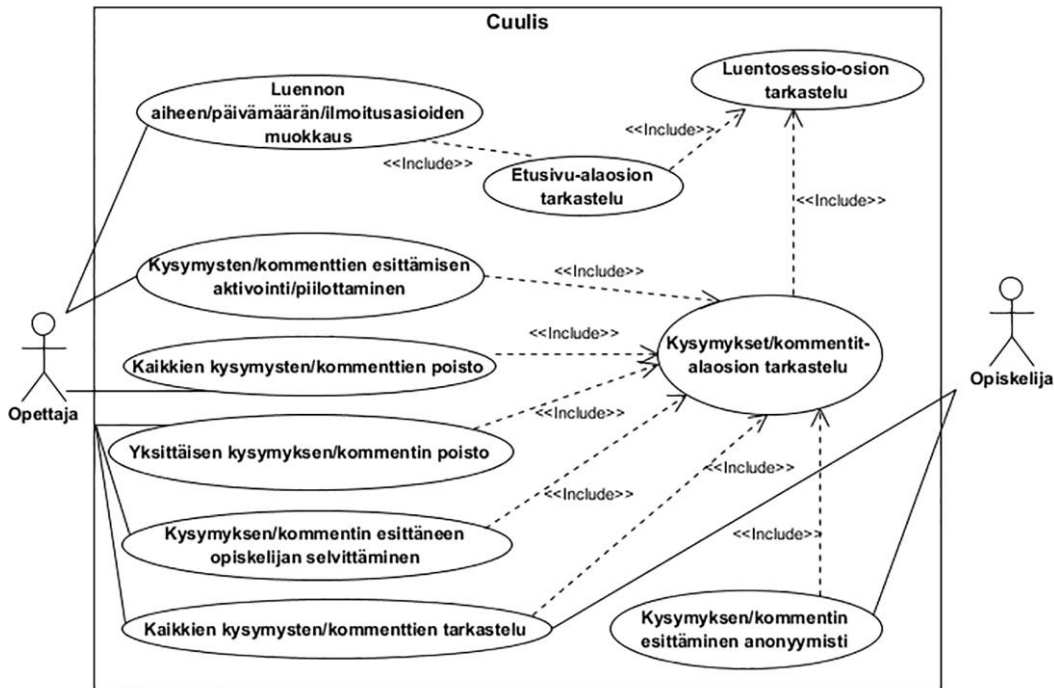
”Opettaja”-käyttäjä voi ryhmien lisäämisen lisäksi poistaa ryhmiä, avata kaikkia suljettuja ryhmiä sekä sulkea kaikki sellaiset ryhmät, joissa minimimäärä on ylittynyt. Lisäksi ”Opettaja”-käyttäjä voi liittää ilman ryhmää olevia opiskelijoita sellaisiin ryhmiin, joissa maksimimäärä ei ole täyttynyt, sekä poistaa opiskelijoita ryhmistä. ”Opettaja”-käyttäjä voi myös arpoa kaikki vailla ryhmää olevat opiskelijat ryhmiin, sekä suorittaa automaattisen ryhmäjaon. ”Opettaja”-käyttäjä voi halutessaan sulkea ryhmäilmoittautumisen ja myöhemmin avata sen uudelleen. Mikäli ryhmätyötiedostojen palautus on sallittu, ”Opettaja”-käyttäjä näkee ryhmäilmoittautumisen sulkemisen jälkeen kaikkien ryhmien palauttamien ryhmätyötiedostot, ja voi avata sekä poistaa niitä.

Itsenäinen työ-projektin sisältämän tehtäväluettelon avulla voidaan tukea aktivoivaa opetusta ja yksilöllistä oppimista muun muassa siten, että luennon aikana opiskelijat voivat itsenäisesti omaan tahtiin tehdä kurssin suorittamiseen liittyviä tehtäviä ja samalla he voivat kirjata etenemistään käytössä olevilla laitteillaan. Kurssin opettaja voi seurata reaaliaikaisesti sitä, miten opiskelijat suoriutuvat tehtävien tekemisessä ja kohdistaa tukensa sitä eniten kaipaaville. Lisäksi opettaja näkee, mitkä tehtävistä ovat osoittautuneet yleisimmin hankalimmiksi, jolloin nämä tehtävät voidaan käydä yhdessä läpi. Opiskelijoiden tekemien merkintöjen perusteella päivittyvän sektoridiagrammin avulla opiskelijat hah-

mottavat visuaalisesti suoriutumistaan tehtävissä ja voivat näin seurata omaa oppimisprosessiaan. Tämä visuaalinen hahmottaminen voi myös herättää opiskelijoiden opiskelumotivaatiota. Mikäli kuitenkin osoittautuu, että sektoridiagrammin ja muiden tehtävien tekemiseen liittyvien tilastotietojen näkeminen osoittautuu enemmän negatiiviseksi, kenties ahdistavaksi tekijäksi, voidaan tämä piilottaa joko yksittäisiltä tai kaikilta kurssin opiskelijoilta.

Itsenäinen työ-projekti sopii myös tavanomaiseen kurssin itsenäiseen suorittamiseen, missä opiskelija suorittaa kurssin kokonaan itsenäisesti oppilaitoksen ulkopuolella. Ryhmätyö-projektilla voidaan Cuulis-oppimisympäristön alkuvaiheessa hallita vain opiskelijoiden jakautumista ryhmiin kurssiin liittyvien ryhmätöiden osalta. Tähän on kuitenkin myöhemmin mahdollista lisätä uusia toimintoja sen mukaan, miten oppimisympäristön käyttöönoton jälkeen tarvetta ja uusia ideoita ilmaantuu. Materiaalipankki-osiota voidaan hyödyntää Projektit-osion yhteydessä lisäämällä Tiedostot- ja Linkit-alaosioihin Itsenäinen työ- ja Ryhmätyö-projekteihin liittyvää materiaalia.

Luentosessio-osio on nimensä mukaisesti luentokohtainen osio, joka sisältää kaksi alaosiota. *Etusivu*-alaosio on luentoan liittyvä etusivu, jossa on nähtävillä luennon aihe, päivämäärä ja luentoan liittyviä ilmoitusasioita, joita ”Opettaja”-käyttäjä voi muokata. *Kysymykset/kommentit*-alaosiossa ”Opiskelija”-käyttäjät voivat esittää anonyymisti kysymyksiä tai kommentteja siten, että ne ovat näkyvissä kaikille kurssin osallistujille. Kysymyksen tai kommentin yhteydessä näkyy myös ajankohta, jolloin se on esitetty, ja ne esitetään siten, että viimeisin lähetetty kysymys/kommentti on luettelon ylimpänä. Vaikka kysymyksiä ja kommentteja on mahdollista esittää nimettömästi, ”Opettaja”-käyttäjällä on mahdollisuus selvittää, kuka tietyn kysymyksen/kommentin on esittänyt. Lisäksi ”Opettaja”-käyttäjä voi poistaa yksittäisiä kysymyksiä/kommentteja, tai kerralla kaikki. Kysymykset/kommentit-alaosion toiminto on oletusarvoisesti piilotettu, mutta ”Opettaja”-voi halutessaan aktivoida sen, sekä myöhemmin piilottaa sen uudestaan. Kuvan 23 käyttötapauskaaviolla esitetään kaikki Luentosessio-osion toiminnot sekä ”Opettaja”- että ”Opiskelija”-käyttäjän osalta.

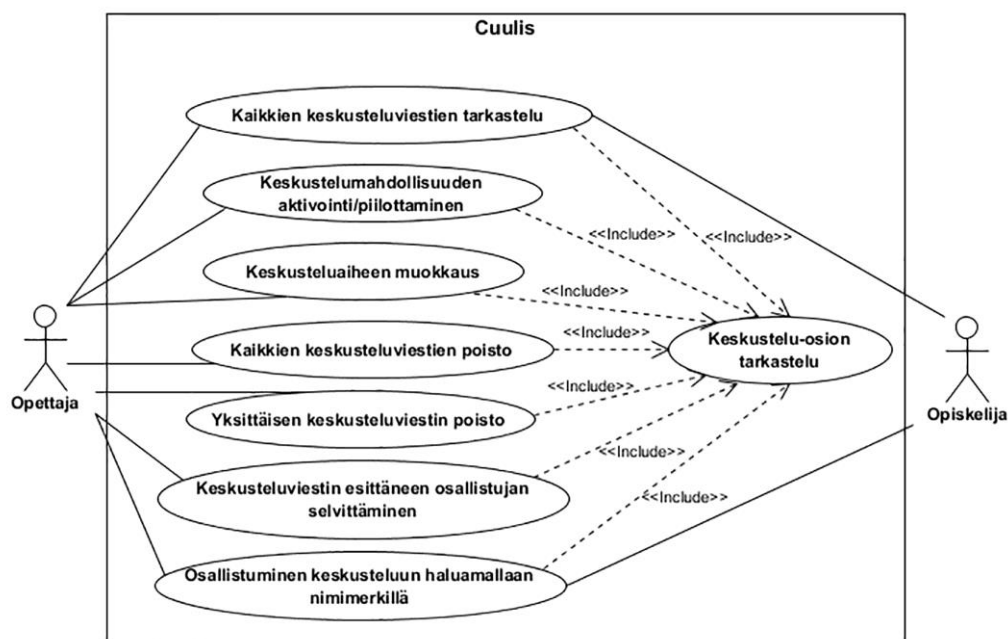


Kuva 23. Käyttötapauskaavio Luentosessio-osion toiminnoista.

Luentosessio-osion kysymysten/kommenttien esittämiseen liittyvää toimintoa voidaan hyödyntää muun muassa Lonkan & Lonkan esittelemissä aktivoiva luento- ja porinaryh-mät-opetusmenetelmissä. Opiskelijat voivat esittää kysymyksiä ja kommentteja luennon aikana käytössään olevilla laitteilla, kuten älypuhelimilla, tableteilla ja tietokoneilla. Opettaja voi seurata esitettyjä kysymyksiä ja kommentteja luennon aikana omalla laitteellaan ja puuttua niihin opetuksen aikana haluamansa mukaan. Toiminto soveltuu siis erinomaisesti sellaisiin opetusmenetelmiin, joissa opetus on suurimmaksi osaksi opettajakeskeistä siten, että suurin osa luennosta perustuu opettajan äänessä olemiseen. Reaaliaikaisen kysymysten ja kommenttien esittämisen mahdollisuuden avulla opiskelijoita aktivoidaan ottamaan osaa opetuksen etenemiseen, jotta he eivät olisi vain passiivisia kuuntelijoita. Etusivu-alaosiota voidaan puolestaan hyödyntää muun muassa Virittävät kysymykset-opetusmenetelmässä. Myöhemmin siihen voidaan lisätä myös mahdollisuus upottaa tiedostoja, kuvia ja videoita tai muuta luento- ja opetusliittävää materiaalia. Opettaja voi esimerkiksi halutessaan videoida luennon ja upottaa tämän Etusivu-alaosioon. Opiskelijat

voivat seurata etänä samanaikaisesti toisessa näkymässä videoitua luentoa ja toisessa näkymässä olevan Kysymykset/kommentit-alaosion sisältöä sekä lähettää omia kysymyksiä ja kommentteja.

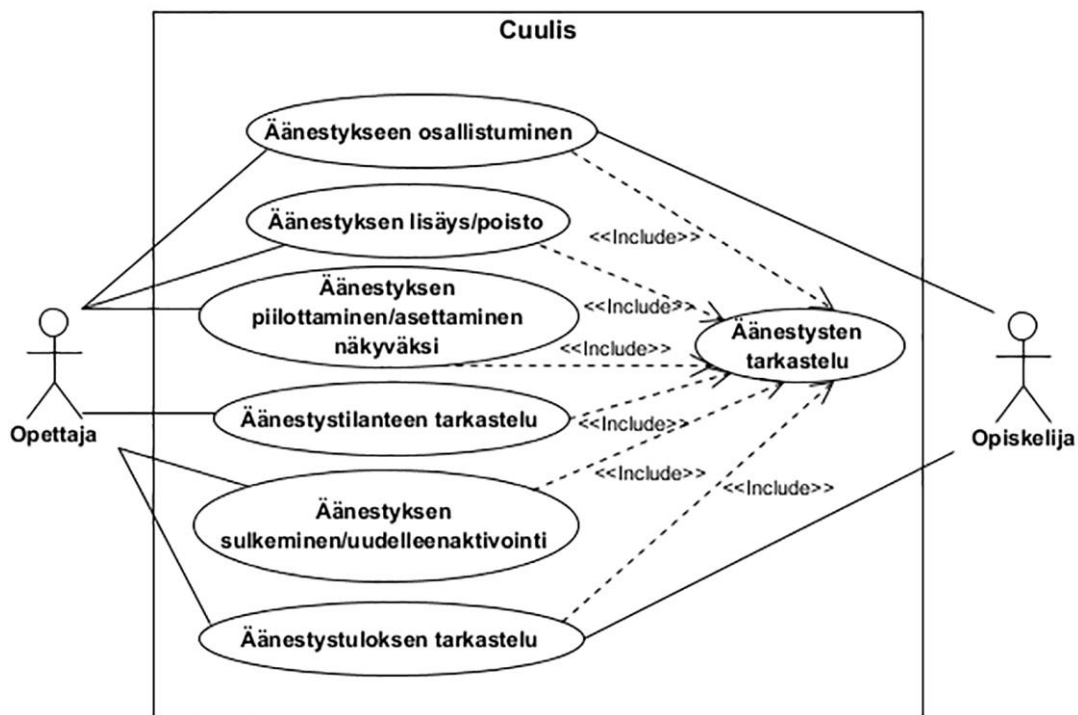
Keskustelu-osiossa sekä ”Opettaja”- että ”Opiskelija”-käyttäjät voivat lähettää keskusteluviestejä, jotka näkyvät kaikille kurssin osallistujille. Keskusteluviestin voi lähettää haluamallaan nimimerkillä. Viestin yhteydessä kaikille käyttäjille näkyy sen lähettäneen käyttäjän nimimerkki ja ajankohta, jolloin viesti on lähetetty. Viestit esitetään siten, että viimeisin lähetetty viesti on luettelon ylimpänä. Vaikka viestejä voi lähettää sekä nimettömästi että haluamalla nimimerkillä, ”Opettaja”-käyttäjällä on kuitenkin mahdollisuus selvittää viestin todellinen lähettäjä. ”Opettaja”-käyttäjä voi myös poistaa sekä yksittäisiä keskusteluviestejä että kaikki viestit kerrallaan. Keskustelu-osion sisältämä toiminto on oletusarvoisesti piilotettu, ja ”Opettaja”-käyttäjä voi halutessaan sekä aktivoida että myöhemmin piilottaa sen. Aktivoinnin yhteydessä ”Opettaja”-käyttäjä voi syöttää keskustelun aiheen ja myöhemmin muokata sitä. Kuvan 24 käyttötapauskaaviolla esitetään kaikki Keskustelu-osion toiminnot roolikohtaisesti.



Kuva 24. Käyttötapauskaavio Keskustelu-osion toiminnoista.

Keskustelu-osion viestittelyä voidaan käydä joko yleisellä tasolla tai opettajan määrittelemän aihealueen puitteissa ja keskusteluviestit voidaan säilyttää koko kurssin ajan. Viestittely voi jatkua myös kotona tai muualla oppilaitoksen ulkopuolella sitä mukaan, kun asioita tulee mieleen. Sekä Keskustelu-osion että Luentosessio-osion toiminnallisuuksien avulla opiskelijoita voidaan aktivoida ja osallistaa koko kurssin ajan. Opiskelijat voivat esittää mieleen tulevia kysymyksiä, kommentteja, mielipiteitä, ehdotuksia jne. He voivat myös vastata opettajan ja muiden opiskelijoiden esittämiin kysymyksiin sekä kommentoida muiden esille tuomia ajatuksia. Lisäksi he voivat tuoda esille omaa tietämystään kurssin asioista, ja tällä tavoin antaa oman panoksensa kurssin opetukseen. Nimettömyysmahdollisuuden avulla arimpiakin opiskelijoita pyritään kannustamaan esittämään mielessä olevia kysymyksiä ja muita ajatuksia. Koska opiskelijat näkevät toistensa esittämät viestit, niin he voivat huomata, etteivät ole ainoita, joille asiat ovat epäselviä, ja toisaalta muiden ajatukset voivat avartaa opiskelijan omaa ajattelua.

Äänestykset-osiossa ”Opettaja”-käyttäjä voi lisätä erilaisia monivalintatyyppejä äänestyksiä, joihin kaikki kurssin osallistujat voivat osallistua. Uutta äänestystä lisätessään ”Opettaja”-käyttäjä syöttää äänestykseen liittyvän kysymyksen ja siihen liittyvät vastausvaihtoehdot. Osallistuessaan äänestykseen kurssin osallistuja voi valita vastausvaihtoehdoista vain yhden. Omaa äänestysvastausta voi muuttaa niin kauan kuin äänestys on auki, mutta vain viimeisin annettu vastaus rekisteröityy. ”Opettaja”-käyttäjä voi piilottaa äänestyksen siten, etteivät ”Opiskelija”-käyttäjät näe sitä, ja halutessaan taas asettaa äänestyksen näkyväksi. ”Opettaja”-käyttäjällä on myös mahdollisuus tarkastella reaaliaikaista äänestystilannetta ja sulkea halutessaan äänestys siten, ettei siihen ole enää mahdollista vastata. Äänestyksen sulkemisen jälkeen ”Opettaja”-käyttäjä näkee lopullisen äänestystuloksen, ja jos äänestys on asetettu näkyväksi, tulokset näkyvät sulkemisen jälkeen myös ”Opiskelija”-käyttäjille. ”Opettaja”-käyttäjä voi halutessaan myös aktivoida suljetun äänestyksen uudelleen siten, että siihen voidaan taas osallistua. ”Opettaja”-käyttäjällä on mahdollisuus myös poistaa luotuja äänestyksiä, mutta äänestysten lukumäärää ei ole rajoitettu. Kuvan 25 käyttötapauskaavio hahmottaa *Äänestykset*-osion sisältämiä toimintoja roolikohtaisesti.



Kuva 25. Käyttötapauskaavio Äänestykset-osion toiminnoista.

Äänestykset-osion äänestykset voivat liittyä moniin eri asioihin. Niitä voidaan hyödyntää muun muassa opiskelijoiden aktivoimiseen ja osallistamiseen kurssin liittyvässä päätöksenteossa. Äänestyskysymyksenä voi esimerkiksi olla ”Kuinka monta koetta kurssin aikana järjestetään?”, näin opiskelijat voidaan ottaa mukaan kurssin suunnitteluun. Äänestyksistä voidaan rakentaa myös tietynlainen kurssin arviointimenetelmä siten, että opiskelijoiden vastausten perusteella kurssia voidaan kehittää sekä kurssin aikana että sen päättymisen jälkeen.

Cuulis-oppimisympäristön kurssisivuston sisältämien toiminnallisuuksien avulla voidaan tukea nykyaikaiseen oppimiskäsitykseen ja aktivoivaan opetukseen liittyvää näkemystä opiskelijan aktiivisesta tiedon konstruomisesta sekä vuorovaikutuksen ja yhteistoiminnallisuuden roolista tässä prosessissa.

4.1.4 Viestiliikennetoiminnot

Cuulis-oppimisympäristön käyttäjät voivat lähettää eri yhteyksissä toisilleen viestejä, kuten edellä esitettyjen etusivun, pääsivuston ja kurssisivuston sisältämien toimintojen esitelystä tuli esille. Käyttäjän lähettämä viesti lähetetään viestin vastaanottavan käyttäjän rekisteröimään sähköpostiosoitteeseen. Lähetettyyn sähköpostiin liitetään lähettäjäksi ja vastaussähköpostiosoitteeksi automaattisesti viestin lähettäjän tiedot, joita viestiä lähettävä käyttäjä ei voi muuttaa. Poikkeuksena tähän on oppimisympäristön etusivulla oleva osio, jossa voi lähettää viestin Cuulis-oppimisympäristön yleiselle ylläpitäjälle. Osiossa viestin lähettäjä syöttää nimensä, sähköpostiosoitteensa ja puhelinnumeron. Käyttäjä voi myös valita, haluaako viestiin vastauksen, ja jos haluaa, niin haluaako sen puhelimitse vai sähköpostitse.

Lisäksi seuraavien toimintojen yhteydessä Cuulis-oppimisympäristö lähettää automaattisesti viestejä, joihin ei ole mahdollista vastata:

- Rekisteröityminen (vahvistuslinkki/vahvistusilmoitus rekisteröityneelle käyttäjälle, ilmoitus valitun ensisijaisen oppilaitoksen ylläpitäjälle)
- Rekisteröitymisen hyväksyminen/hylkääminen (vahvistuslinkki/ilmoitus hylkähäyksestä rekisteröityneelle käyttäjälle)
- Oppilaitokseen liittyminen (ilmoitus kyseisen oppilaitoksen ylläpitäjälle)
- Oppilaitokseen liittymisen hyväksyminen/hylkääminen (ilmoitus oppilaitokseen liittyneelle käyttäjälle)
- Unohtuneen salasanan tiedustelu (uudelleenaktivointilinkki käyttäjälle)
- Käyttäjän poistaminen oppimisympäristöstä/kurssilta (ilmoitus poistetulle käyttäjälle)
- Oppilaitoskohtaisen ylläpitäjän lisäys/poisto (ilmoitus ylläpitäjäksi lisätylle/poistetulle käyttäjälle)

4.2 Ei-toiminnalliset vaatimukset

Monet Cuulis-oppimisympäristöön liittyvät ei-toiminnalliset vaatimukset sisältyvät edellä esitettyihin toiminnallisiin vaatimuksiin, sillä rajanvetoa näiden kahden eri vaatimustyyppin välillä on haastava tehdä. Seuraavassa esitetään selkeämmin nimenomaan ei-toiminnallisiin vaatimuksiin liittyvät Cuulis-oppimisympäristön käytettävyy-, kehitys-, turvallisuus- sekä käyttövarmuus ja saatavuusvaatimukset.

4.2.1 Käytettävyystvaatimukset

Cuulis-oppimisympäristön käyttöliittymä suunnitellaan siten, että se on helposti opittava ja helppokäyttöinen. Tavoitteena on, että Cuulis-oppimisympäristö on helppo ottaa käyttöön eikä erillistä käyttöohjetta tarvitse laatia. Käyttöliittymäsuunnittelussa on myös huomioitava, että Cuulis-oppimisympäristöä käytetään erikokoisilla laitteilla, kuten tietokoneilla, älypuhelimilla ja tableteilla.

4.2.2 Kehitysvaatimukset

Cuulis-oppimisympäristön toteutukseen ja käyttöön liittyvä tietoaineisto tallennetaan tietokantaan. Kaikki käyttäjien oppimisympäristöön lisäämät kuvat (oppilaitoksen logot ja profiilikuvat) sekä tiedostot (kurseihin liittyvät yleiset tiedostot ja opiskelijoiden palauttamat ryhmätyöt) tallennetaan palvelimelle ja niiden sallittu enimmäiskoko on 5 MB. Lisäksi sallitut päätteet ovat kuvatiedostojen osalta .jpg, .gif, .png, .jpeg ja muiden tiedostojen osalta .txt, .pdf, .doc, .docx, .rtf, .dat, .pptx, .ppt, .xls, .xlsx.

4.2.3 Turvallisuusvaatimukset

Cuulis-oppimisympäristö ei tule sisältämään mitään arkaluontoista ja henkilökohtaista tietoa. Käyttäjistä tullaan rekisteröimään ainoastaan nimi ja sähköpostiosoite, joka tulee toimimaan kirjautumisen yhteydessä käyttäjätunnuksena ja johon verkko-oppimisympäristön viestiliikenne tullaan ohjaamaan. Käyttäjien valitsemat salasanat salataan ennen

niiden viemistä tietokantaan siten, että tietokannan sisältöön käsiksi pääsevät henkilöt eivät tunnista näitä salasanoja. Tietokannan sisältö ja kaikki palvelimella olevat tiedostot varmuuskopioidaan säännöllisesti. Tietokannan rakenne ja muut Cuulis-oppimisympäristön turvallisuuteen liittyvät asiat pidetään salassa.

4.2.4 Käyttövarmuus- ja saatavuusvaatimukset

Cuulis-oppimisympäristön tulee olla käytettävissä 24 tuntia vuorokaudessa vuoden jokaisena päivänä. Lisäksi sen tulee toimia yleisimmillä selaimilla, kuten Google Chromella, Firefoxilla, Internet Explorerilla, Operalla ja Safarilla.

5 AKTIVOIVAA OPETUSTA TUKEVAN VERKKO-OPPIMISYMPÄRISTÖN SUUNNITTELUVAIHE

Cuulis-oppimisympäristön suunnitteluvaihe toteutetaan edellä esitettyjen toiminnallisten ja ei-toiminnallisten vaatimusten pohjalta. Ensin suoritetaan arkkitehtuurisuunnittelu, jossa suunnitellaan oppimisympäristön kokonaisrakenne. Tarkoituksena on hahmottaa sekä Cuulis-oppimisympäristön suhde ympäristöönsä että sen sisältämät komponentit ja komponenttien väliset rajapinnat, jotta oppimisympäristölle asetetut vaatimukset voidaan toteuttaa. Tämän jälkeen suunnitellaan Cuulis-oppimisympäristön tietokanta, arkkitehtuurisuunnitelmassa hahmotetut yksittäiset komponentit sekä tuleva käyttöliittymä.

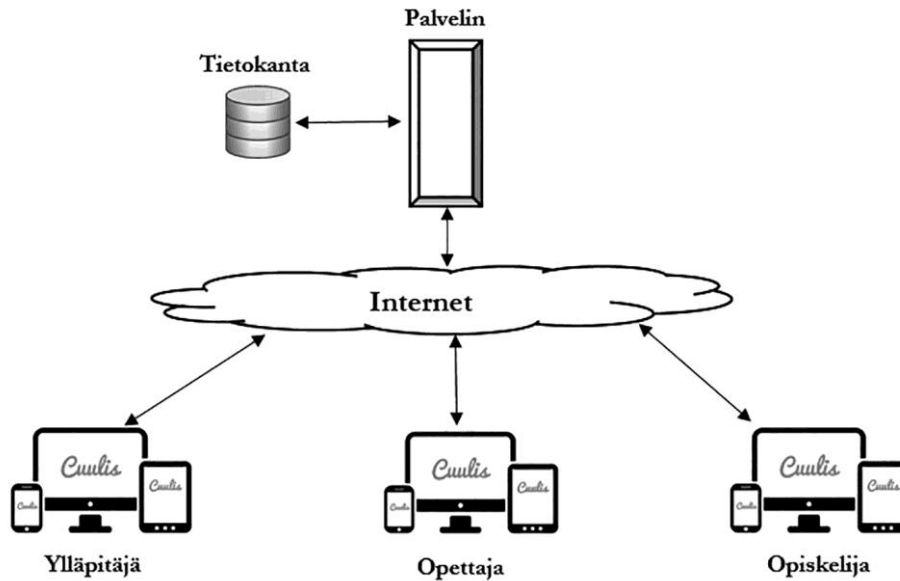
5.1 Arkkitehtuurisuunnittelu

Cuulis-oppimisympäristön arkkitehtuuri suunnitellaan vaatimusten yhteydessä hahmotellun arkkitehtuurikuvauksen pohjalta noudattamalla arkkitehtuurisuunnittelulle tyypillisiä yksinkertaisuus ja suoraviivaisuus-, osittaminen-, lokaalisuus- ja yhdenmukainen toteutusfilosofia-periaatteita. Suunnittelun tuotoksena syntyvät mallit laaditaan luovasti alusta asti itse, eivätkä ne pohjaudu mihinkään standardoituun mallinnuskieleen. Mallit laaditaan rakenteellisesta näkökulmasta siten, että ne toimivat pohjana oppimisympäristön käyttöliittymä- ja komponenttikohtaiselle suunnittelulle sekä antavat selkeän kuvan Cuulis-oppimisympäristön arkkitehtuurista kaikille oppimisympäristön käyttäjille ja tämän työn lukijoille.

Cuulis-oppimisympäristön toiminta perustuu asiakas-palvelin-arkkitehtuuriin siten, että sitä voi käyttää useampi käyttäjä yhtäaikaaisesti omalla laitteellaan. Käyttäjät, eli oppimisympäristöön liittyneet opettajat, opiskelijat ja ylläpitäjät, pyytävät asiakas-palvelin-arkkitehtuurimallin mukaisena asiakkaana toimivalla pöytäkoneellaan tai mobiililaitteellaan Cuulis-oppimisympäristön toimintaan liittyviä palveluita ulkoiselta palvelimelta. Palvelujen tarjoamiseen tarvittavaa tietoa on sekä itse että palvelimella että tietokannassa, johon palvelin on yhteydessä. Palvelupyynnöt prosessoituaan palvelin tarjoaa ne käyttäjille.

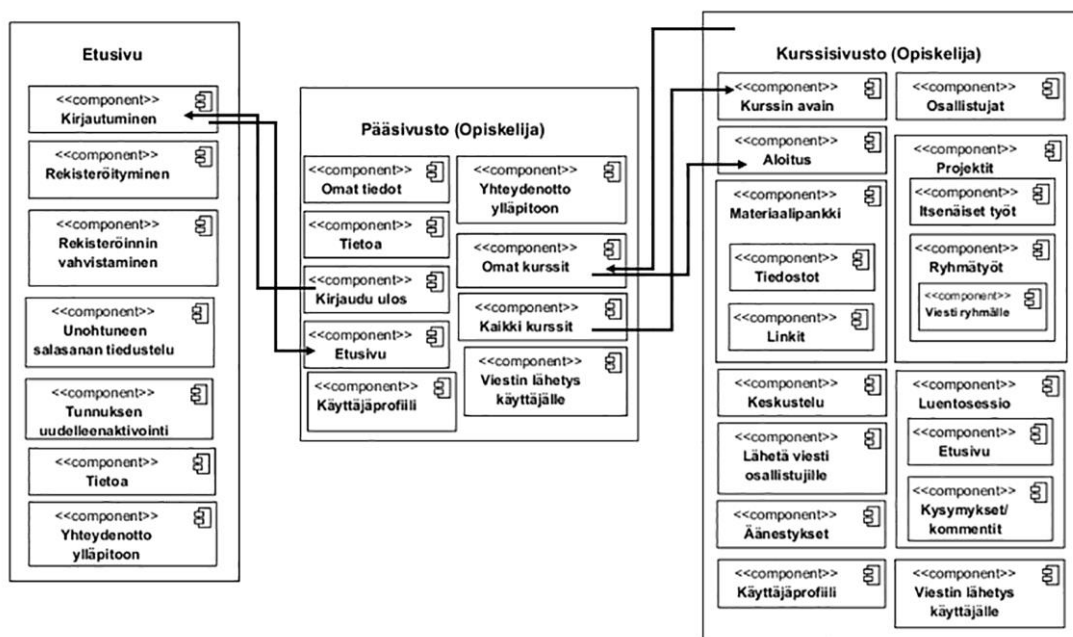
Sekä käyttäjien esittämät pyynnöt että palvelimen tarjoamat palvelut välittää Internet.

Kuva 26 pyrkii hahmottamaan tätä Cuulis-oppimisympäristön ulkoista arkkitehtuuria.

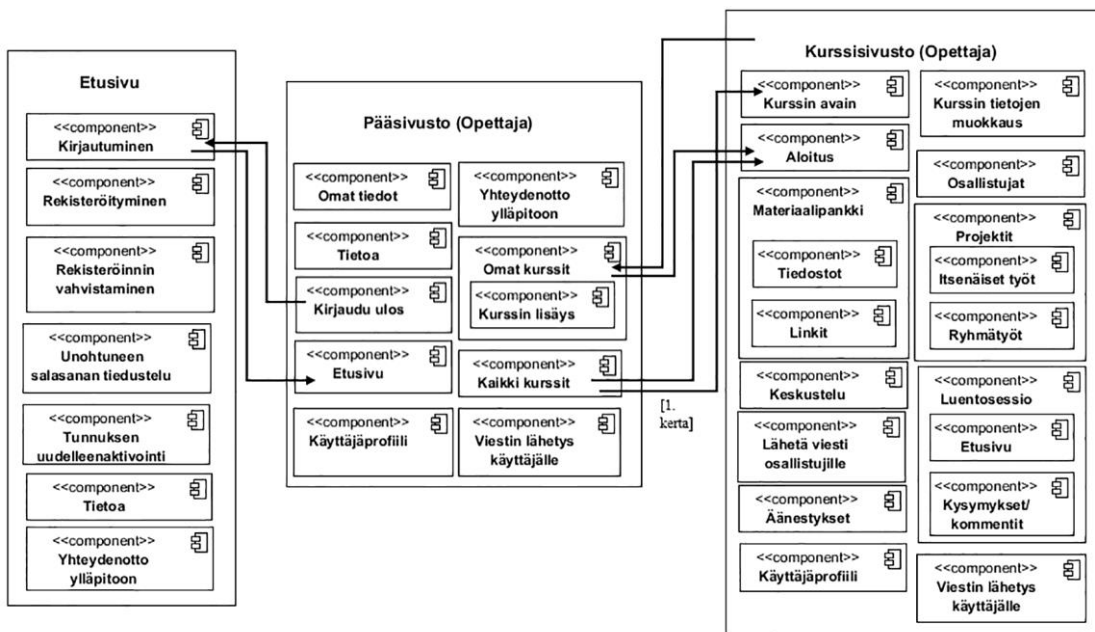


Kuva 26. Cuulis-oppimisympäristön toiminta perustuu asiakas-palvelin-arkkitehtuuriin.

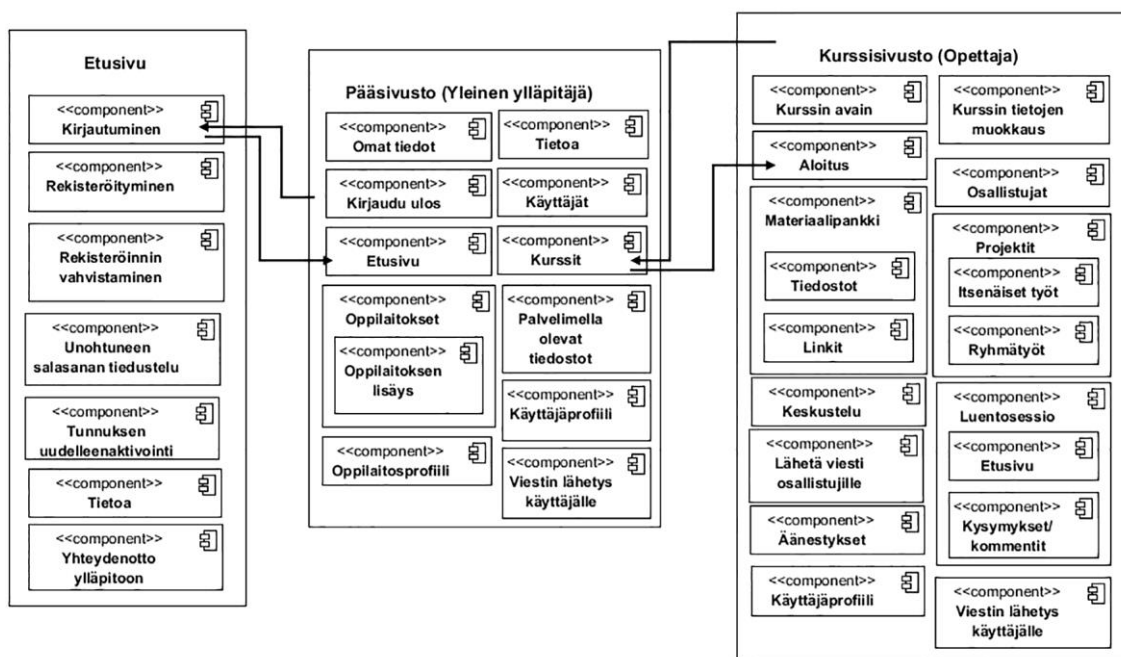
Seuraavassa kuvataan Cuulis-oppimisympäristön sisäinen arkkitehtuuri. Oppimisympäristö koostuu kolmesta eri sivustosta: etusivusta, pääsivustosta ja kurssisivustosta. Sivustojen sisältämät komponentit pohjautuvat vaatimusten esittelyn yhteydessä esiteltyyn toimintojen jakautumiseen eri osioihin kussakin sivustossa. Näiden lisäksi joitain komponentteja lisätään sekä paremman käytettävyyden että tiettyjen toimintojen toteutukseen liittyvien vaatimusten vuoksi. Koska sekä pääsivuston että kurssisivuston sisältämät komponentit ja siirtyminen näiden sivustojen välillä riippuu käyttäjän roolista, niin Cuulis-oppimisympäristön arkkitehtuurisuunnitelma tehdään jokaiselle käyttäjäroolille erikseen, ja ne esitetään kuvissa 27-31.



Kuva 27. Cuulis-oppimisympäristön sisäinen arkkitehtuuri ”Opiskelija”-käyttäjän osalta.



Kuva 28. Cuulis-oppimisympäristön sisäinen arkkitehtuuri ”Opettaja”-käyttäjän osalta.



Kuva 31. Cuulis-oppimisympäristön sisäinen arkkitehtuuri ”Yleinen ylläpitäjä”-käyttäjän osalta.

Cuulis-oppimisympäristön URL-osoitteen valinta vie jokaisen käyttäjän oppimisympäristön etusivun Kirjautuminen-komponenttiin, jos käyttäjän mahdollinen aiempi kirjautuminen ei ole enää selaimen muistissa. Mikäli käyttäjä suorittaa Kirjautuminen-komponenttiin liittyvän toiminnon onnistuneesti, käyttäjä siirtyy pääsivuston Etusivu-komponenttiin. Pääsivuston Kirjautu ulos-komponentista käyttäjä siirtyy takaisin etusivun Kirjautuminen-komponenttiin. Mikäli käyttäjä poistuu Cuulis-oppimisympäristöstä muun kuin Kirjautu ulos-komponentin kautta (esimerkiksi sulkemalla selaimen), niin myöhemmin samalla selaimella tehty Cuulis-oppimisympäristön URL-osoitteen valinta vie käyttäjän suoraan pääsivuston Etusivu-komponenttiin, jos käyttäjän aiempi kirjautuminen on vielä selaimen muistissa. Jos käyttäjän kirjautuminen katkeaa hänen ollessa minkä tahansa sivuston missä tahansa komponentissa, hänet ohjataan etusivun Kirjautuminen-komponenttiin. Etusivun Rekisteröityminen-, Unohtuneen salasanan tiedustelu-, Tietoa-, ja Yhteydenotto ylläpito-komponentteihin siirrytään etusivulla olevien linkkien kautta. Rekisteröinnin vahvistaminen- ja Tunnuksen uudelleenaktivointi-komponentteihin siirrytään puolestaan sähköpostiin saapuneen linkin kautta.

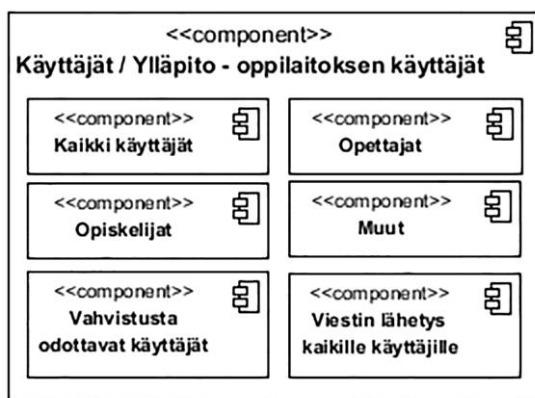
Kurssisivustolla on vain roolit ”Opettaja” ja ”Opiskelija”. Siirtyessään kurssisivustolle kaikilla muilla paitsi ”Opiskelija”-käyttäjillä on ”Opettaja”-rooli. ”Opiskelija”-käyttäjä siirtyy pääsivustosta haluamansa kurssin kurssisivustolle joko pääsivuston Omat kurssit- tai Kaikki kurssit-komponentista (kuva 27). Omat kurssit-komponentista siirtyessään ”Opiskelija”-käyttäjä siirtyy kurssisivuston Aloitus-komponenttiin ja Kaikki kurssit-komponentista siirtyessään kurssisivuston Kurssin avain-komponenttiin. Kurssisivuston sulkemisen jälkeen ”Opiskelija”-käyttäjä palaa takaisin Omat kurssit-komponenttiin.

Myös ”Opettaja”-käyttäjä voi siirtyä pääsivustosta haluamansa kurssin kurssisivustolle sekä Omat kurssit-komponentista että Kaikki kurssit-komponentista (kuva 28). Jos ”Opettaja”-käyttäjä siirtyy ensimmäistä kertaa kurssin kurssisivustolle Kaikki kurssit-komponentista, niin hän siirtyy kurssisivuston Kurssin avain-komponenttiin. Muussa tapauksessa hän siirtyy kurssisivuston Aloitus-komponenttiin. Kurssisivuston sulkeminen palauttaa ”Opettaja”-käyttäjän aina Omat kurssit-komponenttiin.

”Oppilaitoskohtainen ylläpitäjä”-käyttäjä siirtyy pääsivustosta haluamansa kurssin kurssisivustolle Ylläpito-oppilaitoksen kurssit-komponentista siten, että kunkin kurssin kohdalla hän siirtyy ensimmäisellä kerralla kurssisivuston Kurssin avain-komponenttiin ja myöhemmin Aloitus-komponenttiin (kuva 29). Kurssisivuston sulkeminen palauttaa ”Oppilaitoskohtainen ylläpitäjä”-käyttäjän takaisin Ylläpito-oppilaitoksen kurssit-komponenttiin.

”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjä voi siirtyä pääsivustosta kurssisivustolle Omat kurssit-, Kaikki kurssit- ja Ylläpito-oppilaitoksen kurssit-komponenteista (kuva 30). Kaikki kurssit- ja Ylläpito-oppilaitoksen kurssit-komponentista siirtyminen vie ”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjän kunkin kurssin kohdalla ensimmäisellä kerralla kurssisivuston Kurssin avain-komponenttiin ja muutoin Aloitus-komponenttiin. Kurssisivuston sulkeminen palauttaa ”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjän aina takaisin pääsivuston Omat kurssit-komponenttiin.

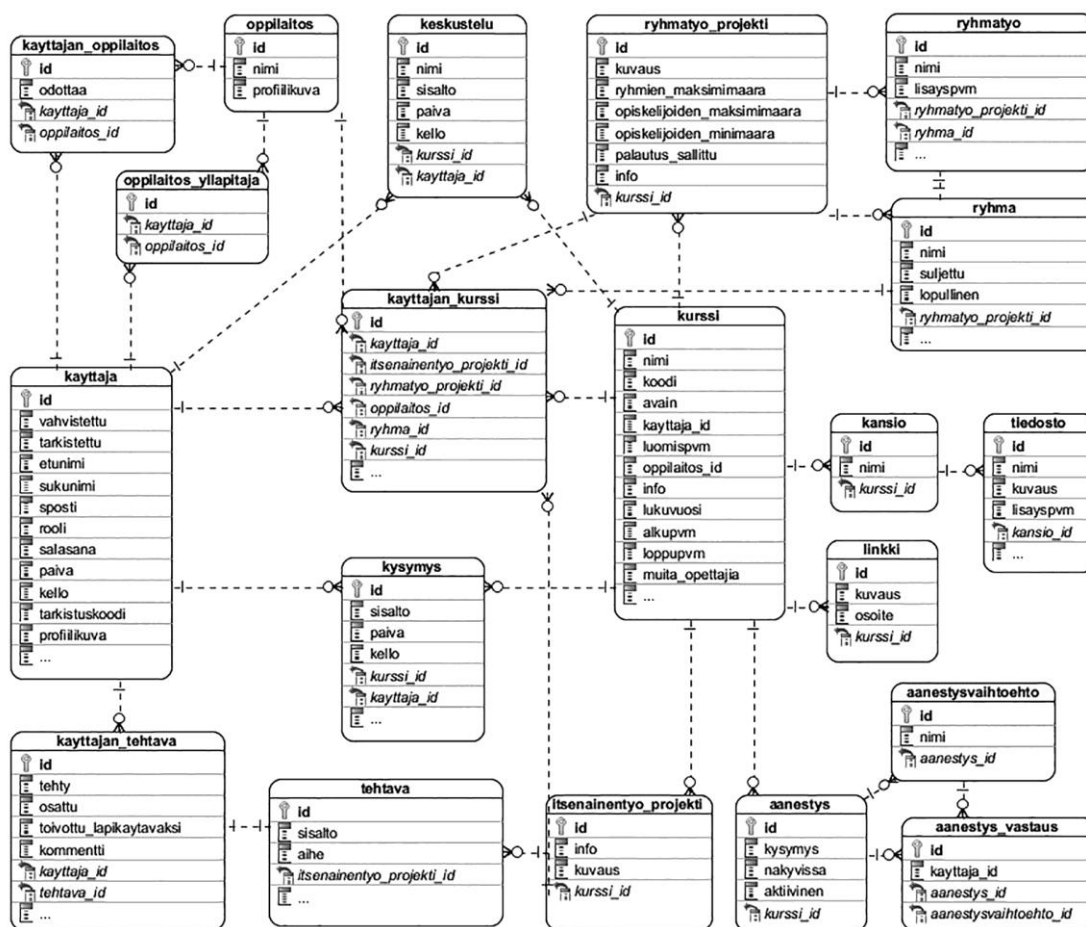
”Yleinen ylläpitäjä”-käyttäjä siirtyy pääsivuston Kurssit-komponentista kurssisivustolle aina sen Aloitus-komponenttiin, ja kurssisivuston sulkeminen palauttaa ”Yleinen ylläpitäjä”-käyttäjän takaisin pääsivuston Kurssit-komponenttiin (kuva 31). ”Yleinen ylläpitäjä”-käyttäjän pääsivuston Käyttäjät-komponentti, sekä ”Oppilaitoskohtainen ylläpitäjä”- ja ”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjien Ylläpito-oppilaitoksen käyttäjät-komponentti sisältää lisäksi kuvan 32 mukaiset osakomponentit.



Kuva 32. Cuulis-oppimisympäristön ylläpitäjien käyttäjiin liittyvien komponenttien sisältämät osakomponentit.

5.2 Tietokantasuunnittelu

Kaikki Cuulis-oppimisympäristön sisältämä tietoaineisto tallennetaan MySQL-relaatiotietokantaan, jota käsitellään SQL-kyselykielellä. Tietokantapalvelimen haltija vastaa tietokannan säännöllisestä varmuuskopioinnista. Cuulis-oppimisympäristön tietokannan rakenne esitellään suurpiirteisesti kuvassa 33. Täsmällisempää kuvausta tietokannan rakenteesta ei Cuulis-oppimisympäristön turvallisuusvaatimusten mukaisesti esitetä tässä työssä.



Kuva 33. Suuntaa-antava kuvaus Cuulis-oppimisympäristön tietokannan sisällöstä.

5.3 Komponenttisuunnittelu

Cuulis-oppimisympäristön käyttäjäroolikohtaisten arkkitehtuurisuunnitelmien sisältämät komponentit toteutetaan siten, että ne yhdessä toteuttavat oppimisympäristön toiminnalliset ja ei-toiminnalliset vaatimukset. Yksityiskohtaisempi komponenttisuunnittelu esitetään toteutusvaiheessa kunkin komponentin toteutuksen yhteydessä.

5.4 Käyttöliittymäsuunnittelu

Cuulis-oppimisympäristön käyttöliittymän suunnittelun taustalla on PACT-analyysi. Cuulis-oppimisympäristön käyttäjiä (People) ovat yleisen ylläpitäjän lisäksi oppilaitosten opettajat ja opiskelijat sekä muut oppilaitoksen ylläpitäjänä toimivat henkilöt, joten käyttäjien tietotekniikan osaaminen voi olla hyvin vaihtelevaa. Näin ollen Cuulis-oppimisympäristön käyttöliittymän tulee olla sellainen, että se on helppokäyttöinen tietoteknisen osaamisen suhteen kaikenlaisille käyttäjille. Koska Cuulis-oppimisympäristöä on tarkoitus käyttää sekä opetuksen aikana että oppilaitoksen ulkopuolella, niin ei voida tietää tarkalleen, missä kaikissa eri yhteyksissä (Context) oppimisympäristöä tullaan käyttämään. Cuulis-oppimisympäristöä on tarkoitus käyttää pöytäkoneiden lisäksi erilaisilla mobiililaitteilla, kuten kannettavilla tietokoneilla, tableteilla sekä älypuhelimilla, ja sen on toimittava kaikilla yleisimmillä Internet-selaimilla (Technologies). Näin ollen responsiivisuus ja eri selainten erityispiirteet on otettava huomioon Cuulis-oppimisympäristön kehityksessä. Cuulis-oppimisympäristön toiminnot (Activities) on esitetty sen vaatimusten esittelyn yhteydessä. Koska toiminnoissa on eroja eri käyttäjäryhmien välillä, niin eri käyttäjärooleille suunnitellaan omat käyttöliittymät. Cuulis-oppimisympäristön käyttöliittymät suunnitellaan luovalla lähestymistavalla hyödyntäen Nielsenin käytettävyyssääntöjä. Taulukossa 4 esitetään suunnitellut toimenpiteet kunkin käytettävyyssäännön osalta.

Cuulis-oppimisympäristön käyttöliittymäsuunnittelu pohjautuu oppimisympäristön sisäiseen arkkitehtuurisuunnitelmaan. Etusivun, pääsivuston ja kurssisivuston käyttöliittymät poikkeavat toisistaan, mutta niiden yleisilme on samanlainen. Kullakin sivustolla on samanlainen ylätunniste (header) ja alatunniste (footer). Ylätunnisteessa on Cuulis-oppimisympäristön logo sekä pilotointivaiheessa tieto siitä, että oppimisympäristö on vielä kehitysvaiheessa. Logosta tehdään linkki siten, että sitä klikkaamalla käyttäjä siirtyy oppimisympäristön etusivulle, eli kirjautumistilanteesta riippuen joko etusivun Kirjautuminen- tai pääsivuston Etusivu-komponenttiin. Pilotointivaiheessa ylätunnisteessa on myös linkki, jonka kautta käyttäjä pääsee lähettämään viestin Cuulis-oppimisympäristön yleiselle ylläpitäjälle. Näin käyttäjät voivat kertoa havaitsemistaan virheistä tai antaa muita kehitysideoita. Alatunnisteessa on Cuulis-oppimisympäristön kehittäjän copyright-tiedot,

ja kehittäjän nimestä tehdään linkki, jota klikkaamalla pääsee tarkastelemaan hänen tietojaan.

Taulukko 4. Cuulis-oppimisympäristön käyttöliittymän suunnittelu Nielsenin käytettävyyssääntöjen avulla.

Sääntö	Toimenpiteet
1. Näkyvyys	<ul style="list-style-type: none"> - Navigointipalkeissa se painike, jota käyttäjä on viimeksi klikannut muuttuu erilaiseksi, jotta käyttäjä tietää, missä osiossa hän kyseisellä hetkellä on. - Kullekin sivulle lisätään kyseiseen sivuun viittaava ns. title-tunniste, joka näkyy selaimen välilehdellä, jolloin käyttäjä tunnistaa, missä osiossa hän kyseisellä hetkellä on. - Käyttäjälle annetaan tieto erilaisten toimintojen, kuten tietojen muokkauksien ja asioiden poiston, onnistumisesta/epäonnistumisesta, ellei hän seuraavassa näkymässä näe tätä.
2. Yhteensopivuus todellisen maailman kanssa	<ul style="list-style-type: none"> - Käyttöliittymässä ei ole mitään epäselviä käsitteitä. - Symboleina käytetään mahdollisimman paljon tuttuja ja yleisesti käytössä olevia kuvia. Sellaisten symbolien kohdalla, jotka eivät välttämättä ole ennestään selviä, lisätään selventävä teksti.
3. Käyttäjien hallinta ja vapaus	<ul style="list-style-type: none"> - Näkymiin lisätään mahdollisuus palata edelliseen näkymään etenkin silloin, jos käyttäjä on valinnut tietyn asian poisto-toiminnon tai paluu ei ole mahdollista suoraan käyttöliittymän kautta.
4. Johdonmukaisuus ja standardit	<ul style="list-style-type: none"> - Symbolit, värit ja fontti pysyvät yhdenmukaisena eri näkymissä. - Suurin osa symboleista on tunnettuja ja yleisesti käytössä olevia.
5. Virheiden ehkäisy	<ul style="list-style-type: none"> - Jokaiseen käyttäjän tekemään toimintoon, kuten asian poistoon, kysytään varmistusta, ellei sen peruminen/muokkaaminen ole myöhemmin enää mahdollista.
6. Muistikuormituksen minimoiminen	<ul style="list-style-type: none"> - Käyttöliittymä on selkeä ja käyttäjää opastetaan eri näkymissä siten, että se kohtelee käyttäjiä kuin he olisivat ensimmäistä kertaa käyttämässä oppimisympäristöä.
7. Käytön tehokkuus ja joustavuus	<ul style="list-style-type: none"> - Käyttöliittymä on kauttaaltaan selkeä, joten sen käyttö on aina mahdollisimman tehokasta huolimatta siitä, käyttäkö käyttäjä oppimisympäristöä ensimmäistä vai ei.
8. Estetiikka ja minimalistinen suunnittelu	<ul style="list-style-type: none"> - Käyttöliittymä sisältää vain tarvittavat elementit ja informaation.
9. Virhetilanteiden käsittely	<ul style="list-style-type: none"> - Jokaisen virhetilanteen, kuten käyttäjän tekemän epäonnistuneen toiminnon, jälkeen käyttäjälle annetaan tästä selkeä tieto.
10. Opastus ja ohjeet	<ul style="list-style-type: none"> - Käyttöliittymä opastaa käyttäjää mahdollisimman paljon siten, että erillistä ohjeosiota ei tarvita oppimisympäristön tehokkaaseen käyttöön.

Värimaailma pidetään suhteellisen maltillisena. Taustaväri on valkoinen. Teksteissä ja eri painikkeissa vaihdellaan mustan ja pinkin erään version (#ff0099) välillä. Taulukoissa väritetään selkeyden vuoksi joka toinen rivi valkoiseksi ja joka toinen rivi haalean vaaleanpunaiseksi (#ffe5f5) siten, että taulukon otsikkorivi on väritetty hieman tummempalla vaaleanpunaisella (#ffcceb). Ylätunnisteen otsikon fontti on Pacifico ja muun tekstin Droid Serif. Symboleissa hyödynnetään valmiita HTML-symboleja. Ne symbolit, joita ei ole valmiina, piirretään kuvankäsittelyohjelmalla. Sivujen layout jaetaan sekä kahteen, kolmeen että neljään osaan siten, että kunkin sivun sisältö näyttää hyvältä kaikenkokoisilla laitteilla. Lisäksi kuvankäsittelyohjelmalla luodaan Cuulis-oppimisympäristön oma ns. favicon, eli kuvake, joka näkyy sen jokaisen sivun välilehdellä.

Cuulis-oppimisympäristön etusivu sisältää valmiiksi Kirjautuminen-komponentin. Navigointi, eli linkit, joilla käyttäjä voi siirtyä Unohtuneen salasanan tiedustelu-, Rekisteröityminen- ja Yhteydenotto ylläpitäjä-komponentteihin sijoitetaan Kirjautuminen-komponentin alapuolelle. Tietoa-komponenttiin vievä linkki puolestaan sijoitetaan Kirjautuminen-komponentin yläpuolelle.

Pääsivustolla on kaksi päällekkäistä käyttäjäryhmäkohtaista navigointipalkkia. Ylemmän navigointipalkin linkkien avulla käyttäjät voivat siirtyä Omat tiedot-, Tietoa-, Kirjautu ulos-komponentteihin, sekä Yleinen ylläpitäjä-käyttäjää lukuunottamatta Yhteydenotto ylläpitoon- komponenttiin. Pilotointivaiheessa tähän lisätään myös linkki, jonka kautta käyttäjä voi lähettää Cuulis-oppimisympäristön yleiselle ylläpitäjälle viestin, jossa voi ilmoittaa havaitsemistaan virheistä ja muista ongelmista, tai antaa kehitysideoita. Seuraavassa esitellään käyttäjäroolikohtaisesti mihin komponentteihin alemman navigointipalkin sisältämät linkit vievät.

”Opettaja”- ja ”Opiskelija”-käyttäjät:

- Etusivu
- Omat kurssit
- Kaikki kurssit

”Oppilaitoskohtaisen ylläpitäjä”-käyttäjä:

- Etusivu
- Ylläpito-oppilaitoksen käyttäjät (nimetään Käyttäjät)
- Ylläpito-oppilaitoksen kurssit (nimetään Kurssit)
- Ylläpito-oppilaitos (nimetään Oppilaitos)

”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjä:

- Etusivu
- Omat kurssit
- Kaikki kurssit
- Ylläpito-oppilaitoksen käyttäjät
- Ylläpito-oppilaitoksen kurssit
- Ylläpito-oppilaitos

”Yleinen ylläpitäjä”-käyttäjä:

- Etusivu
- Oppilaitokset
- Käyttäjät
- Kurssit
- Palvelimella olevat tiedostot

Käyttäjät- ja Ylläpito-oppilaitoksen käyttäjät-komponenteissa on lisäksi oma navigointipalkki, jossa on linkit kyseisen komponentin osakomponentteihin.

Kurssisivuston navigointipalkki sisältää Aloitus-, Osallistujat-, Projektit-, Materiaalipankki-, Luentosessio-, Keskustelu-, Äänestykset-komponentteihin, ja käyttäjäroolista riippuen joko Lähetä viesti osallistujille- tai Lähetä viesti opettajalle-komponenttiin vievät linkit. Navigointipalkin yläpuolella näkyy sen oppilaitoksen logo, johon kurssi liittyy, sekä kurssin nimi. Kurssin nimen vieressä on ”Opettaja”-käyttäjän käyttöliittymässä linkki, jonka kautta käyttäjä voi siirtyä Kurssin tietojen muokkaus-komponenttiin.

Materiaalipankki-, Luentosessio-, Äänestykset- ja Projektit-komponenteissa on lisäksi vasemmalla sivunavigointipalkit. Luentosessio-, Materiaalipankki- ja Projektit-komponenteissa sivunavigointipalkki sisältää niiden osakomponentteihin vievät linkit. Jokaisessa Materiaalipankki-komponentin Tiedostot-osakomponentissa sivunavigointipalkissa on lisäksi linkit sinne lisättyihin kansioihin. Projektit-komponentin Itsenäiset työt- ja Ryhmätyöt-osakomponenteissa sivunavigointipalkissa on linkit kuhunkin projektiin. Äänestykset-komponentin sivunavigointipalkissa on linkit komponenttiin lisättyihin äänestyksiin siten, että linkissä lukee kyseinen äänestyskysymys. Luentosessio-komponentin Kysymykset/kommentit-osakomponentti avautuu uuteen välilehteen, jos sen sisältämä toiminnallisuus on aktivoitu. Uuden välilehden näkymä sisältää vain osakomponentin sisältämän toiminnallisuuden, ja välilehti voidaan sulkea näkymässä olevaa x-symbolia klikkaamalla. Kurssisivustosta voidaan palata pääsivustoon näkymässä olevaa x-symbolia klikkaamalla.

6 AKTIVOIVAA OPETUSTA TUKEVAN VERKKO-OPPIMISYMPÄRISTÖN TOTEUTUS

Seuraavassa esitetään Cuulis-oppimisympäristön toteutusvaihe käyttöliittymän ja komponenttien osalta. Tarkoituksena on tarjota tämän työn lukijalle kokonaiskuva Cuulis-oppimisympäristön käyttöliittymästä ja eri komponenttien toiminnallisuuksista, sekä niiden toteutuksessa huomioon otettavista tekijöistä, joten täysin yksityiskohtaista kuvausta toteutuksesta ei anneta. Komponenteille ei suoriteta systemaattista yksikkötestausta eikä käyttöliittymän käytettävyyttä arvioida systemaattisin menetelmin, sillä ne on rajattu tämän työn ulkopuolelle. Valmiit tiedostot siirretään suomalaiselle palvelimelle, jonka haltija huolehtii tiedostojen säännöllisestä varmuuskopioinnista.

Cuulis-oppimisympäristö vaatii käyttäjän ja oppimisympäristön välistä jatkuvaa vuorovaikutusta, joten toteutuksessa hyödynnetään HTML-kielen lisäksi tähän tarkoitukseen soveltuvia ohjelmointikieliä. Tässä työssä dynaamisuus toteutetaan pääosin PHP-kielillä. Välitöntä vuorovaikutusta vaativat, eli selaimella suoritettavat toiminnot, toteutetaan JavaScriptillä hyödyntämällä mahdollisuuksien mukaan jQueryä. Ulkoasu toteutetaan CSS-tyylikielillä.

6.1 Käyttöliittymä

Cuulis-oppimisympäristön käyttöliittymä toteutetaan käyttöliittymäsuunnitelman mukaisesti. Sivut jaetaan osiin HTML-kielen div-elementtien avulla. Ylä- ja alatunnisteilla on omat div-lohkonsa. Näiden välissä oleva varsinainen sisältöosa jaetaan leveysuunnassa joko kahteen, kolmeen tai neljään div-lohkoon. Responsiivisuus toteutetaan mediakyselyiden avulla siten, että näkymän kooksi tulee aina laitteen koko. Taulukoissa responsiivisuus huomioidaan puolestaan siten, että taulukon yhteyteen tulee vierityspalkki, jos sen koko ylittää sille varatun tilan näkymässä. Välilehdillä näkyvä favicon-kuva tehdään kuvankäsittelyohjelmalla siten, että kuvan koko on 16x16 pikseliä, tallennetaan .png-muotoon ja viedään palvelimelle. Lisäksi jokaiselle sivulle lisätään ns. title-tunniste,

jonka avulla käyttäjä tunnistaa, missä sivulla hän kyseisellä hetkellä on. Kuvassa 34 on esimerkki Cuulis-oppimisympäristön välilehdestä.



Kuva 34. Cuulis-oppimisympäristön etusivun välilehti.

Käyttöliittymään liittyvät seikat pidetään mielessä kaikkien komponenttien suunnittelussa ja toteutuksessa. Esimerkkejä käyttöliittymän toteutuksesta annetaan seuraavassa, komponenttien toteutuksen yhteydessä esitellyissä kuvissa.

6.2 Komponentit

Cuulis-oppimisympäristön eri komponenteissa olevat käyttäjien, kurssien ja oppilaitosten luettelot, sekä itsenäisiin projektitöihin liittyvät tehtäväluettelot toteutetaan taulukoilla. Taulukoiden otsikoista tehdään pääosin klikattavia siten, että rivien järjestyttä voidaan muuttaa. Lisäksi taulukoiden otsikot kiinnitetään jQueryn avulla näkymään siten, että ne näkyvät pidempienkin taulukoiden yhteydessä koko ajan. Taulukoiden yhteyteen lisätään hakutoiminto, jolla helpotetaan halutun kurssin, käyttäjän tai oppilaitoksen löytämistä. Kaikkien käyttäjien, kurssien ja oppilaitosten poistoon liittyvien toimintojen yhteydessä poistetaan tietokannasta kaikki näihin liittyvät tiedot ja palvelimelta kaikki näihin liittyvät tiedostot. Käyttäjältä pyydetään aina varmistusta kaikkiin sellaisiin toimintoihin, joita ei myöhemmin voida kumota.

Kaikki komponentit, jotka liittyvät viestin lähettämiseen, sisältävät lomakkeen, joka koostuu kentästä, joista toinen viittaa viestiä lähettävän käyttäjän nimeen ja toinen viestiä

lähettävän käyttäjän sähköpostiosoitteeseen. Kaikissa muissa paitsi etusivun viestin lähettämiseen liittyvässä komponentissa nämä kentät ovat valmiiksi täytettynä siten, ettei käyttäjä voi niitä muokata. Lomakkeissa on luonnollisesti myös viestiin viittaava tekstialue, johon käyttäjä voi kirjoittaa haluamansa viestin. Viesti lähetetään valitun/valittujen käyttäjän/käyttäjien oppimisympäristöön rekisteröimään sähköpostiosoitteeseen siten, että viestin yhteydessä näkyy viestin lähettäneen käyttäjän nimi ja viestiin voi vastata suoraan viestin lähettäneen käyttäjän sähköpostiosoitteeseen. Etusivun viestin lähettämiseen liittyvän komponentin sisältö esitetään tarkemmin kyseisen komponentin esittelyn yhteydessä.

Cuulis-oppimisympäristö lähettää eri komponenttien sisältämien toiminnallisuuksien yhteydessä automaattisesti viestejä sen käyttäjille. Nämä viestit lähetetään käyttäjien oppimisympäristöön rekisteröimiin sähköpostiosoitteisiin. Kunkin automaattisesti lähtevän viestin lähettäjän sähköpostiosoite muotoillaan siten, ettei viestiin ole mahdollista vastata. Tämä mainitaan myös lähetetyn viestin yhteydessä. Kaikki Cuulis-oppimisympäristöstä lähtevät, sekä käyttäjien lähettämät että automaattisesti lähtevät, viestit muotoillaan siten, että vastaanottaja näkee postilaatikossaan, että viesti on tullut Cuulis-oppimisympäristöstä. Lisäksi viestitoiminnot toteutetaan siten, etteivät viestit mene vastaanottajan roska-postilaatikkoon.

Seuraavassa esitellään tarkemmin Cuulis-oppimisympäristön eri sivustojen sisältämien komponenttien yksityiskohtaisempi suunnittelu ja toteutus painottaen niitä tekijöitä, jotka ovat oppimisympäristön kehityksen kannalta keskeisimpiä ja vaativat eniten huomiota. Toteutuksen kannalta kriittisimmät komponenttikohtaiset suunnitelmat täydennetään UML:n aktiviteettikaavioilla. Siirtyminen komponentista toiseen toteutetaan HTML-linkkien avulla.

6.2.1 Etusivun komponentit

Etusivun *Rekisteröityminen*-komponentti sisältää rekisteröitymislomakkeen, johon käyttäjä voi syöttää etunimen, sukunimen ja sähköpostiosoitteen, sekä valita roolin ja ensisijaisen oppilaitoksen. Rooli-valintakentän vaihtoehdot ovat ”Opiskelija”, ”Opettaja” ja

”Muu”. Ensisijainen oppilaitos-valintakentän vaihtoehtoina ovat kaikki Cuulis-oppimisympäristöön lisätyt oppilaitokset. Koska käyttäjän syöttämä sähköpostiosoite toimii käyttäjätunnuksena ja sitä hyödynnetään viestiliikenteen yhteydessä, niin rekisteröitymislomakkeen tallennuksen yhteydessä tarkistetaan, että jokainen kentistä on täytetty, ja että sähköpostiosoite on oikeanmuotoinen eikä sitä ole jo rekisteröity oppimisympäristöön. Jos nämä ehdot täyttyvät, tietokantaan tallennetaan rekisteröityneen käyttäjän syöttämät tiedot sekä salattu tarkistuskoodi. Kuvassa 35 on näkymä rekisteröitymissivusta epäonnistuneen rekisteröintilomakkeen tallennuksen jälkeen.

The screenshot shows the registration page of the Cuulis learning environment. At the top left, there is a label 'ylätunniste' with an arrow pointing to the Cuulis logo and the text 'Cuulis - Verkko-oppimisympäristö'. Below the logo, a small message states: 'Tämä on oppimisympäristön beta-versio, eli se on vielä kehitysvaiheessa. Voit ilmoittaa esiintyneistä virheistä/ongelmista ja antaa kehitysideita tästä.' The main content area is titled 'Rekisteröityminen' and includes a link '<< Palaa etusivulle'. A message says 'Antamasi sähköpostiosoite on jo rekisteröity!' with an arrow pointing to the text 'mahdollisuus palata alkutilaan (Nielsenin 3. sääntö)'. Below this, it says 'Kaikki tiedot ovat pakollisia.' and 'Huom! Roolia ja ensisijaista oppilaitosta ei voi enää myöhemmin muuttaa!'. A message 'Vahvistuslinkki lähetetään antamaasi sähköpostiosoitteeseen.' is followed by a list of form fields: 'Etunimi:', 'Sukunimi:', and 'Sähköpostiosoite:'. To the right of these fields, there are two arrows pointing to the text 'selkeä virheilmoitus virheellisestä toiminnosta (Nielsenin 9. sääntö)'. Below the form fields, there is a dropdown menu for 'Ensisijainen oppilaitos: (Voit myöhemmin liittyä myös muihin)' with a 'Valitse' button, and another dropdown menu for 'Rooli:' with a 'Valitse' button. Below these is a '✓ Rekisteröidy' button. To the right of the form fields, there is an arrow pointing to the text 'selkeä opastus (Nielsenin 10. sääntö)'. At the bottom of the page, there is a footer with 'Copyright © Cuulis 2015' and 'Marianne Sjöberg'. At the bottom left, there is a label 'alattunniste' with an arrow pointing to the footer area.

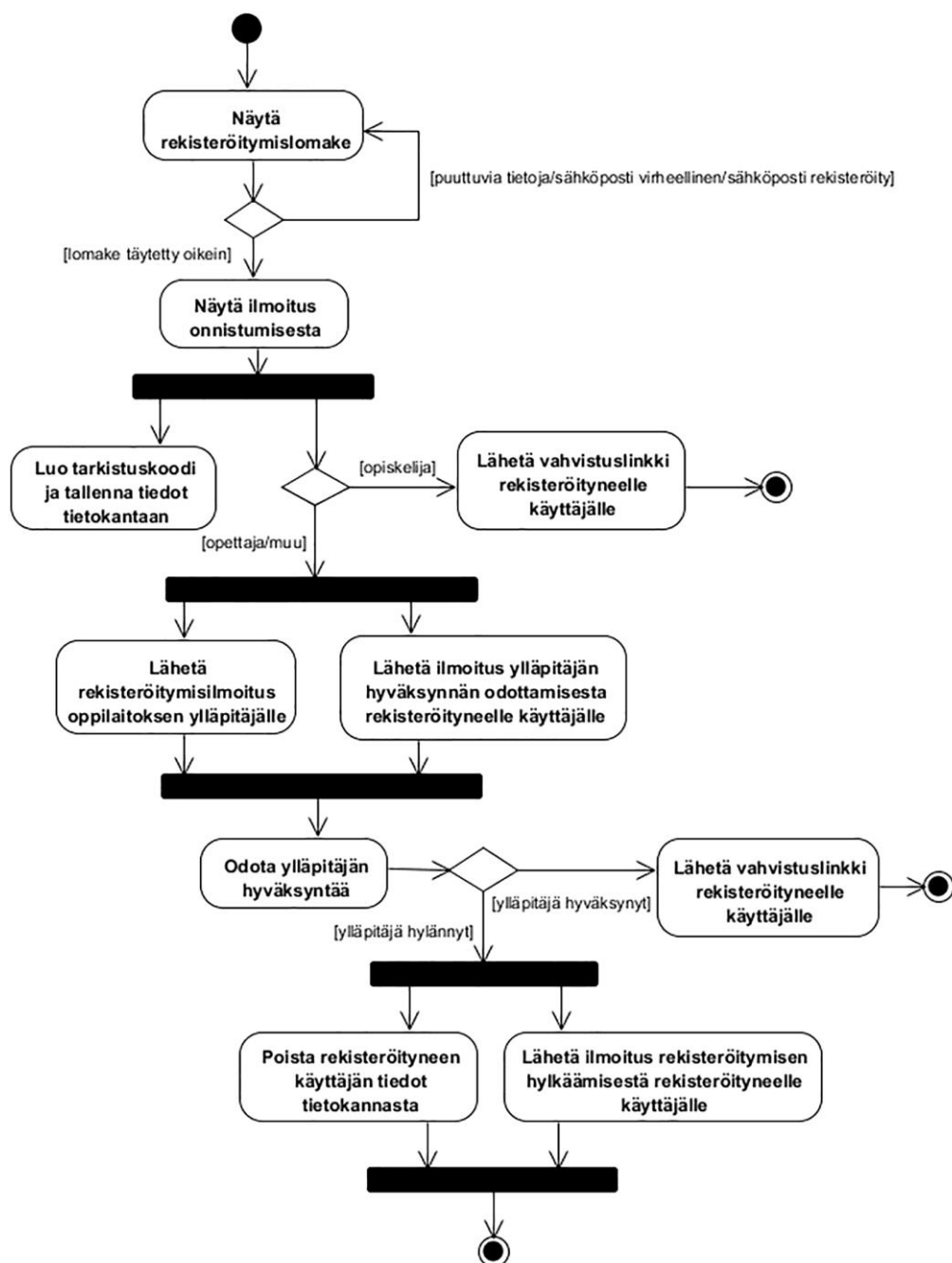
Kuva 35. Cuulis-oppimisympäristön rekisteröitymissivunäkymä epäonnistuneen rekisteröitymiseen liittyvän toiminnon jälkeen.

Onnistuneen tallennuksen jälkeen roolissa “Opiskelija” rekisteröityneelle käyttäjälle lähetetään sähköpostiin tarkistuskoodiin perustuva vahvistuslinkki. Roolissa “Opettaja”/”Muu” rekisteröityneelle käyttäjälle lähetetään puolestaan vain ilmoitus rekisteröitymisen onnistumisesta ja siitä, että rekisteröityminen odottaa vielä sen oppilaitoksen ylläpitäjän hyväksyntää, jonka käyttäjä valitsi ensisijaiseksi oppilaitoksekseen. Samalla lähetetään ilmoitus käyttäjän rekisteröitymisestä kyseisen ensisijaisen oppilaitoksen ylläpitäjälle/ylläpitäjille. Rekisteröityneelle käyttäjälle lähetetään vahvistuslinkki vasta sitten, kun ylläpitäjä on hyväksynyt rekisteröitymisen. Mikäli ylläpitäjä hylkää rekisteröitymisen, käyttäjälle ilmoitetaan tästä ja hänen tietonsa poistetaan tietokannasta. Rekisteröitymiseen liittyvä prosessi havainnollistetaan kuvan 36 aktiviteettikaaviolla. Yleinen ylläpitäjä rekisteröidään oppimisympäristöön lisäämällä tarvittavat tiedot tietokantaan manuaalisesti.

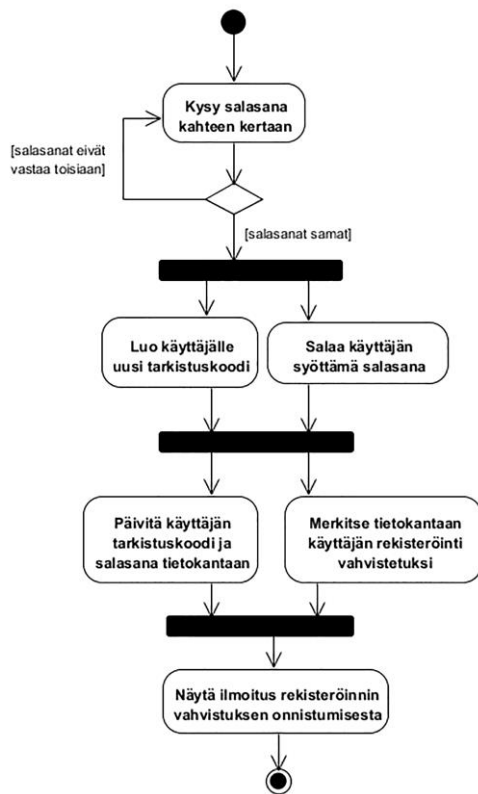
Käyttäjä vahvistaa rekisteröitymisensä sähköpostiin saamansa vahvistuslinkin avulla. Vahvistuslinkin klikkaaminen vie käyttäjän *Rekisteröinnin vahvistaminen*-komponenttiin, joka sisältää lomakkeen, jossa käyttäjä voi syöttää haluamansa salasanan. Lomakkeessa on kaksi kenttää, joten käyttäjän on syötettävä haluamansa salasana kahteen kertaan. Lomakkeen tallentamisen yhteydessä tarkistetaan, että syötetyt salasanat vastaavat toisiaan. Mikäli salasanat vastaavat toisiaan, käyttäjän syöttämä salasana salataan ja tallennetaan tietokantaan. Samalla tietokantaan päivitetään uusi tarkistuskoodi ja tieto siitä, että käyttäjä on vahvistanut rekisteröitymisensä. Rekisteröinnin vahvistamiseen liittyvä prosessi havainnollistetaan kuvassa 37.

Unohtuneen salasanan tiedustelu-komponentissa rekisteröityneellä käyttäjällä on toiminto, jonka avulla hän voi tilata sähköpostiinsa tunnuksen uudelleenaktivointilinkin. Komponentti sisältää lomakkeen, johon käyttäjä voi syöttää käyttäjätunnuksena toimivan sähköpostiosoitteen. Mikäli kyseinen sähköpostiosoite on tallennettu tietokantaan, siihen lähetetään tarkistuskoodiin perustuva linkki, jonka klikkaaminen vie käyttäjän *Tunnuksen uudelleenaktivointi*-komponenttiin. Tämä komponentti sisältää Rekisteröinnin vahvistaminen-komponentin kaltaisen lomakkeen, jossa käyttäjä voi syöttää haluamansa uuden salasanan kahteen kertaan. Salasanan vaihto, ja näin tapahtuvaan tunnuksen uudelleenaktivointiin liittyvä prosessi kuvataan kuvan 38 aktiviteettikaaviolla. Käyttäjä syöttää siis

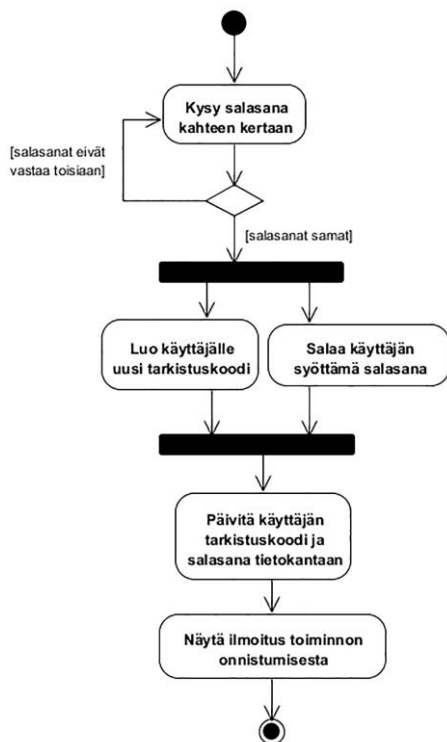
haluamansa salasanan kahteen kertaan, ja jos syötteet vastaavat toisiaan, salasana salataan ja tallennetaan tietokantaan kyseisen käyttäjän tietoihin. Tämän lisäksi käyttäjän tiedoissa oleva tarkistuskoodi uusitaan ja päivitetään tietokantaan.



Kuva 36. Rekisteröitymisprosessia kuvaava aktiviteettikaavio.



Kuva 37. Rekisteröinnin vahvistukseen liittyvää prosessia kuvaava aktiviteettikaavio.



Kuva 38. Käyttäjän salasanan syöttöä kuvaava aktiviteettikaavio.

Etusivun *Tietoa*-komponentti sisältää Cuulis-oppimisympäristön ominaisuuksista kertovan palvelimelle ladatun tekstitiedoston, jonka päivittämisestä vastaa oppimisympäristön yleinen ylläpitäjä. Lisäksi komponentissa on linkki, jonka kautta käyttäjä voi siirtyä Käyttäjäprofiili-komponenttiin, jossa on tiedot ”Yleinen ylläpitäjä”-käyttäjistä. Tietoa-komponentti toteutetaan siten, että siinä avataan palvelimelle viety Cuulis-oppimisympäristöstä kertova tekstitiedosto. Komponentin sisällön päivittämiseksi riittää siis tämän tekstitiedoston muokkaaminen ja päivitetyn tiedoston vieminen uudelleen palvelimelle.

Yhteydenotto ylläpitäjään-komponentti sisältää viestin lähettämiseen liittyvän lomakkeen, johon käyttäjä voi syöttää etunimen, sukunimen, sähköpostiosoitteen ja puhelinnumeron. Lisäksi lomakkeessa on viestin kirjoittamiseen tarkoitettu tekstialue sekä valintaruudut, johon käyttäjä voi merkitä haluaako vastauksen puhelimitse tai sähköpostitse. Viesti lähetetään ”Yleinen ylläpitäjä”-käyttäjän sähköpostiin.

Kirjautuminen-komponentti sisältää lomakkeen, johon käyttäjä voi syöttää käyttäjätunnuksena toimivan sähköpostiosoitteensa ja salasanaansa. Koska rekisteröinnin vahvistuksen yhteydessä käyttäjän antama salasana salattiin ennen tallentamista tietokantaan, niin käyttäjän Kirjautuminen-komponentissa syöttämä salasana salataan vastaavalla menetelmällä ennen käyttäjätunnuksen ja sitä vastaavan salasanan etsimistä tietokannasta. Mikäli käyttäjätunnusta ja salasanaa vastaava rivi löytyy tietokannasta, kyseinen ajankohta tallennetaan käyttäjän tietoihin tietokantaan ja käyttäjä ohjataan pääsivuston Etusivu-komponenttiin. Ainut poikkeus tähän on se, jos käyttäjän rekisteröinnissä valitsema rooli on ”Muu” eikä häntä ole vielä lisätty minkään oppilaitoksen ylläpitäjäksi, eli käyttäjällä ei ole vielä käyttöoikeuksia Cuulis-oppimisympäristöön. Tällöin käyttäjälle annetaan tieto tästä ja ohjataan ottamaan yhteyttä ylläpitoon. Kirjautumista hallitaan ns. istuntojen avulla siten, että selain muistaa käyttäjän kirjautumisen tietyn vakioajan. Jos käyttäjän istunto katkeaa automaattisesti, niin mikä tahansa valinta missä tahansa Cuulis-oppimisympäristön sivulla vie käyttäjän etusivun Kirjautuminen-komponenttiin. Kuvassa 39 on näkymä Cuulis-oppimisympäristön etusivusta.

Cuulis - Verkko-oppimisympäristö

Tämä on oppimisympäristön beta-versio, eli se on vielä kehitysvaiheessa. Voit ilmoittaa esiintyneistä virheistä/ongelmista ja antaa kehitysideoita [tästä](#).

**** Mikä on Cuulis? ****

Kirjaudu sisään:

Sähköpostiosoite:

Salasana:

Salasana unohtunut?
 Rekisteröidy >>
 Ota yhteyttä >>

etusivun navigointi

Copyright © Cuulis 2015
 Marianne Sjöberg

Kuva 39. Cuulis-oppimisympäristön etusivu.

6.2.2 Pääsivuston komponentit

Pääsivuston *Etusivu*-komponentissa on näkyvissä käyttäjän nimi, päivämäärä sekä tieto siitä, milloin käyttäjä on viimeksi kirjautunut Cuulis-oppimisympäristöön. Tätä varten tietokantaan tallennetaan jokaisen kirjautumisen yhteydessä kirjautumisen ajankohta. Lisäksi Etusivu-komponentissa käyttäjälle näkyy juokseva kellonaika, joka toteutetaan jQueryn avulla.

Pääsivuston *Tietoa*-komponentti on sama kuin etusivun *Tietoa*-komponentti. *Kirjaudu ulos*-komponentti sisältää toiminnon, joka katkaisee käyttäjän istunnon, eli komponentin valinta vie käyttäjän suoraan etusivun *Kirjautuminen*-komponenttiin.

Cuulis - Verkko-oppimisympäristö

Tämä on oppimisympäristön beta-versio, ellei ole vielä kehitysvaiheessa. Voit ilmoittaa esiintyneistä virheistä/ongelmista ja antaa kehitysideoita tästä.


käyttäjälle annetaan tieto siitä, missä osiossa hän on (Nielsenin 1. sääntö)

pääsivuston "Oppilaitoskohtainen ylläpitäjä"-käyttäjän navigointipalkit

Omat tiedot | Ota yhteyttä | Tietoa | Kirjaudu ulos | Ongelmia?

Etusivu | Käyttäjät | Kurssit | Oppilaitos

Oma käyttäjäprofiili



Etunimi: Teemu
Sukunimi: Ylläpitäjä
Rooli: Oppilaitoskohtainen ylläpitäjä
Sähköpostiosoite: teemu@testi.fi

Muokkaa tietoja | Poistu oppimisympäristöstä

SALASANAN VAIHTO:

Vanha salasana:

Uusi salasana:

Toista uusi salasana:

Vaihda salasana

Kuva 40. "Oppilaitoskohtainen ylläpitäjä"-käyttäjän *Omat tiedot*-komponenttiin liittyvä näkymä.

Omat tiedot-komponentti sisältää käyttäjän oman käyttäjäprofiilin. Omassa käyttäjäprofiilissa näkyy käyttäjän etunimi, sukunimi, sähköpostiosoite, rooli ja oppilaitokset, johon käyttäjä on liittynyt, sekä käyttäjän profiilikuva. Komponentissa käyttäjällä on toiminto, jonka avulla hän voi muokata tiedoista etunimeään, sukunimeään ja sähköpostiosoitetaan, sekä toiminto, jonka avulla hän voi lisätä, poistaa ja vaihtaa profiilikuvansa. Etunimen, sukunimen ja sähköpostiosoitteen muokkauksen yhteydessä on tarkistettava, että kaikki kentät on täytetty, sähköpostiosoite on oikeanmuotoinen ja ettei sitä ole jo rekisteröity oppimisympäristöön. Profiilikuva tallennetaan palvelimelle ja tässä yhteydessä tarkistetaan, että ei-toiminnallisissa vaatimuksissa esitellyt tiedostoihin liittyvät ehdot täyttyvät. Profiilikuvan vaihtamisen yhteydessä poistetaan palvelimelta kyseisen käyttäjän edellinen kuva. Komponentti sisältää myös lomakkeen, jossa käyttäjä voi vaihtaa salasanansa. Salasanan vaihtamiseen liittyvä prosessi on täysin samanlainen kuin etusivun Tunnuksen uudelleenaktivointi-komponentin esittelyn yhteydessä esitelty prosessi. Kuvassa 40 on ”Oppilaitoskohtainen ylläpitäjä”-käyttäjän *Omat tiedot*-komponenttiin liittyvä näkymä.

Yhteydenotto ylläpitäjään-komponentti sisältää lomakkeen, jossa käyttäjä voi lähettää viestin oppimisympäristön ylläpitäjälle. ”Opettaja”- ja ”Opiskelija”-käyttäjän viesti lähetetään sen oppilaitoksen ylläpitäjälle/ylläpitäjille, jonka käyttäjä valitsi rekisteröinnin yhteydessä ensisijaiseksi oppilaitoksekseen. ”Oppilaitoskohtainen ylläpitäjä”- ja ”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjän viesti lähetetään puolestaan oppimisympäristön yleiselle ylläpitäjälle.

”Opiskelija”- ja ”Opettaja”-käyttäjän *Omat kurssit*-komponentti sisältää luettelon kurseista, joihin käyttäjä on liitetty osallistujaksi. Kurssin nimen tai koodin klikkaaminen vie käyttäjän kyseisen kurssin kurssisivuston Aloitus-komponenttiin, ja kurssin vastuopettajan nimen klikkaaminen vie käyttäjän vastuopettajan tietoja sisältävään Käyttäjäprofiili-komponenttiin. Lisäksi ”Opiskelija”-käyttäjällä ja sillä ”Opettaja”-käyttäjällä, joka on merkitty kyseisen kurssin ns. muuksi opettajaksi, on toiminto, jonka avulla hän voi valita ne kurssit, joilta haluaa poistua. Poistumisen yhteydessä kyseinen käyttäjä poistetaan kaikkien valittujen kurssien osallistujista. ”Opettaja”-käyttäjällä, joka on merkitty

kurssin vastuopettajaksi, on toiminto, jonka avulla hän voi valita ne kurssit, jotka haluaa poistaa.

Omat kurssit-komponentissa ”Opettaja”-käyttäjällä on toiminto, jonka avulla hän voi siirtyä *Kurssin lisäys*-komponenttiin, joka sisältää uuden kurssin lisäämiseen tarkoitetun lomakkeen. Lomakkeessa käyttäjä voi syöttää kurssin nimen, koodin ja avaimen. Lisäksi käyttäjä voi valita, mihin niistä oppilaitoksista, joihin hän on liittynyt, haluaa kurssin liitettävän. Lisäksi hän voi valita, onko kurssilla muita opettajia, sekä lukuvuoden, jolloin kurssi järjestetään. Kurssin aloitus- ja päättymispäivämäärä-kentät toteutetaan jQueryn avulla. Käyttäjän aloittaessa syöttää tietoja näihin kenttiin näkymässä avautuu kalenteri, josta käyttäjä voi valita haluamansa päivän. Tämä toteutetaan näin, jotta jokaisen käyttäjän syöttämän päivämäärän muoto olisi samanlainen. Kalenterikenttien toteutuksessa esitetään, ettei käyttäjä syötä päivämääriä siten, että päättymispäivämäärä on ennen aloituspäivämäärää tai että aloituspäivämäärä on päättymispäivämäärän jälkeen, eikä aloituspäivämääräksi voi valita menneisyydessä olevaa päivää. Lomakkeessa on myös valmiiksi täytetty, käyttäjän nimen sisältävä, kurssin vastuopettajaan viittaava kenttä. Uuden kurssin lisäämisen yhteydessä tietokantaan tallennetaan käyttäjän syöttämien tietojen lisäksi ylläpitotoimia varten automaattisesti kyseinen päivämäärä.

”Opiskelija”- ja ”Opettaja”-käyttäjän *Kaikki kurssit*-komponentissa on luettelo niistä kursseista, jotka liittyvät oppilaitoksiin, joihin käyttäjä on liittynyt. Kurssin nimen tai koodin klikkaaminen vie käyttäjän kyseisen kurssin kurssisivuston Aloitus-komponenttiin, mikäli kyseessä ei ole ensimmäinen kerta, kun käyttäjä klikkaa kyseistä kurssia. Jos kyseessä on ensimmäinen kerta, käyttäjä siirtyy kurssisivuston Kurssin avain-komponenttiin. Kurssin vastuopettajan nimen klikkaaminen vie käyttäjän vastuopettajan tietoja sisältävään Käyttäjäprofiili-komponenttiin. Lisäksi kunkin kurssin yhteydessä on toiminto, jonka avulla käyttäjä voi siirtyä Viestin lähetys käyttäjälle-komponenttiin lähettämään viestin kyseisen kurssin vastuopettajalle. Kuvassa 41 on Kaikki kurssit-komponenttiin liittyvä näkymä ”Opettaja”- ja ”Opiskelija”-käyttäjien käyttöliittymässä.

Omat tiedot | Ota yhteyttä | Tietoa | Kirjaudu ulos Ongelmia? **pääsivuston "Opettaja"- ja "Opiskelija"-käyttäjän navigointipalkit**

Etusivu Omat kurssit **Kaikki kurssit** **käyttäjälle annetaan tieto siitä, missä osiossa hän on (Nielsenin 1. sääntö)**

Oppimisympäristössä olevat kurssit (oppilaitoksissa, joihin olet liittynyt): yleisesti tunnettu hakutoimintoon liittyvä symboli (Nielsenin 1. ja 4. sääntö)

Klikkaamalla kurssin nimeä tai koodia pääset kyseisen kurssin sivulle.

Hae kurssi: **taulukoiden otsikot klikattavia, jotta rivien järjestystä voi muuttaa**

Koodi	Kurssi	Vastuupettaja	Oppilaitos	Lukuvuosi	Alkaa	Päättyy	
123	TestiKurssi	Opettaja Testi	Testikoulu	2015-2016	20.01.2016	29.01.2016	📧 Lähetä viesti opettajalle
MA01	Funktiot ja yhtälöt	Opettaja Testi	Testikoulu	2015-2016	05.03.2016	31.03.2016	📧 Lähetä viesti opettajalle
MA10	Integraalilaskenta	Opettaja Testi	Testikoulu	2015-2016	05.03.2016	26.03.2016	📧 Lähetä viesti opettajalle
OH01	Ohjelmoinnin alkeet	Testaaja Teemu	Testikoulu	2015-2016	02.03.2016	01.05.2016	📧 Lähetä viesti opettajalle
OH07	Ohjelmoinnin kertauskurssi	Opettaja Testi	Testikoulu	2015-2016	02.03.2016	17.03.2016	📧 Lähetä viesti opettajalle

kurssin koodia tai nimeä klikkaamalla voi siirtyä kurssin kurssisivustolle

vastuupettajan nimeä klikkaamalla voi tarkastella hänen käyttäjäprofiilia

viestitoimintoon liittyvän symbolin yhteydessä selvennetään, kenelle viestin voi lähettää (Nielsenin 2. ja 10. sääntö)

Kuva 41. "Opettaja"- ja "Opiskelija"-käyttäjän Kaikki kurssit-komponenttiin liittyvä näkymä.

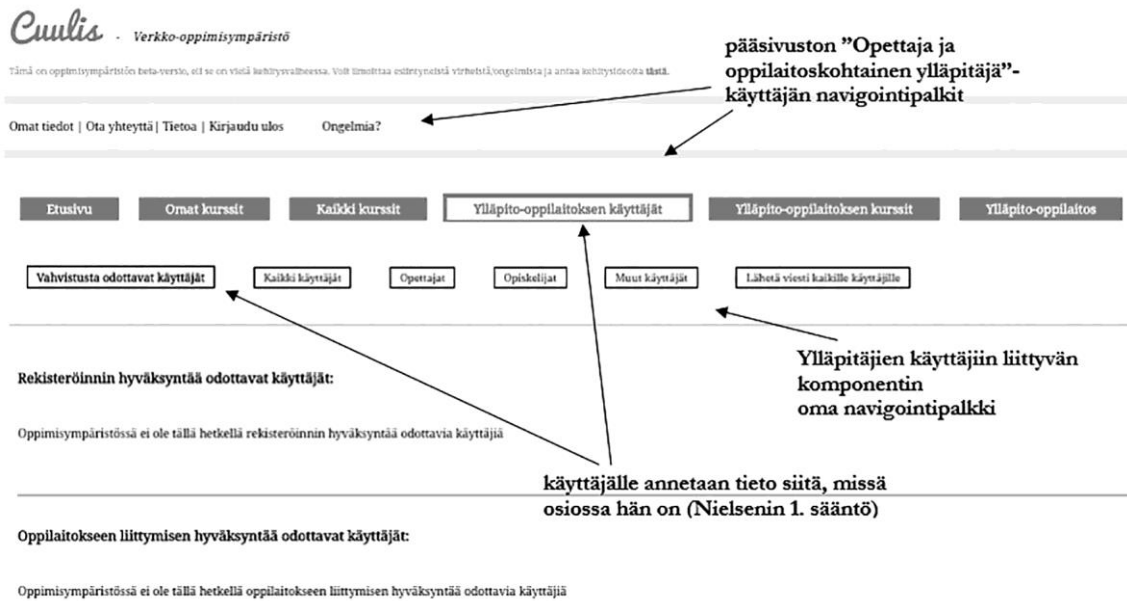
"Yleinen ylläpitäjä"-käyttäjän *Kurssit*-komponentti sisältää kaikki oppimisympäristöön lisätyt kurssit. Kurssin koodin tai nimen klikkaaminen vie käyttäjän kyseisen kurssin kurssisivuston Aloitus-komponenttiin. "Oppilaitoskohtainen ylläpitäjä"- ja "Opettaja ja oppilaitoskohtainen ylläpitäjä"-käyttäjän *Ylläpito-oppilaitoksen kurssit*-komponentti sisältää puolestaan ne kurssit, jotka liittyvät siihen oppilaitokseen, jonka oppilaitoskohtaiseksi ylläpitäjäksi käyttäjä on merkitty. Kurssin nimen tai koodin klikkaaminen vie käyttäjän kyseisen kurssin kurssisivuston Aloitus-komponenttiin vain, jos kyseessä ei ole

ensimmäinen kerta. Muussa tapauksessa käyttäjä siirtyy kurssisivuston Kurssin avainkomponenttiin.

Sekä Kurssit- että Ylläpito-oppilaitoksen kurssit-komponentissa kurssin vastuopettajan nimen klikkaaminen vie käyttäjän vastuopettajan tietoja sisältävään Käyttäjäprofiili-komponenttiin. Kunkin kurssin kohdalla on näkyvissä ajankohta, jolloin kurssi on lisätty oppimisympäristöön. Komponenteissa käyttäjillä on toiminto, jonka avulla he valitsevat kurssit, jotka haluavat poistaa.

Käyttäjät- tai *Ylläpito-oppilaitoksen käyttäjät-*komponentti sisältää aiemmin esiteltyt, kuvan 32 mukaiset osakomponentit. ”Oppilaitoskohtainen ylläpitäjä”- ja ”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjän osakomponentit sisältävät luetteloita niistä käyttäjistä, jotka liittyvät siihen oppilaitokseen, jonka ylläpitäjäksi käyttäjä on merkitty. ”Yleinen ylläpitäjä”-käyttäjän osakomponentit sisältävät puolestaan luettelon kaikista oppimisympäristön käyttäjistä. Pääkomponentin valinta ohjaa käyttäjän Vahvistusta odottavat käyttäjät-osakomponenttiin, joka sisältää luettelon niistä käyttäjistä, jotka odottavat joko rekisteröinnin tai oppilaitokseen liittymisen hyväksyntää. Tässä osakomponentissa käyttäjällä on toiminto, jonka avulla hän voi joko hyväksyä tai hylätä käyttäjien rekisteröitymisiä ja oppilaitokseen liittymisiä. Oppilaitokseen liittymisen hyväksyminen tai hylkäys lähettää liittyneen käyttäjän sähköpostiin ilmoituksen tästä. Hyväksymisen yhteydessä kyseisen käyttäjän oppilaitostietoihin lisätään kyseinen oppilaitos. Rekisteröitymisen hyväksyntä puolestaan lähettää rekisteröityneen käyttäjän sähköpostiin vahvistuslinkin, ja samalla tietokantaan tallennetaan käyttäjän rekisteröinti hyväksytyksi. Mikäli käyttäjän rekisteröityminen hylätään, rekisteröityneen käyttäjän sähköpostiin lähetetään ilmoitus tästä, ja kyseisen käyttäjän tiedot poistetaan tietokannasta. Opettajat-, Opiskelijat- ja Muut-osakomponentit sisältävät luettelon niistä käyttäjistä, jotka rekisteröityessään valitsivat rooliksi kyseisen osakomponentin nimen. Viestin lähetys kaikille käyttäjille-osakomponentissa käyttäjä voi lähettää viestin yhtäaikaaisesti kaikille niille käyttäjille, joiden rekisteröinti on hyväksytty ja jotka ovat vahvistaneet rekisteröitymisensä. Jokaisessa osakomponentissa käyttäjän nimen klikkaaminen vie käyttäjän tietoja sisältävään Käyttäjäprofiili-komponenttiin. Vahvistusta odottavat käyttäjät-osakomponenttia lukuunotta-

matta osakomponenteissa käyttäjä voi valita ne käyttäjät, jotka hän haluaa poistaa oppimisympäristöstä. Kuvassa 42 on Vahvistusta odottavat käyttäjät-osakomponenttiin liittyvä näkymä ”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjän käyttöliittymässä.



Kuva 42. "Opettaja ja oppilaitoskohtainen ylläpitäjä"-käyttäjän Ylläpito-oppilaitoksen käyttäjät-komponenttiin liittyvä näkymä.

”Yleinen ylläpitäjä”-käyttäjän *Oppilaitokset*-komponentti sisältää luettelon niistä oppilaitoksista, jotka on lisätty oppimisympäristöön. Komponentissa käyttäjällä on toiminto, jonka avulla hän voi valita ne oppilaitokset, jotka hän haluaa poistaa oppimisympäristöstä. Oppilaitoksen nimen klikkaaminen vie käyttäjän oppilaitoksen tietoja sisältävään Oppilaitosprofiili-komponenttiin.

Oppilaitosprofiili-komponentissa käyttäjällä on toiminnot, joiden avulla hän voi muokata oppilaitoksen nimeä ja logoa sekä lisätä ja poistaa ylläpitäjiä. Uusi logo, eli kuvatiedosto, tallennetaan palvelimelle, ja tässä yhteydessä tarkistetaan, että ei-toiminnallisissa vaatimuksissa esitellyt kuvatiedostoihin liittyvät ehdot täyttyvät. Logon vaihtaminen poistaa

vanhan kuvatiedoston palvelimelta. Oppilaitoskohtaisen ylläpitäjän lisääminen toteutetaan yksikenttäisellä lomakkeella, johon käyttäjä voi syöttää haluamansa käyttäjän oppimisympäristöön rekisteröimän sähköpostiosoitteen. Jotta ylläpitäjän lisääminen onnistuu, kyseisen käyttäjän tulee olla rekisteröitynä oppimisympäristöön siten, että rekisteröinti on sekä hyväksytty että vahvistettu. Lisäksi käyttäjän tulee olla liittyneenä kyseiseen oppilaitokseen. Onnistuneen ylläpitäjän lisäämisen, ja vastaavasti ylläpitäjän poistamisen, jälkeen kyseiselle käyttäjälle lähetetään sähköpostiin ilmoitus tästä. Lisäksi Oppilaitosprofiili-komponentissa näkyvän ylläpitäjän nimen klikkaaminen vie käyttäjän kyseisen ylläpitäjän tietoja sisältävään Käyttäjaprofiili-komponenttiin.

Cuulis - Verkko-oppimisympäristö

Tämä on oppimisympäristön beta-versio, eli se on vielä kehitysvaiheessa. Voit ilmoittaa esiintyneistä virheistä/ongelmista ja antaa kehitysideoita [tästä](#).

Omat tiedot | Tietoa | Kirjautu ulos

pääsivuston "Yleinen ylläpitäjä"-käyttäjän navigointipalkit

Etusivu | Oppilaitokset | Käyttäjät | Kurssit | Palvelimella olevat tiedostot

Oppimisympäristössä olevat oppilaitokset:

Klikkaamalla oppilaitoksen nimeä pääset tarkastelemaan ja muokkaamaan sen tietoja.

Hae oppilaitos:

Valitse kaikki Oppilaitos

<input type="checkbox"/>		TAPIOLAN LUKIO Tapiolan lukio
<input type="checkbox"/>		Testikoulu

X Poista oppilaitos

+ Lisää uusi oppilaitos

mahdollisesti epäselvien symbolien yhteydessä selventävä teksti (Nielsenin 2. sääntö)

Kuva 43. "Yleinen ylläpitäjä"-käyttäjän Oppilaitokset-komponenttiin liittyvä näkymä.

Oppilaitokset-komponentin kautta käyttäjän on mahdollista siirtyä *Oppilaitoksen lisäys*-komponenttiin, jossa käyttäjä voi lisätä oppimisympäristöön uuden oppilaitoksen. Uuden

oppilaitoksen lisääminen toteutetaan lomakkeella, johon käyttäjä voi syöttää oppilaitoksen nimen, ladata sen logon ja lisätä oppilaitokseen oppilaitoskohtaisen ylläpitäjän. Uuden oppilaitoksen lisääminen toteutetaan vastaavalla tavalla kuin Oppilaitosprofiili-komponentin oppilaitoksen tietojen muokkaukseen liittyvä toiminto. Kuvassa 43 on Oppilaitokset-komponenttiin liittyvä näkymä ”Yleinen ylläpitäjä”-käyttäjän käyttöliittymässä.

”Yleinen ylläpitäjä”-käyttäjän Palvelimella olevat tiedostot-komponentti sisältää luettelon kaikista käyttäjien palvelimelle ladatuista tiedostoista. Tiedostot ryhmitellään sen mukaan, onko kyseessä opettajan lisäämä yleinen kurssitiedosto vai opiskelijan ryhmätöiden yhteydessä lisäämä tiedosto. Tiedoston nimen klikkaaminen avaa käyttäjälle kyseisen tiedoston. Käyttäjällä on toiminto, jonka avulla hän voi valita ne tiedostot, jotka haluaa poistaa. Lisäksi komponentissa on luettelo niistä käyttäjistä, jotka ovat lisänneet profiilikuvan. Käyttäjän nimen klikkaaminen avaa kyseisen profiilikuvan. Käyttäjän profiilikuvaa ei voi poistaa tässä komponentissa, vaan se vaatii siirtymisen käyttäjän tietojen sisältävään Käyttäjäprofiili-komponenttiin, jossa profiilikuvan poisto on mahdollista.

”Oppilaitoskohtainen ylläpitäjä”- ja ”Opettaja ja oppilaitoskohtainen ylläpitäjä”-käyttäjän *Ylläpito-oppilaitos*-komponentti sisältää tiedon sen oppilaitoksen nimestä, logosta ja oppilaitoskohtaisista ylläpitäjistä, jonka oppilaitoskohtaiseksi ylläpitäjäksi käyttäjä on merkittänyt. Käyttäjällä on komponentissa toiminto, jonka avulla hän voi muokata näitä oppilaitoksen tietoja. Toiminto toteutetaan vastaavalla tavalla kuin ”Yleinen ylläpitäjä”-käyttäjän Oppilaitosprofiili-komponentin oppilaitoksen tietojen muokkaukseen liittyvä toiminto.

Käyttäjäprofiili-komponenttiin siirrytään siis eri yhteyksissä käyttäjän nimeä klikkaamalla, ja se sisältää tietoja siitä käyttäjästä, jonka nimeä on klikattu. Käyttäjän tiedoista näytetään kaikille käyttäjille kyseisen käyttäjän profiilikuva, etunimi, sukunimi, rooli ja oppilaitokset, johon käyttäjä on liittynyt. ”Oppilaitoskohtainen ylläpitäjä”-, ”Opettaja ja oppilaitoskohtainen ylläpitäjä”- ja ”Yleinen ylläpitäjä”-käyttäjille komponentissa näkyy lisäksi käyttäjän sähköpostiosoite. Lisäksi heillä on toiminto, jonka avulla he voivat muokata kaikkia käyttäjään liittyviä tietoja roolia ja oppilaitoksia lukuunottamatta, sekä toiminto, jonka avulla he voivat poistaa käyttäjän oppimisympäristöstä. Käyttäjän tietojen

muokkaukseen liittyvä toiminto toteutetaan vastaavalla tavalla kuin Omat tiedot-komponentin omien tietojen muokkaukseen liittyvä toiminto. Jokaisella käyttäjällä Käyttäjäprofiili-komponentissa on toiminto, jonka avulla he voivat siirtyä Viestin lähetys käyttäjälle-komponenttiin, jossa käyttäjä voi lähettää viestin kyseiselle käyttäjälle.

Viestin lähetys käyttäjälle-komponenttiin siirrytään eri yhteyksissä, kun käyttäjä on valinnut oppimisympäristön tarjoaman toiminnon yksilöllisen viestin lähettämiseen. Komponentti sisältää lomakkeen, jonka avulla käyttäjä voi lähettää viestin valitsemalleen käyttäjälle.

6.2.3 Kurssisivuston komponentit

Kurssisivuston *Kurssin avain*-komponentti sisältää lomakkeen, johon käyttäjä voi syöttää kyseiseen kurssiin liittyvän avaimen. Lomakkeen tallennuksen yhteydessä tarkistetaan, vastaako käyttäjän antama syöte tietokantaan tallennettua avainta. Tässä huomioidaan isojen ja pienten kirjainten ero, eli jos kurssin avain on esimerkiksi ”Testi”, niin käyttäjän syöte ”testi” ei kelpaa. Käyttäjän syötteen ollessa oikea, käyttäjä ohjataan kurssisivuston Aloitus-komponenttiin. Jos käyttäjän rooli on ”Opiskelija”, hänet lisätään kyseisen kurssin osallistujaluetteloon. Samalla tarkistetaan, onko kurssiin lisätty Itsenäinen työ-projekti, ja jos on, kyseiselle käyttäjälle lisätään projektiin liittyvä tehtäväluettelo.

Aloitus-komponentti sisältää ilmoitustaulun. ”Opettaja”-käyttäjällä on toiminto, jonka avulla hän voi muokata ilmoitustaulun sisältöä. Ilmoitustaulun yläpuolella ”Opettaja”-käyttäjällä on näkyvissä kurssin avain.

Osallistujat-komponentissa on luettelo kurssin opettajista ja opiskelijoista. Kunkin osallistujan nimen yhteydessä on toiminto, jonka avulla käyttäjät voivat siirtyä *Viestin lähetys käyttäjälle*-komponenttiin, jossa he voivat lähettää viestin haluamalleen kurssin osallistujalle. Lisäksi kunkin osallistujan nimen klikkaaminen vie käyttäjän *Käyttäjäprofiili*-komponenttiin, jossa käyttäjä voi tarkastella kyseisen osallistujan käyttäjäprofiilia. Kurssisivuston Viestin lähetys käyttäjälle- ja Käyttäjäprofiili-komponentit ovat samat kuin pääsivuston vastaavat komponentit.

Luentosessio-komponentti sisältää Etusivu- ja Kysymykset/kommentit-osakomponentit. Pääkomponentin valinta vie käyttäjän suoraan *Etusivu*-osakomponenttiin, joka sisältää luennon oman ilmoitustaulun, luennon aiheen ja päivämäärän. ”Opettaja”-käyttäjällä on tässä osakomponentissa toiminto, jonka avulla hän voi muokata ilmoitustaulun sisältöä, sekä toiminto, jonka avulla hän voi muokata luennon aihetta ja päivämäärää.

Kysymykset/kommentit-osakomponentin sisältämä toiminnallisuus on oletusarvoisesti piilotettu, ja ”Opettaja”-käyttäjällä on toiminto, jonka avulla hän voi aktivoida sen. Toiminnallisuuden ollessa aktivoitu, Kysymykset/kommentit-osakomponentin valinta avaa käyttäjälle uuden sivun selaimen uuteen välilehteen. Tällä sivulla kaikille käyttäjille on näkyvissä kaikki lähetetyt kysymykset ja kommentit siten, että viimeisin lähetetty viesti on luettelon ylimpänä. Lisäksi kunkin kysymyksen/kommentin edessä on nähtävissä ajankohta, jolloin se on lähetetty. ”Opiskelija”-käyttäjällä on kysymys/kommentti-luettelon alapuolella lomake, johon hän voi syöttää haluamansa kysymyksen tai kommentin. ”Opettaja”-käyttäjällä on puolestaan toiminto, jonka avulla hän voi valita ne kysymykset/kommentit, jotka haluaa poistaa, ja toiminto, jonka avulla hän voi poistaa kerrallaan. ”Opettaja”-käyttäjällä on myös mahdollisuus tarkastella, kuka kunkin kysymyksen/kommentin on esittänyt. Tämä tapahtuu siten, että käyttäjä klikkaa haluamaansa kysymystä/kommenttia, minkä jälkeen käyttäjälle näytetään lähettäneen käyttäjän nimi. Tässä yhteydessä käyttäjällä on toiminto, jonka avulla hän voi poistaa kyseisen kysymyksen/kommentin. Edellä kuvattujen ominaisuuksien vuoksi ”Opiskelija”-käyttäjän kysymyksen/kommentin tallennuksen yhteydessä tietokantaan tallennetaan lähetetyn kysymyksen/kommentin lisäksi ajankohta, jolloin viesti lähetetään, sekä käyttäjä, joka viestin lähettää. ”Opettaja”-käyttäjällä on lisäksi toiminto, jonka avulla hän voi piilottaa koko Kysymykset/kommentit-osakomponentin toiminnallisuus. Tämän toiminnon valinnan jälkeen sen tehneen käyttäjän välilehti suljetaan automaattisesti. Samanaikaisesti avoimien ”Opiskelija”-käyttäjien näkymissä piilotetaan lähetetyt viestit ja estetään uusien kysymysten/kommenttien lähetys sekä annetaan tieto tästä. Kuvassa 44 on Kysymykset/kommentit-osakomponenttiin liittyvä näkymä ”Opettaja”-käyttäjän käyttöliittymässä ja kuvassa 45 vastaava ”Opiskelija”-käyttäjän käyttöliittymässä. Välilehti suljetaan näkyvässä olevaa x-symbolia klikkaamalla.

Cuulis - Verkko-oppimisympäristö ✕

Luennon aihe: **Logiikan perusteet** Päivämäärä: **4.3.2016**

Kysymykset/kommentit Piilota toiminto

Yksittäisen kysymyksen/kommentin saat poistettua klikkaamalla sitä. Samalla saat selville sen lähettäneen opiskelijan.

✕ Poista kaikki kysymykset ✕ Valitse poistettavat kysymykset

5.3.2016 12:54:49: **Kumpi operaatio tehtäinkään ensin JA vai TAI?**
 5.3.2016 12:52:40: **Mitä eroa on käsitteillä OR ja XOR?**

Kuva 44. "Opettaja"-käyttäjän Kysymykset/kommentit-osakomponenttiin liittyvä näkymä.

Cuulis - Verkko-oppimisympäristö ✕

Luennon aihe: **Logiikan perusteet** Päivämäärä: **4.3.2016**

Kysymykset/kommentit

Kaikki kysymykset/kommentit lähetetään anonyymisti. Vain opettaja saa halutessaan tiedon lähittäjästä.

5.3.2016 12:54:49: **Kumpi operaatio tehtäinkään ensin JA vai TAI?**
 5.3.2016 12:52:40: **Mitä eroa on käsitteillä OR ja XOR?**

Kysy/kommentoi:

Lähetä

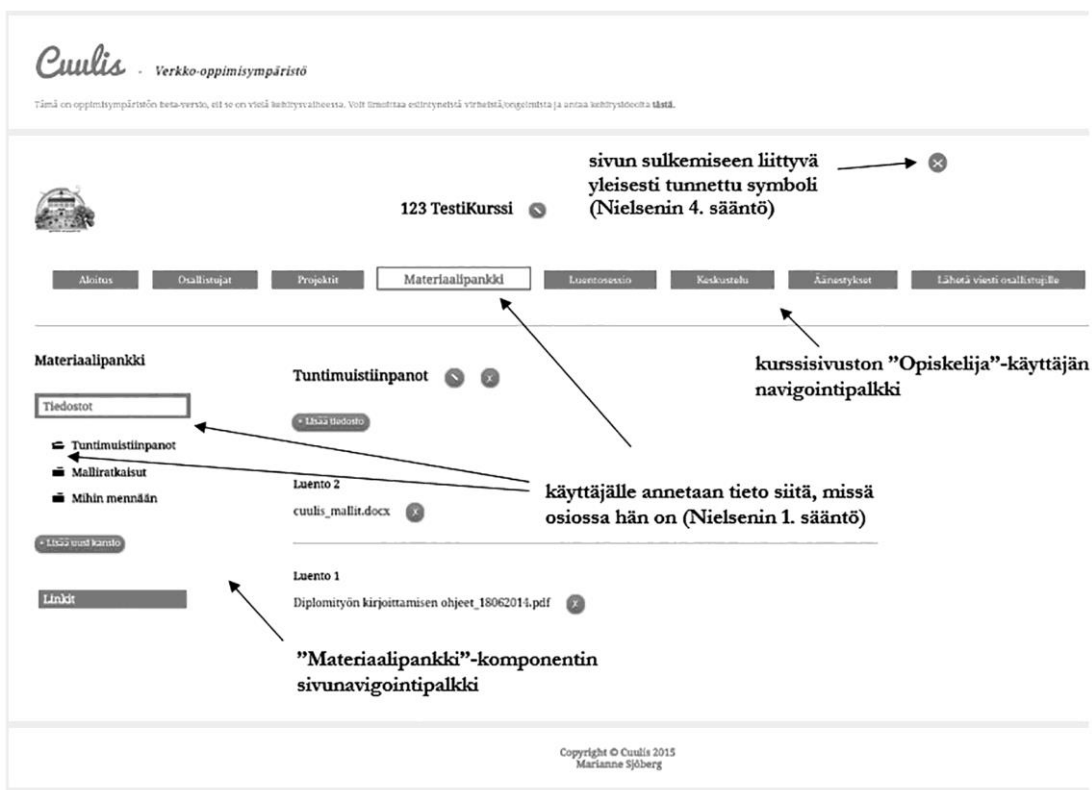
Kuva 45. "Opiskelija"-käyttäjän Kysymykset/kommentit-osakomponenttiin liittyvä näkymä.

Keskustelu-komponentin sisältämä toiminnallisuus on myös aluksi piilotettu, ja ”Opettaja”-käyttäjällä on toiminto, jonka avulla hän voi aktivoida sen. Kun toiminnallisuus on aktivoitu, kaikille käyttäjille on näkyvissä kaikki lähetetyt viestit siten, että viimeisin lähetetty viesti on viestiluettelon ylimpänä. Lisäksi viestin edessä on nähtävissä viestin lähittäneen käyttäjän valitsema nimimerkki sekä ajankohta, jolloin viesti on lähetetty. Viestiluettelon alapuolella on lomake, johon käyttäjä voi syöttää haluamansa nimimerkin ja viestin. Nimimerkki-kentässä on automaattisesti käyttäjän nimi, mutta tätä on siis mahdollista muokata. ”Opettaja”-käyttäjällä on toiminto, jonka avulla hän voi valita ne viestit, jotka haluaa poistaa, ja toiminto, jonka avulla hän voi poistaa kaikki viestit kerrallaan. ”Opettaja”-käyttäjällä on myös mahdollisuus tarkastella, kuka kunkin viestin on esittänyt. Tämä tapahtuu siten, että käyttäjä klikkaa haluamaansa viestiä, minkä jälkeen käyttäjälle näytetään sen lähittäneen käyttäjän nimi. Tässä yhteydessä käyttäjällä on toiminto, jonka avulla hän voi poistaa kyseisen viestin. Edellä kuvattujen ominaisuuksien takia viestin tallennuksen yhteydessä tietokantaan tallennetaan viestin ja nimimerkin lisäksi myös ajankohta, jolloin viesti lähetetään, sekä käyttäjä, joka viestin lähettää. ”Opettaja”-käyttäjällä on lisäksi toiminto, jonka avulla hän voi piilottaa Keskustelu-komponentin toiminnallisuus. Toiminnon valinnan jälkeen kaikkien käyttäjien näkymissä piilotetaan lähetetyt viestit ja estetään uusien viestien lähetys sekä annetaan tieto tästä.

Keskustelu-komponentissa sekä Luentosessio-komponentin Kysymykset/Kommentit-osakomponentissa käyttäjien lähettämät, tietokantaan tallennetut viestit haetaan näkyviin JavaScriptin avulla, jotta käyttäjän ei tarvitse päivittää sivua saadakseen uusimmat viestit näkyviin. Lisäksi komponenttien sisältämien toiminnallisuuksien piilottaminen toteutetaan JavaScriptin avulla, jotta kysymykset/kommentit ja viestit piilotetaan sekä uusien lähettäminen estetään reaaliaikaisesti käyttäjien näkymissä.

Materiaalipankki-komponentti sisältää Tiedostot- ja Linkit-osakomponentit. Pääkomponentin valinnan jälkeen käyttäjän on vielä valittava haluamansa osakomponentti. *Tiedostot*-osakomponentissa käyttäjän on lisäksi valittava haluamansa kansio, minkä jälkeen käyttäjälle avautuu luettelo tähän kansioon liittyvistä tiedostoista. Tiedoston nimen klikkaaminen avaa kyseisen tiedoston. ”Opettaja”-käyttäjällä on toiminto, jonka avulla hän voi lisätä uusia tiedostoja. Tämä tapahtuu siten, että hän valitsee joko olemassa olevan

kansion tai lisää uuden. Uuden kansion lisääminen tapahtuu lomakkeella, johon käyttäjä voi syöttää kansion nimen. Tallentamisen jälkeen uusi kansio avautuu, ja käyttäjällä on toiminto, jonka avulla hän voi lisätä siihen tiedostoja. Tiedostojen lisääminen tapahtuu lomakkeella, johon käyttäjä voi syöttää tiedostoon liittyvän kuvauksen ja ladata laitteeltaan haluamansa tiedoston. Tiedostot tallennetaan palvelimelle, ja tässä yhteydessä tarkistetaan, että ei-toiminnallisten vaatimusten yhteydessä esitellyt tiedostoihin liittyvät ehdot täyttyvät. ”Opettaja”-käyttäjällä on myös toiminto, jonka avulla hän voi poistaa sekä yksittäisiä tiedostoja että koko kansion kerrallaan, jolloin kaikki kansioon liittyvät tiedostot poistetaan automaattisesti. Kuvassa 46 on Tiedostot-osakomponenttiin liittyvä näkymä ”Opiskelija”-käyttäjän käyttöliittymässä.



Kuva 46. ”Opiskelija”-käyttäjän Tiedostot-osakomponenttiin liittyvä näkymä.

Linkit-osakomponentin valinta avaa käyttäjälle luettelon sinne lisätyistä linkeistä. ”Opettaja”-käyttäjällä on toiminto linkkien lisäämiseen. Uuden linkin lisääminen toteutetaan

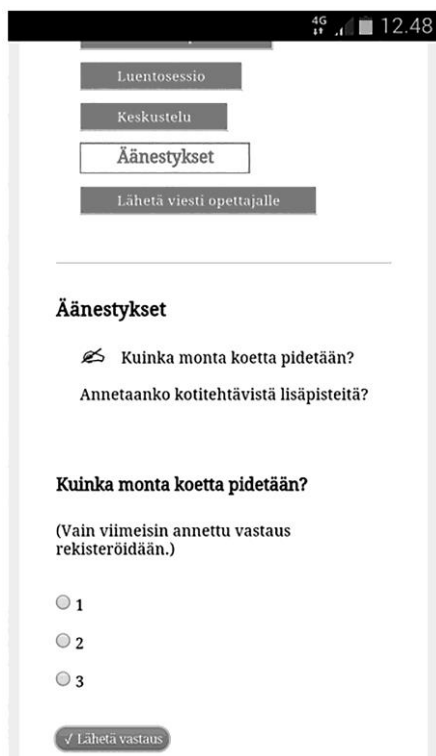
lomakkeella, johon käyttäjä voi syöttää kuvauksen linkistä ja linkin URL-osoitteen täydellisessä muodossaan. Käyttäjän klikatessa osakomponentissa luettua linkkiä se avautuu käyttäjän selaimen uuteen välilehteen.

Äänestykset-komponentin valinnan jälkeen käyttäjälle avautuu luettelo sinne lisättyjen äänestysten kysymyksistä. Äänestyskysymyksen klikkaaminen avaa kyseisen äänestyksen äänestysivun. on vielä valittava haluamansa äänestys. Komponentissa ”Opettaja”-käyttäjällä on toiminto uuden äänestyksen lisäämiseen. Uuden äänestyksen lisääminen toteutetaan kaksiosaisella lomakkeella. Ensimmäiseen lomakkeeseen käyttäjä voi syöttää äänestykseen liittyvän kysymyksen ja valita vastausvaihtoehtojen lukumäärän. Ensimmäisen lomakkeen tallennus avaa käyttäjälle toisen lomakkeen, jossa on valitun vastausvaihtoehtojen lukumäärän mukainen määrä kenttiä, joihin käyttäjä voi syöttää kunkin vaihtoehdon kuvauksen. Tämän lomakkeen tallennus lisää lopullisesti äänestyksen. Mikäli käyttäjä keskeyttää uuden äänestyksen lisäämisen ennen jälkimmäisen lomakkeen tallentamista, niin kaikki äänestykseen liittyvät tiedot poistetaan tietokannasta. Uusi äänestys ei ole alussa näkyvissä ”Opiskelija”-käyttäjille, vaan ”Opettaja”-käyttäjän on ensin valittava toiminto, jonka avulla äänestys asetetaan kaikille näkyväksi. Myöhemmin ”Opettaja”-käyttäjällä on toiminto, jonka avulla hän voi piilottaa äänestyksen ”Opiskelija”-käyttäjiltä. Jokaisen äänestyksen kohdalla ”Opettaja”-käyttäjällä on myös toiminto, jonka avulla äänestys voidaan sulkea, ja jos äänestys on suljettu, käyttäjällä on toiminto, jonka avulla hän voi aktivoida sen uudelleen. Niin kauan kun äänestys on aktivoitu ja näkyvissä, kaikki käyttäjät voivat osallistua siihen valitsemalla haluamansa vaihtoehdon ja lähettämällä vastauksensa. Kuhunkin äänestykseen voi siis osallistua useampaan kertaan, mutta vastaukset tallennetaan tietokantaan siten, että käyttäjän tietoja päivitetään lähetetyn vastauksen mukaisesti. Näin vain viimeisin lähetetty vastaus on tallennettu tietokantaan. Äänestyksen ollessa aktivoitu ”Opettaja”-käyttäjällä on toiminto, jonka avulla hän voi tarkastella äänestystilannetta. Kun äänestys on suljettu ja näkyvissä, kaikki käyttäjät näkevät lopullisen äänestystuloksen. Jos äänestys on suljettu mutta piilotettu, niin vain ”Opettaja”-käyttäjät näkevät äänestystuloksen. Äänestysnäkyvässä ”Opettaja”-käyttäjällä on myös toiminto, jonka avulla hän voi poistaa kyseisen äänestyksen. Kuvassa 47 on *Äänestykset*-komponenttiin liittyvä näkymä ”Opiskelija”-käyttäjän käyttöliitty-

mässä, kun sitä katsotaan tietokoneella, ja kuvassa 48 vastaava näkymä, kun sitä katsotaan älypuhelimella. Kuvista hahmottuu se, miten responsiivisuus on huomioitu Cuulis-oppimisympäristön käyttöliittymän toteutuksessa.



Kuva 47. "Opiskelija"-käyttäjän Äänestykset-komponenttiin liittyvä tietokonenäkymä.



Kuva 48. "Opiskelija"-käyttäjän Äänestykset-komponenttiin liittyvä älypuhelinäkymä.

Projektit-komponentti sisältää Itsenäiset työt- ja Ryhmätyöt-osakomponentit. Pääkomponentin valinnan jälkeen käyttäjän on valittava haluamansa osakomponentti. Osakomponentin valinnan jälkeen käyttäjälle avautuu luettelo sinne lisätyistä projekteista. Haluamansa projektin nimen klikkaaminen avaa käyttäjälle kyseisen projektin projektisivun. Kullakin projektisivulla ”Opettaja”-käyttäjällä on toiminto, jonka avulla hän voi muokata projektin nimeä, ja toiminto, jonka avulla hän voi poistaa projektin. Molemmissa osakomponenteissa ”Opettaja”-käyttäjällä on toiminto uuden projektin lisäämiseen.

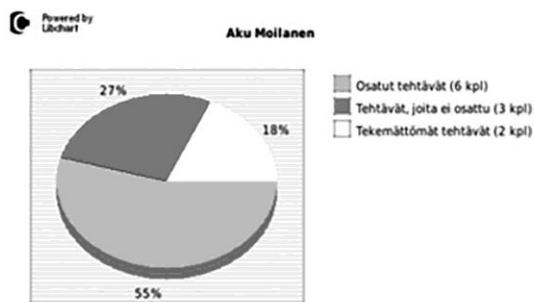
Itsenäiset työt-osakomponentissa uuden projektin lisääminen tapahtuu siten, että käyttäjälle avautuu lomake, johon käyttäjä voi syöttää projektin nimen. Lomakkeen tallentamisen jälkeen kyseinen projektisivu aukeaa. Uuden Itsenäinen työ-projektin lisäämisen yhteydessä jokaiselle kurssin opiskelijalle lisätään oma tehtäväluettelo. Projektisivun yläosassa on projektiin liittyvä ilmoitustaulu, jonka yhteydessä ”Opettaja”-käyttäjällä on toiminto, jonka avulla hän voi muokata sen sisältöä.

”Opiskelija”-käyttäjän Itsenäinen työ-projektin projektisivulla ilmoitustaulun alapuolella on tehtäväluettelon merkintöjen mukaan automaattisesti päivittyvä sektoridiagrammi. Sektoridiagrammissa se alue, joka vastaa osattuja tehtävien osuutta on vihreä, se alue, joka vastaa yritettyjen mutta ei osattujen tehtävien osuutta on punainen, ja se alue, joka kuvastaa tekemättömien tehtävien osuutta on valkoinen. Sektoridiagrammin toteutuksessa hyödynnetään PHP:n valmista Libchart-kirjastoa. Sektoridiagrammin alla on kaikkien tehtävien lukumäärä sekä tehtyjen tehtävien lukumäärä ja osuus kaikista tehtävistä. Tämän jälkeen ”Opiskelija”-käyttäjän projektisivulla on projektiin liittyvä tehtäväluettelo. Tehtäväluettelon otsikot ovat *Osasin*, *Yritin*, *mutta en osannut*, *Haluan*, *että tehtävä käydään yhdessä läpi* ja *Kommentti*. Tehtäväluettelon tallentamattomien tehtävien solut ovat muokattavissa olevia kenttiä, johon käyttäjä voi tehdä merkintöjä. Tehtäväluettelon lopussa käyttäjällä on toiminto, jonka avulla hän voi tallentaa luettelon. Tallennuksen jälkeen ne taulukon rivit, joissa *Osasin*-otsikon kohdalla on merkintä, muuttuvat vihreäksi, ja ne rivit, joissa *Yritin*, *mutta en osannut*-otsikon kohdalla on merkintä, muuttuvat punaiseksi. Niihin soluihin, joissa on merkintä *Haluan*, *että tehtävä käydään yhdessä läpi*-otsikon kohdalla, lisätään tähän viittaavaa HTML-symboli. Lisäksi ne taulukon rivit, joissa on merkintä vain *Haluan*, *että tehtävä käydään yhdessä läpi*- tai *Kommentti*-otsikon

kohdalla, muuttuvat keltaiseksi Osasin- ja Yritin, mutta en osannut-otsikkoihin viittaa-
vissa soluissa. Lisäksi soluihin merkitään !-merkki. Tällä pyritään korostamaan sitä, että
käyttäjä ei ole vielä merkinnyt, että olisi tehnyt tehtävän. Tallennettujen tehtävien yhtey-
dessä käyttäjällä on toiminto, jonka avulla hän voi muokata kyseistä tehtäviä. Muok-
kaustoiminnon klikkaaminen muuttaa kyseisen rivin solut jälleen tekstikentiksi, joissa
oletusarvona on käyttäjän vanhat merkinnät. Kuvassa 49 on näkymä ”Opiskelija”-käyttä-
jän Itsenäinen työ-projektin projektisivusta.

Testi3000

Materiaalipankista löytyy tehtäviin liittyvää materiaalia, kuten tuntimuistiinpanot ja linkit.



Tehtäviä yhteensä: 11 kpl. Tehyjä tehtäviä: 81.8 %.

Tehtävä	Osasin	Yritin, mutten osannut	Haluan käydä tunnilla läpi	Kommentti
Logiikan perusteet, oppikirja s. 5-7				
1	✓			
2	✓			
3	✓			
8		×	!	Tee myöhemmin uudelleen!
9	✓			
12		×		
Algoritmien perusteet, oppikirja s. 8-12				
13	✓		!	
15	✓			
22		×		
23	!	!	!	En ymmärrä tehtävänantoa.
24				

✓ Tallenna

Kuva 49. "Opiskelija"-käyttäjän Itsenäinen työ-projektin projektisivunäkymä.

Cuulis - Verkko-oppimisympäristö

Tämä on oppimisympäristön beta-versio, eli se on vielä kehitysvaiheessa. Voit ilmoittaa odotettavista virheistä, ongelmista ja antaa kehitysohjeita [täällä](#).

OH01 Ohjelmoinnin alkeet

Alustus Osallistajat **Projektit** Materiaalipankki Luontosessio Keskustelu Äänestykset Lähetä viesti osallistujille

Projektit

Itsenäiset työt

Testi3000
Testi4000

11000 uusia projekteja

Ryhmätyöt

kurssisivuston "Opettaja"-käyttäjän navigointipalkki

Materiaalipankista löytyy tehtäviä tiimiyvä materiaalia, kuten tuntemustilipaneelit ja tinaat.

muokkaukseen liittyvä yleisesti tunnettu symboli (Nielsenin 4. sääntö)

käyttäjälle annetaan tieto siitä, missä osiossa hän on (Nielsenin 1. sääntö)

"Projektit"-komponentin sivunavigointipalkki

Tehtäviä yhteensä: 11 kpl.

Tehtävä	Osattu	Yritetty, muttei osattu	Toivottu yhdessä läpikäytyväksi	Kommentoitu
Logiikan perusteet, oppikirja s. 5-7				
1	2	0	0	0
2	2	0	0	0
3	2	0	0	0
8	1	1	1	1
9	2	0	0	0
12	0	2	0	0
Algoritmien perusteet, oppikirja s. 8-12				
13	2	0	1	0
15	0	1	1	0
22	0	2	0	1
23	0	1	2	1
24	0	0	0	0

Muokkaa tehtäväluetteloa

Kuva 50. "Opettaja"-käyttäjän Itsenäinen työ-osakomponenttiin liittyvä näkymä.

"Opettaja"-käyttäjän Itsenäinen työ-projektin projektisivulla ilmoitustaulun alapuolella on tehtäväluettelo, jota hän voi muokata. Tehtäväluettelon muokkaus toteutetaan siten, että käyttäjä voi lisätä siihen otsikon tai tehtävän joko luettelon loppuun tai käyttäjän haluamaan kohtaan. Lisäksi käyttäjällä on toiminto, jonka avulla hän voi liittää tehtäväluetteloon jo aiemmin luomansa tehtäväluettelon. Tämän toiminnon valitessaan käyttäjälle avautuu luettelo niistä kursseista, joissa hän on vastuopettajana. Kurssin nimen klikkaaminen avaa käyttäjälle luettelon kyseisen kurssin tehtäväluetteloista, joista käyttäjä voi valita haluamansa. Tallennuksen jälkeen valittuun tehtäväluetteloon liittyvät otsikot ja

tehtävät sisältyvät osaksi muokattavana ollutta tehtävälueteloa. Muokatun tehtäväluetelolon tallentaminen päivittää automaattisesti jokaisen kurssin opiskelijan henkilökohtaisen tehtäväluetelolon. ”Opettaja”-käyttäjän projektisivun tehtäväluetelolon otsikot ovat *Osattu, Yritetty, muttei osattu, Toivottu yhdessä läpikäytyäksi ja Kommentoitu*. Tehtäväluetelolon solujen sisällöt perustuvat ”Opiskelija”-käyttäjien tekemiin tehtäväkohtaisten merkintöjen yhteislukumäärään. Jos jokin näistä soluista sisältää muun luvun kuin nolla, niin kyseisen rivin lopussa on käyttäjällä on toiminto, jonka avulla käyttäjä pääsee tarkastelemaan kyseiseen tehtävään liittyviä tietoja. Tässä käyttäjälle näkyy luettelot opiskelijoista, jotka ovat osanneet kyseisen tehtävän, jotka ovat yrittäneet tehtävää osaamatta sitä, ja jotka ovat toivoneet, että tehtävät käytäisiin yhdessä läpi. Lisäksi näkymässä on luettelo tehtävään tehdyistä kommenteista, ja tieto opiskelijasta, joka kyseisen kommentin on tehnyt. ”Opettaja”-käyttäjällä on myös toiminto, jonka avulla hän voi tarkastella opiskelija-kohtaisia tilastoja. Toiminnon valinta avaa käyttäjälle luettelolon kurssin opiskelijoista, ja kunkin opiskelijan kohdalla on nähtävillä tehtyjen, osattujen ja yritettyjen muttei osattujen osuudet kaikista tehtäväluetelolon tehtävistä. Opiskelijan nimen klikkaaminen avaa käyttäjälle kyseisen ”Opiskelija”-käyttäjän tehtäväluetelonäkymän. Kuvassa 50 on Itsenäiset työt-osakomponenttiin liittyvä näkymä ”Opettaja”-käyttäjän käyttöliittymässä.

Ryhmätyöt-osakomponentissa uuden Ryhmätyö-projektin lisääminen tapahtuu lomakkeella, johon käyttäjä voi syöttää projektin nimen ja valita ryhmien maksimimäärän sekä ryhmäkohtaisen minimi- ja maksimimäärän. Valintakentät toteutetaan jQueryn avulla siten, että minimimäärän valinta piilottaa maksimimäärävaihtoehdoista kaikki valittua minimimäärää pienemmät vaihtoehdot. Tämä tehdään siksi, ettei käyttäjä ei voi tehdä valintaa siten, että ryhmäkohtainen minimimäärä olisi suurempi kuin ryhmäkohtainen maksimimäärä. Lisäksi lomakkeessa käyttäjä voi valita, sallitaanko, että opiskelijat palauttavat ryhmätyötiedoston projektisivulle.

Ryhmätyö-projektin projektisivulla on projektin ilmoitustaulu, jonka yhteydessä ”Opettaja”-käyttäjällä on toiminto, jonka avulla hän voi muokata sen sisältöä. Ilmoitustaulun alapuolella on luettelo ryhmistä ja niihin liittyneistä opiskelijoista. Ryhmäluetelolon lopussa ”Opettaja”-käyttäjälle on näkyvissä myös luettelo opiskelijoista, joilla ei ole vielä

ryhmää. Tässä yhteydessä ”Opettaja”-käyttäjällä on toiminto, jonka avulla hän voi arpoa vailla ryhmää olevat opiskelijat ryhmiin.

Niin kauan kun ryhmäilmoittautuminen on avoinna kaikki käyttäjät voivat lisätä uusia ryhmiä. Lisäksi kunkin ryhmän kohdalla ”Opettaja-käyttäjällä on toiminnot, joiden avulla hän voi sulkea/avata ja poistaa ryhmä, sekä liittää ja poistaa yksittäisiä opiskelijoita ryhmästä. Opiskelijan liittäminen ryhmään tapahtuu siten, että toiminnon valinta avaa käyttäjälle luettelon opiskelijoista, jotka ovat vielä ilman ryhmää. Haluamansa opiskelijan nimen klikkaaminen lisää opiskelijan kyseiseen ryhmään. Lisäksi projektisivulla ”Opettaja”-käyttäjällä on toiminto, jonka avulla hän voi sulkea/avata ryhmäilmoittautumisen sekä suorittaa automaattinen ryhmäjako. Automaattinen ryhmäjako poistaa ne ryhmät, joissa asetettu ryhmäkohtainen minimimääräehto ei täyty, sekä lopulta arpoo kaikki vailla ryhmää olevat opiskelijat ryhmiin ehtojen määrittelemällä tavalla. Koska automaattista ryhmäjakoa ei ole aina mahdollista tehdä asetettujen ehtojen puitteissa, niin toiminnon suorituksen aikana lasketaan suoritukseen kulunut aika. Jos aika ylittää asetetun aikarajan, suoritus keskeytetään, ja käyttäjälle ilmoitetaan, ettei automaattinen ryhmäjako ole mahdollista. ”Opiskelija”-käyttäjällä on puolestaan toiminto, jonka avulla hän voi liittyä haluamaansa ryhmään. Jos ”Opiskelija”-käyttäjä on liittynyt/liitetty ryhmään, kyseisen ryhmän yhteydessä käyttäjällä on toiminnot, joiden avulla hän voi sulkea/avata ryhmän, poistua ryhmästä sekä siirtyä Viesti ryhmälle-osakomponenttiin, jossa hän voi lähettää viestin muille ryhmän jäsenille. Ryhmien lisäämiseen, ryhmään liittymiseen/liittämiseen ja ryhmän sulkemiseen liittyvät toiminnot ovat esillä vain asetettujen ehtojen (ryhmien maksimimäärä, ryhmäkohtainen minimi- ja maksimimäärä) puitteissa.

Kun ryhmäilmoittautuminen on suljettu, Ryhmätyö-projektin projektisivulla näkyy lopullinen ryhmäjako. Jos ryhmätyötiedostojen palautus on sallittu, niin ”Opiskelija”-käyttäjällä on toiminto, jonka avulla hän voi palauttaa ryhmänsä tiedoston, jos kukaan ryhmästä ei ole sitä vielä tehnyt. Palautustoiminto toteutetaan lomakkeella, johon käyttäjä voi syöttää työn nimen ja valita laitteeltaan haluamansa tiedoston. Tiedostot tallennetaan palvelimelle ja tässä yhteydessä tarkistetaan, että ei-toiminnallisten vaatimusten esittelyn yhteydessä mainitut tiedostoihin liittyvät ehdot täyttyvät. Jos ”Opiskelija”-käyttäjän ryhmästä

on palautettu tiedosto, käyttäjä voi klikata sen nimeä, mikä avaa kyseisen tiedoston. Lisäksi tiedoston nimen yhteydessä on toiminto, jonka avulla ”Opiskelija”-käyttäjä voi poistaa ryhmänsä tiedoston. ”Opettaja”-käyttäjä näkee omalla projektisivullaan kaikki palautetut ryhmätyötiedostot. Tiedoston nimen klikkaaminen avaa kyseisen tiedoston, ja nimen yhteydessä on myös ”Opettaja”-käyttäjällä toiminto, jolla tiedoston voi poistaa. Kaikki Ryhmätyö-projektin projektisivulla olevat ryhmien ja ryhmätyötiedostojen luettelot toteutetaan taulukkoina.

”Opettaja”-käyttäjän *Kurssin tietojen muokkaus*-komponentti sisältää kurssin tietojen perusteella valmiiksi täytetyn lomakkeen. Lomake on samankaltainen kuin pääsivuston Kurssin lisäys-komponentin lomake sillä erolla, että Kurssin tietojen muokkaus-komponentin lomakkeessa ei ole mahdollista valita, onko kurssilla muita opettajia. Jos kurssin lisäyksen yhteydessä on valittu, että kurssilla on muita opettajia, tämän tilalla lomakkeessa on valintakenttä, jossa käyttäjä voi valita muut opettajat. Valintakenttä sisältää luettelon niistä roolissa ”Opettaja”-rekisteröityneistä käyttäjistä, jotka ovat liittyneet opilaitokseen, johon kyseinen kurssi on liitetty. Muuten Kurssin tietojen muokkaus-komponentin lomake toteutetaan siis samoin kuin Kurssin lisäys-komponentin lomake.

Lähetä viesti kurssin Ohjelmoinnin alkeet opettajalle

Viesti lähetetään opettajan sähköpostiin. Viestissä näkyy oheiset lähettäjän tiedot.

Lähettäjän nimi:
Aku Moilanen

Lähettäjän sähköpostiosoite (vastaanottaja voi vastata suoraan tähän osoitteeseen):
aku@moilanen.fi

Viesti:

 Lähetä

oma, kuvankäsittelyohjelmalla tehty symboli

Kuva 51. ”Opiskelija”-käyttäjä Lähetä viesti opettajalle-komponenttiin liittyvä näkymä.

”Opettaja”-käyttäjän *Viestin lähetys osallistujille*-komponentti sisältää toiminnon, jonka avulla käyttäjä voi lähettää viestin kaikille kurssin osallistujille, eli opiskelijoille ja vastuopettajalle sekä mahdollisille muille opettajille paitsi itselleen. ”Opiskelija”-käyttäjän *Viestin lähetys opettajalle*-komponentti sisältää puolestaan toiminnon, jonka avulla käyttäjä voi lähettää viestin kurssin vastuopettajalle sekä mahdollisille muille opettajille. Kuvassa 51 on Viestin lähetys opettajalle-komponenttiin liittyvä näkymä.

7 TULOKSET

Tämän diplomityön tulos on edellä esitetyllä tavalla kehitetty uusi konstruktio, aktivoivaa opetusta tukeva verkko-oppimisympäristö. Näin ollen työlle asetettu tutkimustehtävä on suoritettu työlle asetettujen rajausten puitteissa. Uusi Cuulis-oppimisympäristö sijaitsee tämän työn valmistumisen aikaan osoitteessa <http://cuulis.cm8solutions.fi>.

8 JOHTOPÄÄTÖKSET

Opetuksen digitalisaatiolla pyritään tuomaan uusia, teknologian mahdollistamia opetuksen ja oppimisen ratkaisuja oppilaitoksille. Tämän, ja syksyllä 2016 uusien opetussuunnitelmien käyttöönoton myötä, teknologian rooli opetuksessa kasvaa entisestään. Tarve uusille innovatiivisille, käytössä olevaa teknologiaa hyödyntäville oppimisympäristöille korostuu.

Tämä diplomityö tuotti uuden ja ainutlaatuisen konstruktion, aktivoivaa opetusta tukevan, kurssimuotoisille oppilaitoksille suunnatun verkko-oppimisympäristön, jonka nimeksi annettiin Cuulis-oppimisympäristö. Cuulis-oppimisympäristö kehitettiin vesiputousmallia mukaillen suorittamalla mallin sisältämät määrittely-, suunnittelu- ja toteutusvaiheet, jotka kaikki sisältyvät tähän diplomityöhön. Kehitystyö pohjustettiin perehtymällä yleisiin verkko-oppimisympäristön kehityksessä huomioitaviin tekijöihin. Tämän jälkeen määrittelyvaiheessa tämän työn tekijän analyttisen mallintamisen tuloksena muodostuivat Cuulis-oppimisympäristön vaatimukset, jotka toimivat pohjana suunnitteluvaiheelle. Suunnitteluvaiheen tuotosten perusteella Cuulis-oppimisympäristö lopullisesti toteutettiin. Vesiputousmallin sisältämiä testausvaiheita ei suoritettu työlle alussa asetettuihin rajauksiin nojaten.

Cuulis-oppimisympäristö on kehitetty painottaen opiskelijan aktiivista roolia oppimisprosessissa. Cuulis-oppimisympäristöä voidaan käyttää sekä luentojen aikana että oppilaitoksen ulkopuolella tietokoneella ja muilla käytössä olevilla mobiililaitteilla, joilla voidaan ottaa yhteys Internetiin. Cuulis-oppimisympäristö pyrkii siis vastaamaan digitalisoituvan, nykyaikaiseen oppimiskäsitykseen perustuvan opetuksen tarpeisiin, ja siitä voi näin ollen olla suurta käytännön hyötyä kaikille kurssimuotoisille oppilaitoksille.

Cuulis-oppimisympäristön kehitysvaiheet on esitetty tässä työssä mahdollisimman selkeästi ja täsmällisesti siten, että kaikki tämän työn lukijat saavat kattavan kokonaiskuvan

oppimisympäristöstä, sen rakenteesta ja ominaisuuksista. Lisäksi Cuulis-oppimisympäristön käyttöliittymään ja käytettävyyteen on panostettu, jotta sen käyttöönotto olisi helppoa ja sujuvaa.

Cuulis-oppimisympäristö on siirretty pilotointiin Espoon Tapiolan lukioon, jossa sitä käytetään eri vuositasojen kemian kursseilla. Myöhemmin Cuulis-oppimisympäristöä on tarkoitettu kehittää erityisesti pilotoinnin yhteydessä saatujen käyttökokemusten, kuten käyttäjien antamien virheilmoitusten ja kehitysideoiden perusteella. Myös muiden uusien ideoiden ja tarpeiden perusteella Cuulis-oppimisympäristöön voidaan lisätä uusia toiminnallisuuksia ja muita ominaisuuksia. Oppimisympäristön oppilaitoskohtainen tietynasteinen kustomointi on myös mahdollista siten, että eri oppilaitosten erityistoiveet voidaan ottaa huomioon. Cuulis-oppimisympäristö on tällä hetkellä vain suomenkielinen, mutta sen sisältö voidaan myöhemmin kääntää myös muille kielille.

8.1 Työn luotettavuuden arviointi

Tieteessä pyritään aina objektiiviseen totuuteen, eli tutkimustulosten tulisi olla samat riippumatta siitä, kuka tutkimuksen on tehnyt. Todellisuutta voidaan kuitenkin tarkastella monesta eri näkökulmasta, mikä ainakin osittain vaikuttaa tutkimuksen tuloksiin. (Uusitalo 1998: 25.) Konstruktion luominen, eli uuden verkko-oppimisympäristön kehittäminen, perustuu täysin työn tekijän kirjallisuuskatsauksen pohjalta suorittamaan ajatusprosessiin, mikä osaltaan vaikuttaa työn objektiivisuuteen. Diplomityö on kuitenkin toteutettu valitun, päätöksentekometodologisen tutkimusotteen mukaisesti. Päätöksentekometodologinen tutkimusote perustuu nimenomaan konstruktion luomiseen tutkimuksen tekijän analyyttisellä mallintamisella.

Laadullisen tutkimuksen luotettavuutta voidaan parantaa antamalla mahdollisimman tarkka kuvaus tutkimuksen toteuttamisesta (Hirsjärvi, Remes & Sajavaara 1997:227). Tätä varten diplomityössä on määritelty siinä esiintyvät keskeiset käsitteet ja kuvattu mahdollisimman tarkasti verkko-oppimisympäristön kehityksessä suoritettavat vaiheet. Diplomityön luotettavuutta parantaa myös se, että työn tulosta voidaan soveltaa käytäntöön.

8.2 Jatkotutkimusaiheita

Cuulis-oppimisympäristön systemaattinen ohjelmistotestaus on ehdottomasti yksi suositeltava jatkotutkimusaihe. Toinen mielenkiintoinen jatkotutkimusaihe on oppimisympäristön käytettävyyden testaaminen sekä perusteellisella, näkymäkohtaisella asiantuntija-arvioinnilla että loppukäyttäjien suorittamalla käyttäjätestauksella. Asiantuntija-arviointi voidaan suorittaa esimerkiksi Nielsenin käytettävyyssääntöjen avulla. Käyttäjätestauksessa voidaan puolestaan hyödyntää Päivi Sampolan (2008) väitöskirjassaan kehittämää käyttäjän ja kontekstin huomioivaa verkko-oppimisympäristöjen käytettävyyden arviointimenetelmää. Liitteessä 1 on tämän arviointimenetelmän muokattu versio, joka soveltuu Cuulis-oppimisympäristön käytettävyyden arviointiin. Vaikka Cuulis-oppimisympäristö ei sisällä arkaluontoista tietoa, niin erilaisten tietoturvaluotteiden tutkiminen ja niiden korjaaminen on myös varteenotettava jatkotutkimusaihe. Lisäksi yksi jatkotutkimuksen kohde voisi olla se, miten Cuulis-oppimisympäristöä tulisi muokata, jotta se soveltuisi jatkossa enemmän myös monimuotokoulutukseen.

LÄHTEET

- Benyon, D (2010). *Designing Interactive Systems. A Comprehensive Guide to HCI and Interaction Design*. 2. painos. Harlow, Englanti: Pearson Education Limited. ISBN 978-0-321-43533-0.
- Berners-Lee, T., R. Fielding & L. Masinter (2005). *Uniform Resource Identifier (URI): Generic Syntax* [Verkkodokumentti]. The Internet Engineering Task Force (IETF)-järjestön julkaisema Internetiä koskeva standardi. Viitattu [25.2.2016]. Saatavissa: <https://greenbytes.de/tech/webdav/rfc3986.pdf>.
- Boehm, B. W. (1987). Software process management: lessons learned from history. *ICSE '87 Proceedings of the 9th international conference on Software Engineering*, 296-298. Los Alamitos, USA: IEEE Computer Society Press. ISBN 0-89791-216-0.
- Connolly, T. & C. Begg (2005): *Database Systems, A Practical Approach to Design, Implementation, and Management*. 4. painos. Harlow, Englanti: Pearson Education Limited. ISBN 0-321-21025-5.
- Eriksson, H-E. & M. Penker (2000). *UML*. Jyväskylä: Gummerus Kirjapaino Oy. ISBN 951-826-026-5.
- Faisal (2012), M. *Responsive Web Designs are Better Than Ordinary Web Layouts* [online]. Responsiivisten verkkosivujen suunnittelua koskeva blogikirjoitus. Viitattu [25.2.2016]. Saatavissa: <http://graphicdesignjunction.com/2012/12/responsive-web-designs-are-better-than-ordinary-web-layouts/>.
- Fowler, M. & K. Scott (2002). *UML*. Jyväskylä: Docendo Finland Oy. ISBN 951-846-168-6.

- Haikala, I. & T. Mikkonen (2011). *Ohjelmistotuotannon käytännöt*. 12., uudistettu painos. Hämeenlinna: Talentum Media Oy. ISBN 978-952-14-1754-2.
- Haikala, I. & J. Märijärvi (2004). *Ohjelmistotuotanto*. 10., uudistettu painos. Hämeenlinna: Talentum Media Oy. ISBN 952-14-0850-2.
- Hirsjärvi, S., P. Remes & P. Sajavaara (1997). *Tutki ja kirjoita*. 13.-14., osin uudistettu painos. Helsinki: Kirjayhtymä Oy. ISBN 951-26-4184-4.
- ISO/IEC/IEEE 24765 (2010). *Systems and software engineering — Vocabulary*. Sveitsi: ISO. Kansainvälisen standardoimisjärjestö ISO:n (International Organization for Standardization) tarjoama järjestelmien ja ohjelmistojen tuotantoon liittyvä standardoitu sanasto. ISBN 978-0-7381-6205-8.
- Lonka, K. & I. Lonka (1991). *Aktivoiva opetus. Käsikirja aikuisten ja nuorten opettajille*. Helsinki: Kirjayhtymä. ISBN 951-26-3529-1.
- Marcotte, E. (2011): *Responsive Web Design*. New York, USA: A Book Apart. ISBN 978-0-9844425-7-7.
- Microsoft (2011). *Windows Server Administration Fundamentals, Exam 98-365*. USA: Wiley. ISBN 978-0-470-90182-3.
- Munassar, N. M. A. & A. Govardhan (2010). A comparison between five models of software engineering. *International Journal of Computer Science Issues* 7: 5, 94-101. IJCSI Publications (www.IJCSI.org). ISSN (Online) 1694-0814.
- Mäcklin, A. (2012). *Pedagoginen näkökulma verkko-oppimisympäristön suunnittelussa* [Verkkodokumentti]. Ammatillisen opettajankoulutuksen kehittämishanke. Tampereen ammattikorkeakoulu. [Viitattu 15.2.2016]. Saatavissa: https://publications.theseus.fi/bitstream/handle/10024/40732/Macklin_Anu.pdf?sequence=4.

- Nam, C. S. & T. L. Smith-Jackson (2007). Web-based learning environment: a theory-based design process for development and evaluation. *Journal of Information Technology Education* 6, 23–43. Kalifornia, USA: Informing Science Institute ISSN 1547-9714.
- Neilimo, K. & J. Näsi (1980). *Nomoteettinen tutkimusote ja suomalainen yrityksen taloustiede. Tutkimus positivismiin soveltamisesta*. Yrityksen taloustieteen ja yksityisoikeuden laitoksen julkaisuja. Sarja A2: Tutkielmia ja raportteja 12. Tampere: Tampereen yliopisto. ISBN 951-44-1072-6.
- Nielsen, J. (1994). Heuristic Evaluation. Teoksessa: *Usability Inspection Methods*, 25–62. Toim. Jakob Nielsen & Robert L. Mack. New York, USA: John Wiley & Sons. ISBN 0-471-01877-5.
- Opetushallitus (2013). *Opetussuunnitelman perusteiden uudistamisen tavoitteet* [online]. [Viitattu 15.2.2016]. Saatavissa: <http://www.oph.fi/ops2016/tavoitteet>.
- Opetushallitus (2015). *Lukion opetussuunnitelman perusteiden päivittäminen* [online]. [Viitattu 15.2.2016]. Saatavissa: http://www.oph.fi/saadokset_ja_ohjeet/opetussuunnitelmien_ja_tutkintojen_perusteet/lukiokoulutus/lops2016/103/0/lukion_opetussuunnitelman_perusteiden_paivittamisen_suuntaviivat.
- Pantzar, E. & T. Väliharju (1996). *Kohti virtuaalisia oppimisympäristöjä: Avoimet oppimisympäristöt aikuisten ammatillisen oppimisen puitteina*. Helsinki: Ammatti-instituutti 1/96. ISBN 951-97578-0-5.
- Peltomäki, J. & O. Nykänen (2006): *Web-selainohjelmointi*. Jyväskylä: Docendo Finland Oy. ISBN 951-846-286-0.
- Pressman, R. S. (2010). *Software Engineering: A Practitioner's Approach*. 7. painos. New York, USA: McGraw-Hill. ISBN 978-0-07-337597-7.

- Rantala, A. (2005). *Web-ohjelmointi*. Jyväskylä: Docendo Finland Oy. ISBN 951-846-264-X.
- Riihiaho, S (1998). Käytettävyyden arviointi ilman käyttäjiä. *Systeemyö* 4, 4-8. Helsinki: Systeemyöyhdistys SYTYKE ry. ISSN 1237-0525.
- Royce, W. W. (1970). Managing the development of large software systems. *Proceedings, IEEE WESCON*, 1-9 [Verkkodokumentti]. Los Angeles, USA: The Institute of Electrical and Electronics Engineers (IEEE). [Viitattu 10.6.2016]. Saatavissa: <http://www.cs.umd.edu/class/spring2003/cmsc838p/Process/waterfall.pdf>. Julkaistu myös teoksessa *ICSE '87 Proceedings of the 9th international conference on Software Engineering*, 328-338. Los Alamitos, USA: IEEE Computer Society Press. ISBN 0-89791-216-0.
- Räsänen, S (2002). *Verkko-opetuksen tietotekniikkaa – Verkko-oppimisympäristöt* [Verkkodokumentti]. Raportti B/2002/1. Kuopion yliopisto, Tietotekniikan ja sovelletun matematiikan laitos. [Viitattu 6.1.2016]. Saatavissa: <http://www.mit.jyu.fi/ope/kurssit/TIES462/Materiaalit/R%E4s%E4nen.pdf>.
- Sadik, A. (2004). The design elements of web-based learning environments. *International Journal of Instructional Technology & Distance Learning* 1:8, 27-43. USA: Duquesne University. ISSN 1550-6908.
- Sampola, P. (2008). *Käyttäjakeskeisen käytettävyyden arviointimenetelmän kehittäminen verkko-opetusympäristöihin soveltuvaksi*. Vaasan yliopisto. Teknillinen tiedekunta. Väitöskirja. Acta Wasaensia, 192. ISBN 978-952-476-234-2.
- Sanastokeskus TSK ry. *Tietotekniikan termitalkoot* [online]. Tietotekniikan, kielen ja viestinnän ammattilaisten laatimia suosituksia suomenkielisistä tietotekniikan termeistä. Viitattu [25.2.2016]. Saatavissa: <http://www.tsk.fi/tsk/termitalkoot/>.

Silius, K. & A-M. Tervakari (2003). An evaluation of the usefulness of web-based learning environments. The evaluation tool into the portal of Finnish virtual university [Verkkodokumentti]. *International Conference on Network Universities and e-Learning, 8-9 May 2003, Valencia. Espanja*. [Viitattu 26.1.2016]. Saatavissa: http://ans.hsh.no/lu/inf/menu/final/publ/kirsi_silius.pdf.

Sommerville, I (2011). *Software Engineering*. 9. painos. USA: Addison-Wesley. ISBN 978-0-13-703515-1.

The jQuery Foundation (2016). *What is jQuery?* [online]. Voitto tavoittelemattoman järjestön ylläpitämä, verkkosivustojen kehittäjille suunnattu jQueryä käsittelevä verkkosivusto. Viitattu [4.3.2016]. Saatavissa: <https://jquery.com>.

Uusitalo, H (1998). *Tiede, tutkimus ja tutkielma. Johdatus tutkielman maailmaan*. 1.-5. painos. Juva: WSOY. ISBN 951-0-17457-2.

Valtioneuvoston kanslia (2015a). *Ratkaisujen Suomi. Pääministeri Juha Sipilän hallituksen strateginen ohjelma 29.5.2015*. Hallituksen julkaisusarja 10/2015. ISBN 978-952-287-181-7.

Valtioneuvoston kanslia (2015b). *Toimintasuunnitelma strategisen hallitusohjelman kärkihankkeiden ja reformien toimeenpanemiseksi*. Hallituksen julkaisusarja 13/2015. ISBN 978-952-287-195-4.

LIITTEET

LIITE 1. Verkko-opetusympäristön käytettävyyden arviointilomake, muokattu lähteestä (Sampola 2008: 172-174).

Arviointiasteikko: 1 = täysin eri mieltä, 2 = osittain eri mieltä, 3 = en osaa sanoa, 4 = osittain samaa mieltä ja 5 = täysin samaa mieltä.

Käyttäjät, teknologia ja päämäärä	1	2	3	4	5
1. Verkko-oppimisympäristö tukee ja ohjaa kaikissa käyttämissäni toiminnoissa.					
2. Verkko-oppimisympäristöstä löytyy helposti tarvittavat toiminnot.					
3. Verkko-oppimisympäristön käytössä ei ole rajoituksia.					
4. Verkko-oppimisympäristön ominaisuudet tukevat oppimista ja päämäärään pääsemistä.					
5. Verkko-oppimisympäristön rakenne on selkeä ja looginen.					
6. Verkko-oppimisympäristössä ei ole ristiriitoja erilaisten käyttäjän tavoitteiden kanssa.					

Ympäristö ja konteksti	1	2	3	4	5
7. Käsitteitä, sanastoa ja toimintoja käytetään yhdenmukaisesti verkko-oppimisympäristössä.					
8. Ominaispiirteet ja käsitteistö ovat yhdenmukaiset käyttöympäristön kanssa.					
9. Verkko-oppimisympäristö tukee yhteistoiminnallisuutta (esim. keskustelualusta).					
10. Kuvakkeet, painikkeet ja linkit ovat havainnollisia.					
11. Verkko-oppimisympäristössä ei ole teknisiä ongelmia.					
12. Verkko-oppimisympäristö antaa palautetta virheellisestä toiminnosta.					
13. Käyttäjä tietää sijaintinsa verkko-oppimisympäristössä ja toimiiko verkko-oppimisympäristö.					
14. Navigointi verkko-oppimisympäristössä onnistuu hyvin ilman turhia hypyjä					
15. Käyttäjä voi itse vaikuttaa omaan verkko-oppimisympäristön näkymään					
16. Verkko-oppimisympäristön antamat virheilmoitukset ovat selkeitä.					
17. Verkko-oppimisympäristön itseopiskelumahdollisuus tukee oppimista.					

Oppiminen	1	2	3	4	5
18. Verkko-oppimisympäristö opastaa käyttäjää.					
19. Verkko-oppimisympäristön ohjeet ovat helposti ymmärrettäviä.					
20. Dokumenttien liittäminen onnistuu vaikeuksitta.					
21. Verkko-oppimisympäristön käytön oppiminen (sisäistäminen) onnistuu ilman ponnisteluja.					
22. Verkko-oppimisympäristö tukee ja auttaa löytämään ratkaisun ongelmatilanteessa					
23. Verkko-oppimisympäristö muistaa mitkä harjoitukset käyttäjä on jo tehnyt.					
24. Oppimisprosessi verkko-oppimisympäristössä on käyttäjän hallinnassa.					
25. Verkko-oppimisympäristö ei vaadi muistamista ruudulta toiselle siirtäessä.					
26. Verkko-oppimisympäristö auttaa välttämään tarpeetonta oppimista.					

Kehitys	1	2	3	4	5
27. Verkko-oppimisympäristötekniologialle asetetut odotukset täyttyivät hyvin.					
28. Asenteeni on tullut positiivisemmaksi verkko-oppimisympäristöä kohtaan.					
29. Asenteeni on tullut positiivisemmaksi verkko-opiskelua kohtaan.					

Sanallinen arviointi verkko-oppimisympäristöstä

Sopiiko verkko-oppimisympäristö tämän opintojakson opiskeluun? Miksi?

Tukeeko verkko-oppimisympäristö tiedon prosessointia, aktiivista ajattelua ja oppimista? Miten?

Tukeeko verkko-oppimisympäristö yhteistoiminnallisuutta ja ryhmätyön tekemistä? Miten?

Mitkä ovat mielestäsi verkko-oppimisympäristön vahvuudet, heikkoudet, uhat ja mahdollisuudet?

Mitä muuta haluaisit verkko-opiskelustasi kertoa?
