



TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

JUHO KORO
ILMIÖPOHJAISEN VERKKO-OPPIMATERIAALIN RAKENTEINEN
SISÄLLÖNTUOTANTO

Diplomityö

Tarkastaja: Ossi Nykänen
Tarkastaja ja aihe hyväksytty
Tieto- ja sähkötekniikan tiedekunta-
neuvoston kokouksessa 4. touko-
kuuta 2016

TIIVISTELMÄ

JUHO KORO: Ilmiöpohjaisen verkko-oppimateriaalin rakenteinen sisällöntuotanto

Tampereen teknillinen yliopisto

Diplomityö, 60 sivua

Kesäkuu 2016

Tietotekniikan diplomi-insinöörin tutkinto-ohjelma

Pääaine: Hypermedia

Tarkastaja: dosentti Ossi Nykänen

Avainsanat: ilmiöpohjainen oppiminen, oppimisympäristöt, rakenteiset dokumentit, oppimisen analytiikka

Erilaiset pedagogiset lähestymistavat vaikuttavat myös verkko-opetuksen toteuttamistapoihin. Työssä kokeiltiin uudentyyppistä tapaa esittää ja tuottaa rakenteinen oppimateriaali ilmiölähtöisyyden ehdoilla niin, että opiskelijoiden toimintaa pystytään havainnoimaan. Työn tuloksena toteutettiin rakenteinen malli ilmiöpohjaiselle oppimateriaalille sekä prototyyppi web-teknologioihin pohjautuvasta sisällöntuotantotyökalusta Ilmiöpohjaistin, jonka avulla mallin mukaista sisältöä on mahdollista luoda ja hallita. Tapaus-tutkimuksessa mallia pilotoitiin ammatillisen koulutuksen matemaattis-luonnontieteellisillä perusopinnoilla.

Toteutettu malli perustuu rakenteisiin XML-dokumentteihin sekä Tin Can API -teknologiaan, joka mahdollistaa sisällön paketoinnin niin, että opiskelijan toiminta tallentuu Learning Record Store –repositorioon. Mallin mukaisen sisällön käytöstä syntyvää dataa voidaan analysoida dataa visualisoivilla palautenäkymillä.

Työn tulosten perusteella voidaan todeta, että rakenteisen dokumentaation periaatteiden avulla oppimateriaalia voidaan tuottaa, hallita ja julkaista ilmiöpohjaisen opetuksen edellyttämällä tavoilla. Mahdollista jatkokäyttöä ajatellen koottiin sisällöntuottajien sekä teknisten kehittäjien kokemusten ja teoreettisen tarkastelun kautta suosituksia ilmiöpohjaiselle materiaali- ja sisällöntuotantomallille, jonka perusteella olemassa olevaa toteutusta voidaan jalostaa eteenpäin.

ABSTRACT

JUHO KORO: Structured authoring of phenomenon-based learning material
Tampere University of Technology
Master of Science Thesis, 60 pages
June 2016
Master's Degree Programme in Information Technology
Major: Hypermedia
Examiner: Adjunct Professor Ossi Nykänen

Keywords: phenomenon-based learning, structured documents, learning analytics

Different pedagogical methods also affect technical implementations of e-learning. A goal of this thesis was to develop a new way of authoring and providing structured, phenomenon-based e-learning material. We implemented a structured model of the learning material and a prototype of web-based authoring tool Phenomenizer (Ilmiöpohjaistin), where the content based on the model can be created and managed.

The implemented model was piloted with compulsory mathematics, physics and chemistry studies of Finnish vocational students. The learning objects and their collections are based on XML documents, and they were published to Tin Can API format for allowing student activity to be captured in a Learning Record Store repository.

As a result of this thesis, we can state that phenomenon-based learning material can be managed and authored based on principles of structured authoring. With suitable technologies, data from the usage of model could be analyzed with visual analytics methods and tools. For developing the model further, we list recommendations for authoring phenomenon-based based learning material on theoretical analysis and experiences from the case study.

ALKUSANAT

Tämä työ on tehty TTY:n Matematiikan laitokselle osana hanketta, jossa toteutettiin Pohjois-Karjalan Koulutuskuntayhtymälle malli ilmiöpohjaiseen verkko-oppimisaihioon. Ennen diplomityöni aloittamista olin toiminut laitoksen tutkimusryhmässä useita vuosia erilaisissa projekteissa verkko-oppimiseen, web-teknologioihin ja analytiikkaan liittyen, joiden aihealueet lopulta määrittivät diplomityöni suunnan. Projekti alkoi loppuvuodesta 2014 ja kesti kesään 2015. Diplomityötä valmistelin vielä tämän jälkeen. Prosessi on ollut erittäin opettava – monia asioita tekisin toisin, mutta palkitsevaa on ollut huomata varsinkin se, että työn tuloksilla on ollut positiivinen vaikutus, ja toteutetut ratkaisut ovat vielä näiden alkusanojen kirjoitushetkellä aktiivisessa käytössä.

Erityiset kiitokset projektia koordinoineelle Ville Tuomiselle, jonka ansiosta diplomityö oli mahdollista tehdä. Yhteistyökumppanina toimineen Arcusysin edustajia kiitän saumattomasta yhteistyöstä, joka mahdollisti projektin onnistumisen. Matematiikan laitokselta haluan kiittää työn tarkastajaa Ossi Nykästä sekä Pekka Rantaa asiantuntevasta ohjauksesta, kärsivällisyydestä ja arvokkaista kommentteista työn loppuun saattamisen mahdollistamiseksi.

Tampereella, 16.5.2016

Juho Koro

SISÄLLYSLUETTELO

1.	JOHDANTO	1
1.1	Ilmiöpohjainen oppiminen	1
1.2	Tutkimusmenetelmät	2
1.3	Tutkimuksen rakenne.....	3
2.	STANDARDIT JA TEKNOLOGIAT	4
2.1	Opetusteknologian standardoinnista.....	4
2.2	Tiedon rakenteistamisen teknologiat	4
2.2.1	XML-rakenteistaminen	5
2.2.2	Aihelähtöinen sisällöntuotanto ja DITA	6
2.2.3	Asiasanastojen rakenteistaminen	9
2.3	Oppimateriaalstandardeja ja –määrittäjiä	9
2.3.1	SCORM.....	11
2.3.2	Tin Can API.....	11
2.3.3	Muita standardeja ja toteutuksia	15
3.	MATERIAALIMALLIN JA SISÄLLÖNTUOTANTOYMPÄRISTÖN TOTEUTUS	16
3.1	Projektin osapuolet ja aikataulu	16
3.2	Materiaalimalli	17
3.3	Kerättävän datan tunnistaminen	19
3.4	Sisältöobjektit.....	21
3.4.1	Oppimisobjektien rakenne.....	23
3.4.2	Tin Can -paketointi	25
3.4.3	Materiaalin metatiedot ja luokittelu	27
3.5	Kokonaisarkkitehtuuri	29
3.6	Käyttäjien prosessit ilmiölähtöisessä oppimisaihiassa	31
3.7	Sisällön tuottaminen verkko-oppimisympäristöihin.....	31
3.7.1	Moodle	32
3.7.2	SCORM Cloud.....	32
3.7.3	Valamis.....	33
3.7.4	Rakenteisen sisällöntuotannon työkaluja	35
3.7.5	Sisällöntuotantoprosessi materiaalimalliin.....	36
3.8	Oppimateriaalin rakenne ja julkaisu	38
3.9	Sisällöntuotantoympäristö.....	39
3.9.1	Tekninen toteutus.....	40
3.9.2	Sisällöntuottajan käyttöliittymä	42
3.9.3	Materiaalin lisääminen	43
3.9.4	Tarinan rakenteistaminen ja julkaisu	45
3.10	Raportoinnin suhde sisällöntuotantoon.....	47

4.	TAPAUSTUTKIMUKSEN TULOKSET	49
4.1	Mallin soveltuvuus	49
4.1.1	Käyttöönotto	49
4.1.2	Kokemuksia mallista ja sisällöntuotannosta.....	50
4.2	Raporttinäkymät	51
4.2.1	Arviointitaulukko	52
4.2.2	Aikakaavio.....	53
4.2.3	Kriittinen polku.....	54
4.3	Sisällöntuotantoympäristön jatkokehitys	55
4.4	Tulosten arviointia	56
4.5	Suuntaviivoja ilmiöpohjaiselle verkko-oppimateriaalille	56
4.5.1	Sisältömalli	57
4.5.2	Sisällöntuotantoprosessit ja –työkalut.....	58
4.5.3	Tiedon keruu ja palaute	59
5.	YHTEENVETO.....	60
	LÄHTEET.....	61

LYHENTEET JA MERKINNÄT

DITA	XML-pohjainen standardi informaatioasisältöjen hallintaan.
DTD	Document Type Definition
IDE	Integrated Development Environment
JSON	JavaScript Object Notation. Dataformaatti, joka koostuu avain-arvo-pareista
LMS	Learning Management System, oppimisen hallintajärjestelmä. Ohjelmisto, jolla hallitaan oppimiseen liittyvää sisältöä sekä sen käyttöä.
LRS	Learning record store, oppimistapahtumien säilöntään tarkoitettu repositorio
PKKY	Pohjois-Karjalan Koulutuskuntayhtymä
REST	Representational State Transfer. HTTP-protokollaan perustuva arkkitehtuurityyli ohjelmointirajapintojen toteuttamiseen.
RLO	Reusable Learning Object. Uudelleenkäytettävä oppimisobjekti.
SCORM	Shareable Content Object Reference Model, kokoelma standardeja uudelleenkäytettävän oppisisällön hyödyntämiseen eri oppimisympäristöissä.
SKOS	Simple Knowledge Organization System, kontrolloitujen, luokiteltujen asianastojen rakenteen kuvaamiseen tarkoitettu standardi
URI	Uniform Resource Identifier, merkkijono jolla kerrotaan tiedon yksiselitteinen nimi tai sijainti
Valamis	Arcusys Oy:n kehittämä Liferay-pohjainen oppimisalusta
XML	eXtensible Markup Language
XSL	eXtensible Stylesheet Language

1. JOHDANTO

Erilaiset pedagogiset lähestymistavat opetukseen vaikuttavat väistämättä myös verkko-opetuksen toteutustapoihin. Perinteisesti peruskoulussa ja toisen asteen oppilaitoksissa opetus on oppiainekeskeistä, ja niiden toteuttaminen sähköisissä oppimisympäristöissä opetussuunnitelman mukaiseksi on varsin suoraviivaista. Ilmiöpohjaisuus tuo mukanaan sekä sisällöntuotantoon että opetukseen uuden haasteen - monialaiset kokonaisuudet käsittävät useita oppiaineita, ja täten opetuksen toteuttaminen sekä oppimissisältöjen tuotanto monimutkaistuvat.

Tässä diplomityössä esitetään malli verkkopohjaisen ilmiölähtöisen oppimateriaalin tuottamiseen, jonka avulla opettaja voisi hallita sisältöä sekä havaita opiskelijoiden toimintaa oppimisympäristössä myös oppiainekohtaisesti. Osana työtä on ilmiöpohjaisen oppimisen ja opettamisen tarpeisiin soveltuvan materiaalmallin kehittäminen sekä sisällöntuotantoympäristön toteuttaminen ja arviointi.

Diplomityö on toteutettu projektin yhteydessä, jonka tavoitteena oli määrittää toisen asteen ammatillisen koulutuksen tarpeisiin oppiaineita eheyttävä ja ilmiöpohjainen verkko-opetusmalli, jossa opetus sidotaan opiskelijoille tuttuun tarinoiden ympärille. Projektissa pilotoitiin mallia toteuttamalla sen varaan matemaattis-luonnontieteellisten opintojen (matematiikka, fysiikka ja kemia) ilmiöpohjainen oppimisaihio rakenteisen tuotantoprosessin avulla. Tutkimuskysymykset tulevat suoraan projektin aihealueesta: kuinka rakenteistaminen tehdään, miten palaute toteutetaan, ja millaisia kokemuksia tästä toteutuksesta saatiin.

1.1 Ilmiöpohjainen oppiminen

Ilmiöpohjaisella oppimisella tarkoitetaan oppiainerajat ja jopa oppilaitosrajat ylittävää kokonaisvaltaista, todellisen maailman ilmiöitä aidossa kontekstissa tarjoavaa opetusta. Lähtökohta poikkeaa perinteisestä oppiainejakoista koulukulttuurista, jossa usein opettavat asiat on hajotettu suhteellisen pieniksi ja irrallisiksi palasiksi [1]

Ilmiöpohjainen oppiminen on lähtökohdiltaan konstruktivistista. Konstruktivistisen oppimisteorian mukaan oppiminen on aktiivista uusien kokemusten ja tietojen sovittamista aiempiin tietorakennelmiin. [2] Skenaariopohjaisessa oppimisessa opiskelija kohtaa eri tilanteita ja valitsee itse reittinsä. Reitillä saatu palaute kertoo millainen reaktio, merkitys tai vaikutus on opiskelijan tekemillä valinnoilla. Tällaiset signaalit ohjaavat ihmisen toimintaa myös arjessa. Näiden palautteiden perusteella reagoimme, harkitsemme ja

teemme toimintaa ohjaavia päätöksiä. Saatu palaute valinnoista luo kuvan kokonaisuudesta ja antaa toimintamalleja vastaavien asioiden kohtaamiselle. [3]

Verkko-oppimisessa ilmiöiden sitominen todelliseen maailmaan voidaan toteuttaa **tarinapohjaisuuden** avulla. Tarinan avulla rakennetaan opiskelijalle mielikuva kokonaisuudesta ja asiayhteyksistä. Tarinan avulla pohjustetaan oppimistilannetta kohti ilmiölähtöistä tarkastelua ja skenaarioiden hahmottamista. Ilmiölähtöisen oppimisen lähtökohtana on aina tarkasteltava kokonaisuus, josta tarinalla luodaan kuva opiskelijalle. Skenaarioissa tarina toimii vastaavasti auttamalla hahmottamaan asiayhteyksiä ja auttamalla opiskelijaa saamaan tilanteesta kokonaiskuva. Tarinaa voidaan käyttää ongelmanasettelussa ja projektin tehtäväksiantona tai toimintakuvauksena. [3]

1.2 Tutkimusmenetelmät

Diplomityössä on kolme keskeistä tutkimuskysymystä:

1. Kuinka sisällön rakenteistaminen tulee tehdä, jotta se sopii ilmiöpohjaisen oppimateriaalin tuottamiseen (ja käyttöön)?
2. Miten rakenne kytkeytyy palautteisiin?
3. Millaisia kokemuksia mallista saatiin?

Ongelmia lähdettiin lähestymään tapaustutkimuksen menetelmin. Tapaustutkimus (*case study*) on tutkimusstrategia ja lähestymistapa, jossa tutkitaan yksittäistä tapahtumaa tai rajattua kokonaisuutta käyttäen eri menetelmillä hankittuja tietoja.

Tapaustutkimuksen avulla pyritään ymmärtämään ja tulkitsemaan syvällisesti yksittäisiä tapauksia omassa kontekstissaan. Tutkimuskohteeksi valitaan yksittäinen tapaus, tilanne, tapahtuma tai joukko tapauksia, joiden tarkastelussa kiinnostuksen kohteena ovat usein prosessit. Yksittäistapauksia pyritään tutkimaan niiden luonnollisessa ympäristönsään kuvailemalla yksityiskohtaisesti tutkittavaa ilmiötä. Kuvailevien menetelmien tavoitteena on tutkimuskohteen ominaispiirteiden systemaattinen, tarkka ja totuudenmukainen kuvailu. [4]

Sisällöntuotannon reunaehdot määritettiin asiantuntijahaastattelulla, jossa mukana ovat sekä sisällöntuottajan että teknisen kehittäjän roolissa toimivat haastateltavat. Haastatte- luissa kerättiin kokemuksia toteutetusta aiheesta sekä tulevaisuuden visioita mallin kehittämiseen liittyen. Tämän lisäksi lähdeaineistona hyödynnetään projektin aikana tuotettuja palaverimuistioita sekä epämuodollisemmin kerättyjä muistiinpanoja osapuolten välisistä tapaamisista.

Opiskelijoiden toiminta ympäristössä todettiin analysoimalla pintapuolisesti heidän tuottamaansa ja käytöstä syntyntä dataa. Dan avulla voitiin määritellä, kuinka mallin tuottama palaute voidaan muotoilla edelleen visualisoiduiksi raporteiksi opiskelijoiden toiminnasta.

1.3 Tutkimuksen rakenne

Seuraavassa luvussa (luku 2) käydään läpi teknologioita, menetelmiä ja oppimisympäristöihin liittyviä standardeja, jotka ovat relevantteja toteutuksessa.

Kolmas luku kuvaa projektissa toteutetun materiaalimallin ja siihen liittyvän sisällöntuotantoprosessin kehityksen. Materiaali- ja sisällöntuotantomallia taustoitetaan käyttäjien prosessien kartoituksella. Luvussa esitetään ilmiöpohjaisuuden asettamat vaatimukset sekä teknologioille että sisällöntuotannolle, ja kuvataan sisällöntuotantoympäristön tekninen toteutus.

Neljännessä luvussa esitetään tutkimuksen tulokset: mallin soveltuvuus käyttöprosesseihin sekä palautteisiin. Teknologiaavaintojen toimivuus arvioidaan ilmiöpohjaisuuden toteuttajana sekä jatkokehityksen kannalta. Kokemusten ja teorian perusteella määritellään lisäksi suosituksia ilmiöpohjaisen verkko-oppimateriaaliaihion toteutukselle.

Viides luku sisältää yhteenvedon ja loppupäätelmät.

2. STANDARDIT JA TEKNOLOGIAT

Verkko-oppimateriaali voidaan toteuttaa monella eri tavalla. Kuten useissa verkko-oppimisen ratkaisuissa, ilmiöpohjaista oppimisaihiota sekä materiaalimallia toteuttaessa voidaan hyödyntää olemassa olevia ratkaisuja, kuten erilaisia teknologioita sekä oppimateriaalin käyttöön ja rakentamiseen kehitettyjä standardeja.

2.1 Opetusteknologian standardoinnista

Nykyaikaiset verkko-oppimisympäristöalustat mahdollistavat verkko-opetuksen toteuttamisen vaihtoehtoisin opetusmenetelmin. Ympäristöt hyödyntävät erilaisia materiaalmalleja ja standardeja (mm. SCORM), joilla edesautetaan oppimateriaalien toiminnallisuutta ja uudelleenkäyttöä [5]. Standardi voidaan määritellä ”jonkin organisaation esittämäksi suositukseksi siitä, miten jokin asia tulisi tehdä”. [6]

Standardien tarkoitus opetusteknologiassa on toteuttaa yhteentoimivuus eri järjestelmien yhteydessä, jolloin esimerkiksi oppimateriaalin sisältävät paketit ovat uudelleenkäytettäviä ja hyödynnettävissä eri järjestelmissä.

Oppimisalustat ovat verkko-opiskeluun ja verkko-opetukseen tehtyjä järjestelmiä, jotka sisältävät usein paljon tietoa opiskelijoista, oppimateriaaleista sekä opiskeluprosessiin ja opiskelun arviointiin liittyvää tietoa. Oppimisalustat ovatkin hyvin keskeisessä asemassa, kun opetusteknologian standardoinnin tuloksia siirretään käytäntöön. [7]

Opiskelijoiden osaamisen näkyväksi tekemiseen on kiinnitetty viime aikoina erityisesti huomiota. Uudet verkko-oppimisteknologiat ja -standardikokoelmat mahdollistavat oppimisaihion toteuttamisen niin, että opiskelijoiden toimintaa on mahdollista havaita yksityiskohtaisella tasolla erilaisia tarpeita varten. [8] Ilmiölähtöisen opintojakson suorittamista voidaan tarkastella esimerkiksi oppiaineiden tasolla. Erilaisissa oppimisen analytiikan sovelluksissa esitetään erilaisia ratkaisuja tiedonlouhinnasta datan visualisointiin opetuskontekstissa. On mahdollista teettää toiminnasta kertyvästä datasta erilaisia raportteja, jotka tukevat päätöksentekoa.

2.2 Tiedon rakenteistamisen teknologiat

Verkko-oppimISRatkaisujen taustalla on kokoelma standardeja ja rakenteistettuun tietoon liittyviä teknologioita. Rakenteistettu sisällöntuotanto tarkoittaa informaation perusteella tehtävää järjestelyä. Perinteisessä julkaisutoiminnassa sisältöjen ”säännöt” määritellään erilaisin tyylioppain, kun rakenteinen julkaisutoiminta sisältää teknisesti DTD:ssä

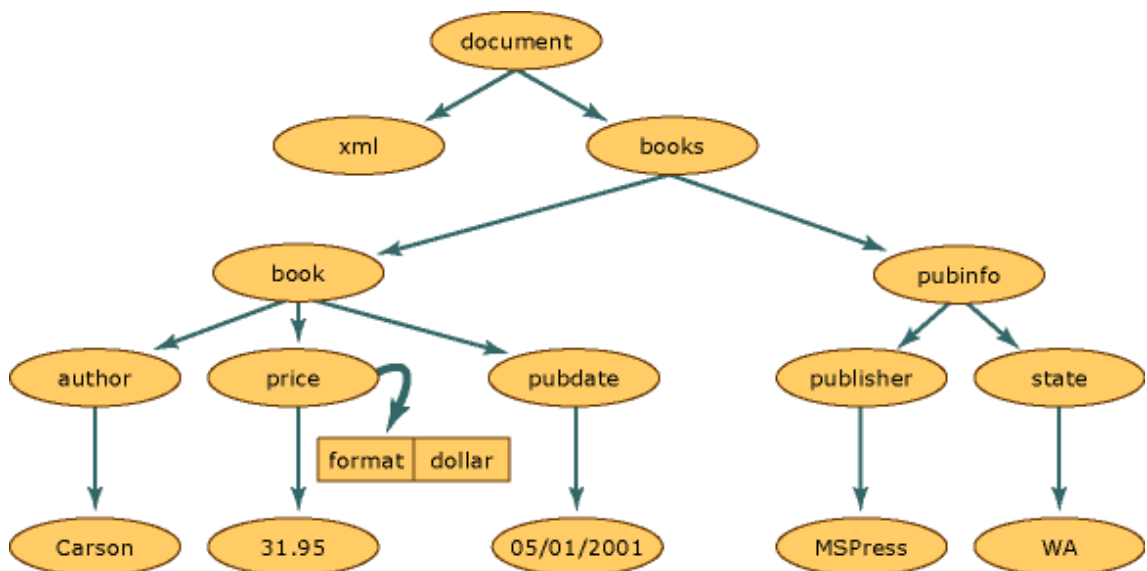
(Document Type Definition) tai skeemassa määritellyt säännöt, joita sisällön täytyy noudattaa. Sisältöä tuottava ohjelmisto validoi tuotetut dokumentit vertaamalla niiden sisältöä dokumenttityypin määrittelyn sääntöihin. [9]

Rakenteistetulla tiedolla on erilaisia käyttökohteita sisällön julkaisusta tietojenkäsittelyyn. Erilaiset organisaatiot kehittävät ja ylläpitävät informaation hallintaan ja web-palveluihin liittyviä standardeja. W3C (World Wide Web Consortium) kehittää WWW-tekniologioihin liittyviä standardeja ja suosituksia. OASIS (Organization for the Advancement of Structured Information Standards) on järjestö, joka kehittää yleisemmin ”informaatioyhteiskunnan avoimia standardeja”, joita ovat muun muassa DocBook ja DITA. [10]

2.2.1 XML-rakenteistaminen

Eräs tärkeimmistä rakenteistetun tiedon standardi on XML (eXtensible Markup Language) [22], joka on menetelmä rakenteisen tiedon muodon esittämiseen dokumenteissa sekä esimerkiksi tietorakenteissa. XML toimii itsessään myös metakielenä muiden merkkaukielten määrittämiselle.

XML-standardiin liittyy läheisesti eräät rinnakkaisteknologiat, kuten dokumenttimuunnoksia suorittava XSL, dokumenttien rakenteen määrittävä XML Schema [11] sekä dokumenteista tietoa hakeva XPath-kyselykieli. [12]



Kuva 1. Esimerkki XML-dokumentin rakenteesta [23]

Rakenteistettu sisällöntuotanto XML-muotoon perustuu elementteihin, jotka ovat sisällön yksiköjä. Elementit voivat sisältää tekstiä tai muita elementtejä, ja niille voidaan määrittellä attribuutteja, jotka ovat elementtiin assosioituja nimi-arvo -pareja. Esimerkiksi kurssia kuvaavassa rakenteisessa dokumentissa attribuutit voivat olla kuvailevia

tietoja. Elementeistä koostuva dokumentin rakenne voidaan esittää hierarkkisena puuna. [13]

Sisällön rakenteistaminen ja erittely yksittäisiksi XML-elementeiksi mahdollistaa julkaisuprosessin joustavuuden. Sama sisältö voidaan tuottaa yhdessä ympäristössä, ja tämän jälkeen julkaista useassa eri formaatissa tai esitysmuodossa. Yksinkertaistettuna esimerkiksi kurssin rakenteinen kuvaus voidaan esittää kurssi-elementin sisältäminä tehtävä- ja oppimateriaali-elementteinä, jotka voivat sisältää edelleen hienojakoisempia elementtejä.

Rakenteistamattomassa dokumentissa elementtien väliset suhteet implikoidaan visuaalisen ilmeen ja typografian avulla, mutta näitä osia on joko vaikeaa tai mahdotonta tunnistaa koneellisesti. Tästä johtuen raakadatan rakenteistaminen vaatii käsityötä.

XML-pohjaisen dokumentin rakenne voidaan kuvata XML-skeeman avulla. Rakenteen täsmällinen kuvaaminen on hyödyllistä, jotta tiedon automaattinen käsittely on sovittu yhteisesti eri järjestelmissä. XML Schema Definition (XSD) on itsessään XML-dokumentti, jossa assosioidun dokumentin sallitut elementit, tietotyypit, attribuutit sekä sisältö esitetään. [11]

Muita XML-dokumentin rakenteen määrittämiseen soveltavia teknologioita on SGML-syntaksiin pohjautuva DTD (Document Type Definition) sekä XML Schemaan verrannolliset skeemakielet RELAX NG ja Schematron. DTD:hen verrattuna uudemmat skeemakielet mallintavat tietoa yksityiskohtaisemmin tarjoamalla esimerkiksi tietotyyppien määrittäykset sekä nimiavaruuksien tuen.

2.2.2 Aihelähtöinen sisällöntuotanto ja DITA

Useammasta dokumentista koostuvan laajemman kokonaisuuden - kuten oppimateriaalin tai teknisen dokumentaation - rakenteistamista voidaan lähestyä aihelähtöisesti ("topic-based authoring").

DITA (Darwin Information Typing Architecture) on XML-pohjainen standardi informaatio- ja sisällöntuotannon hallintaan. Teknologia mahdollistaa tehokkaan dokumenttien uudelleen käytön, ja sitä käytetään usein varsinkin teknisen dokumentaation luomisessa. [14][15]

DITA:n keskeinen periaate on sisällön uudelleen käytössä ja jaettavuudessa. Rakennemallissa informaatio voidaan purkaa modulaarisiksi rakenteiksi, joita voidaan tuottaa, yhdistellä, julkaista ja ylläpitää joustavasti. Modulaariset sisältöyksiköt sijaitsevat keskitetyssä repositoriossa, joista niitä voidaan poimia laajempiin sisältökokonaisuuksiin. Rakenteen avulla voidaan dokumenttien hallinnassa estää redundantin sisällön aiheuttamat ongelmat sekä määrittää sisällöntuotannolle yhtenäiset periaatteet.

Rakenteellisena perusosana DITA:ssa pidetään aihetta (*topic*). Hyvänä käytäntönä pidetään, että yksittäiset aiheet ovat informaation sisällöltään mahdollisimman lyhyitä, mutta kuitenkin tarpeeksi pitkiä ollakseen riippumattomia kokonaisuuksia. Muunnosten kautta dokumenteista ja niitä yhdisteleivistä dokumenttikartoista (*DITA map*) muodostetaan loppukäyttäjälle toimitettuja sisältöjä. [15] Dokumenttikarttaa käytetään muodostamaan resursseista järjestettyjä hierarkioita. Näitä voidaan käyttää eri tarkoituksiin, kuten jaettujen resurssien hallintaan (sanastot), dokumentin osion tai kokonaisen dokumentin määrittämiseen. DITA map käyttää *topicref*-elementtejä määrittääkseen dokumentin rakenteen. Elementit voivat viitata muihin DITA-resursseihin, kuten aiheisiin ja toisiin dokumenttikarttoihin. [15]

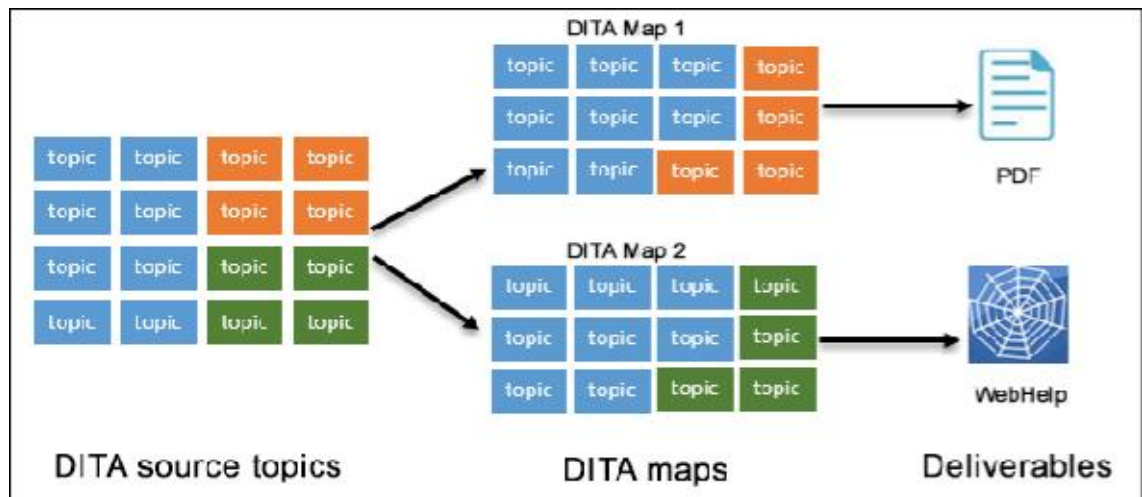
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE map PUBLIC "-//OASIS//DTD DITA Map//EN" "http://docs.oasis-open.org/dita/v1.1/OS/dtd/map.dtd">
<map title="Growing Flowers">
  <topicref href="topics/introduction.dita"/>
  <topicref href="topics/care.dita">
    <topicref href="tasks/pruning.dita"/>
    <topicref href="tasks/gardenPreparation.dita"/>
  </topicref>
  <topicref href="topics/index.dita" collection-type="sequence">
    <topicref href="concepts/springFlowers.dita" collection-type="sequence">
      <topicref href="topics/flowers/iris.dita"/>
      <topicref href="topics/flowers/snowdrop.dita"/>
    </topicref>
    <topicref href="concepts/summerFlowers.dita" collection-type="sequence">
      <topicref href="topics/flowers/gardenia.dita"/>
      <topicref href="topics/flowers/lilac.dita"/>
    </topicref>
    <topicref href="concepts/autumnFlowers.dita" collection-type="sequence">
      <topicref href="topics/flowers/chrysanthemum.dita"/>
      <topicref href="topics/flowers/salvia.dita"/>
    </topicref>
    <topicref href="concepts/winterFlowers.dita" collection-type="sequence">
      <topicref href="topics/flowers/gerbera.dita"/>
    </topicref>
  </topicref>
  <topichead navtitle="Glossary" type="topic">
    <?oxy_comment_end?>
    <topicref navtitle="Genus" href="concepts/glossaryGenus.dita"
      keys="flowers.genus" print="yes"/>
    <topicref navtitle="Pollination" href="concepts/glossaryPollination.dita"
      keys="flowers.pollination" print="yes"/>
    <topicref navtitle="Sepal" href="concepts/glossarySepal.dita"
      keys="flowers.sepal" print="yes"/>
    <topicref navtitle="Rhizome" href="concepts/glossaryRhizome.dita"
      keys="flowers.rhizome" print="yes"/>
    <topicref navtitle="Bulb" href="concepts/glossaryBulb.dita"
      keys="flowers.bulb" print="yes"/>
    <topicref navtitle="Culivar" href="concepts/glossaryCulivar.dita"
      keys="flowers.culivar" print="yes"/>
    <topicref navtitle="Perennial" href="concepts/glossaryPerennial.dita"
      keys="flowers.perennial" print="yes"/>
    <topicref navtitle="Panicle" href="concepts/glossaryPanicle.dita"
      keys="flowers.panicle" print="yes"/>
  </topichead>
</map>
```

```

</topic chead>
<topic cref href="topi cs/copyri ght. di ta"/>
</map>

```

DITA 1.0 ja 1.1 -standardeissa tarjotaan kolme erityyppistä aiheyyppiä: käsite (*concept*), tehtävä (*task*) ja viite (*reference*). Aiheina voidaan käyttää myös muita kuin DITA-tyyppisiä tiedostoja.



Kuva 2. DITA-julkaisun peruseriaate

DITA on hyödynnettävissä erilaisissa kaupallisissa XML-editoreissa (esim. oXygen) [16], jotka tukevat standardia ja tarjoavat sisällöntuotantotyökaluja esimerkiksi dokumenttikarttojen luomiseen ja julkaisuprosessin tukemiseen. Avoimen lähdekoodin työkaluja on myös saatavilla. Esimerkiksi alun perin IBM:n kehittämä DITA Open Toolkit (DITA-OT) on kokoelma komentorivipohjaisia työkaluja, jonka avulla formaatista voidaan julkaista tuotoksia eri muodoissa, kuten HTML- ja PDF-tiedostoina. Useat DITA-formaattia hyödyntävät sisällönhallintajärjestelmät sisällyttävät DITA-OT -paketin sisällöntuotantoon liittyvän työnkulun julkaisuvaiheeseen. [17]

DITA:sta löytyy erillinen spesialisaatio opetuskäyttöön. DITA Learning and Training Content Specialization on DITA-pohjainen sisältömalli opetusmateriaaleille, jotka sisältävät aihelehtöistä sisältöä, uudelleenkäytettäviä oppimisobjekteja ja tarvittavia oppimissisältöjen tyyppisiä. [36]

2.2.3 Asiasanastojen rakenteistaminen

Uudelleenkäytettävyyden kannalta on tarkoituksenmukaista hyödyntää teknologiaa, joka mahdollistaa viitattavuuden sekä ylläpidon erillään sisällöstä. SKOS (Simple Knowledge Organization System) on standardi luokiteltavien, hierarkkisten sanastojen kuvaamiseen. Se on osa semanttisen webin standardiperhettä, ja perustuu RDF (Resource Description Framework) -tietomalliin. Sanasto voidaan kuvata esimerkiksi TURTLE- tai RDF/XML-syntaksin avulla. [18]

SKOS-sanasto perustuu käsitteisiin (*concept*), jotka ovat kuvaavia yksiköitä. Käsitteet yksilöidään uuden tai olemassa olevan yksilöllisen URI:n avulla. Käsitteet kuvataan tarkemmin nimilapuilla (*label*), joissa yksiköt nimetään pääasiallisilla (*prefLabel*) ja mahdollisesti vaihtoehtoisilla (*altLabel*) nimikkeillä. Myös kieliversiointi on mahdollista yksinkertaisten kielitagien avulla. Varsinainen luokittelu tapahtuu käsitteiden välisten semanttisten suhteiden merkitsemisellä. Käsitteelle voidaan määritellä esimerkiksi alakäsite (*skos:narrower*) sekä yläkäsite (*skos:broader*) viittaamalla sanaston muihin käsitteisiin. [18]

Sanastojen hallintaa varten on tarjolla paikallisesti asennettavia editoreita, kuten Protégé ja TopBraid Composer. Web-pohjaisia työkaluja on myös saatavilla, mutta ne ovat ominaisuuksiltaan riisutumpia. Useat työkalut ovat yleisesti ottaen suunnattu enemmän ontologioiden suunnittelutyöhön kuin sanaston sisällöntuotantoon. [19]

2.3 Oppimateriaalistandardeja ja –määrityksiä

Verkko-oppimateriaalien käyttöä ja jakelua varten on kehitetty ylemmän tason standardeja ja niiden kokoelmia, jotka pohjautuvat usein web-standardeihin. Standardit mahdollistavat oppimissisältöjen käyttämisen useissa oppimisalustoissa. Taustalla on usein idea uudelleenkäytettävistä oppimisobjekteista (*learning objects*, LO), jotka määritellään teknisesti yhdenmukaisiksi. [20]

Opetushallituksen koostamissa E-Oppimateriaalin laatukriteereissä on määritelty erilais-
ten materiaalityyppien käsitteitä:

Materiaalityyppi	Määritelmä	Esimerkki
Oppimisaihio	monikäyttöinen, rajatun sisäl- lön tai toiminnan kokonaisuus	harjoitus, simulaatio, havain- nollistus
Teemakokonaisuus	tiettyyn teemaan liittyvä ko- konaisuus, jossa on erilaisia toiminnallisia osioita, esimer- kiksi oppimisaihiota	sää ja ilma kokonaisuus, jos- sa on sekä tekstisisältöä että tehtäviä
Oppimisaihiopankki	teemapohjainen oppimisaihi- oiden kokoelma	matematiikan tehtäväpankki
Kurssin osa tai koko kurssi	sisältöä ja oppimisen ohjausta, lisäksi esimerkiksi oppi- misaihiota	etälukion verkkokurssi, jo- hon liittyy aihioita, ohjausta sekä yhteisöllistä työskente- lyä oppimisalustalla
Oheisaineisto	toista oppimateriaalia, esimer- kiksi kirjaa, täydentävä mate- riaali	verkkoon laitetut oppikirjan lisätehtävät tai syventävä materiaali
Opettajan aineisto	opettajan työtä ohjaava ja tu- keva aineisto	työohje, käsikirja, esitysrun- ko, projektiohje

Taulukko 1. E-Oppimateriaalin käsitteitä [21]

Käsitteistössä määritellyt **teemakokonaisuudet** ovat ilmiöpohjaisessa verkko-
opetuksessa käytännössä monialaisia sisältökokonaisuuksia, jotka määritetään ilmiöläh-
töisyyden edellyttämällä tavalla. Kokonaisuuksien voidaan ajatella koostuvan **oppi-
misaihiosta**, ja oppimisaihiopankit ovat tällaisten sisältöjen kokoelmia. Kurssit toteute-
taan oppimisalustoissa, jotka pystyvät käsittelemään edellä mainittuja sisältöjä. Sisällöt
ovat parhaassa tapauksessa jonkin oppimateriaalistandardin mukaisia.

2.3.1 SCORM

SCORM (Shareable Content Object Reference Model) on kokoelma standardeja ja määrittäjiä web-pohjaisille oppimisympäristöille. Mallin on kehittänyt Yhdysvaltain puolustusministeriön alainen tutkimusryhmä ADL (Advanced Distributed Learning).

Mallin idea on antaa tarkat määrittymiset useissa eri oppimisalustoissa käytettävien sisältöobjektien luomiseksi. Vaikka SCORM:ia pidetään de facto standardina, se koostuu pikemminkin useista aiemmin hyödynnetyistä standardeista ja teknisistä ratkaisuista. Malli antaa ohjeet oppimissisältöjen kehittäjille käyttää näitä standardeja oikein niin, että sisältöä voidaan käyttää monissa eri oppimisalustoissa ja -ympäristöissä. [5]

Useat suositut verkko-oppimisympäristöt, kuten Moodle [24], tukevat nykyään SCORM-pohjaisia sisältöpaketteja. Paketit pohjautuvat teknisiltä määrittämisiltään vahvasti XML-kuvauksiin, joihin sisältö ja navigointijärjestelmä paketoidaan. Oppimisaihiona toimii pakattu kansio tiedostoja, jotka voidaan viedä modulaarisena standardia tukevaan alustaan käytettäväksi.

2.3.2 Tin Can API

Tin Can API (tunnetaan myös nimellä xAPI, Experience API) on Rustici Softwaren kehittämä SCORM-standardin seuraajaksi kaavailtu spesifikaatio, joka pyrkii erottamaan sisällön ja oppimisalustan selkeämmin keräämällä dataa opiskelijoiden toiminnasta. Ajatus on samankaltainen kuin SCORMissa, jossa formaattiin pakatut sisällöt ovat standardimuotoisia ja hyödynnettävissä erilaisilla oppimisalustoilla. [27]

Sisällön määrittelyn sijaan Tin Canin lähtökohdat ovat kuitenkin oppimiskokemuksen mittareiden määrittelyssä. Sisällön tasolla Tin Can -muotoinen sisältö on kääritty pakettiin, jolloin se on valmiina asennettavaksi standardia tukevassa oppimisalustassa.

Toisin kuin SCORM, Tin Can -spesifikaation täyttävä sisältö voi olla muodoltaan mikälaista tahansa, eikä se rajoitu selaimessa esitettäviin sisältöihin. Tin Canissa on ratkaistu myös monia muita SCORM-mallin rajoituksia, kuten ryhmässä oppimisen mahdollistaminen sekä oppimistapahtumien tarkempi seuranta. Samaan kurssiin kuuluvia oppimistapahtumia voidaan tallentaa riippumatta siitä, millä laitteella tai sovelluksella oppijat niitä suorittavat. Seurattavat oppimissisällöt voivat tällöin olla web-sivujen ja muiden dokumenttien ohella esimerkiksi pelejä tai simulaatioita, joiden kanssa opiskelija on vuorovaikutuksessa. [27]

Esimerkiksi opettajilla on monesti tarve nähdä yksityiskohtaisemmin, millä tavoin oppijat ovat suorittaneet tehtäviään. SCORMin mahdollistama seuranta toimii ainoastaan web-selaimen sivulatausten tasolla, kun Tin Can mahdollistaa hienojakoisemman seurannan toteuttamisen.

Tin Can -rajapinnan hyödyntäminen vaatii oppimisalustan yhteyteen tai erillisenä järjestelmänään sijaitsevan Learning Record Storen (LRS), minne käyttäjältä kerätty data tallennetaan. Rekisteri voi olla eri tavoin toteutettu riippuen alustan tarjoajasta. LRS voi sijaita oppimisalustassa tai olla itsenäinen järjestelmänsä. LRS voi lähettää dataa muihin järjestelmiin, kuten oppimisalustoihin, laitteisiin tai muihin LRS:ihin.

Perinteisissä oppimisalustoissa oppijan data säilyy ainoastaan sen organisaation hallinnassa, jonka hallussa oppimisalusta on. Tin Can mahdollistaa lauseiden lähettämisen useampaan LRS:ään samalla kerralla. Yksittäisillä oppijoilla voi olla myös henkilökohtainen LRS (“personal data locker”), johon tallennetaan kaikki kyseisen henkilön oppimistapahtumat. Tämä mahdollistaa oppimistapahtumien jakamisen eri organisaatioiden, kuten esimerkiksi työpaikan ja koulujen, välillä.

Standardia tukevia alustoja ovat muun muassa SCORM Cloud [31] ja Valamis [32]. Oppimisalusta toteuttaa toiminnallisuuden, jolla kerättävä data viedään LRS:ään. Rajapinta käyttää LRS:n kanssa kommunikointiin REST-rajapintaa, ja lähettää sen kautta JSON-dataformaatin mukaisia viestejä. Mikäli viestit noudattavat oikeaa formaattia ja validoituvat LRS:ssä, aktiviteetit tallentuvat tämän jälkeen järjestelmän tietokantaan.

Lähetettävät viestit eli lauseet (statements) ovat perusmuodoltaan tyyppiä aktori (subjekti) - verbi - objekti. Yksinkertaisena esimerkkinä artikkelin lukeminen voidaan kuvata luonnollisella kielellä “*Student experienced Article*”. Datan rikastamiseksi lausetta on mahdollista täydentää haluamallaan metatiedoilla.

Aktori muodostaa tässä viestissä toimijan, mikä sisältää esimerkiksi käyttäjän yksilölliset tunnukset ja siihen liittyvät oheistiedot. Tieto on usein saatavilla oppimisalustasta, jonne ne tallentuvat käyttäjähallinnan avulla kirjautuessa.

Verbi käsittää toiminnan muodon, joka on määritelty yksilöllisenä URI:na (Uniform Resource Identifier). URI kertoo resurssin yksikäsitteisen nimen tai sijainnin. Verbeiksi on muotoutunut parhaiden käytäntöjen mukaan kokoelma soveltuvimpia (mm. ADL) sanoja, mutta niitä voi määritellä myös itse. Sääntönä pidetään, että verbi on aina menneessä muodossa. [30]

Objektina käytetään oppimissisältöä, jolle määritellään myös yksilöllinen URI. Objektina voidaan käsitellä eri tasoisia kokonaisuuksia - esimerkiksi kurseja, tehtäviä ja oheissisältöjä. Yksinkertaisimmillaan nämä ovat HTML-dokumentteja, mutta voivat olla esimerkiksi selaimessa suoritettavien HTML5- tai Flash-muotoisten tehtävien eri osioita.

Koko lauseelle voi määritellä kontekstin, mikä mahdollistaa esimerkiksi aktiviteettien välisten suhteiden määrittämisen ja oppijan määrittelyn työskentelemään osana tiimiä. Konteksti voi kattaa useissa tapauksissa kurssin, jonka osa tehtävä tai muu oppimateriaali on. Tietyille lauseille on myös mielekästä määrittää tulos result-objektissa, johon

voidaan sijoittaa esimerkiksi pisteytettävien tehtävien pistemäärä tai osuus suoritettavasta kokonaisuudesta. Näin laajennetun lauseen luonnollisen kielen muoto voisi olla “*X experienced Y in context Z with result N*”.

Objektin tarkempi konteksti oppimateriaalin osana on myös mahdollista määritellä. Monipuolisempi esimerkki kattaa osana kurssia olevan tehtävän, johon opiskelija vastaa oikein tietyllä pistemäärällä.

Alla esitetään esimerkkinä Tin Can -lause, jossa Erkki Esimerkki on lukenut “Case-kaara” -kurssilla Auton kuntoon laitto -tarinan, ja käyttänyt siihen minuutin ja viisi sekuntia (1M5S, ISO 8601 -formaatti):

```
{
  "id": "88a7a47c-9e28-4c5f-83f3-6b4479476302",
  "actor": {
    "objectType": "Agent",
    "name": "Erkki Esimerkki",
    "mbox": "mailto:erkki.esimerkki@tut.fi"
  },
  "verb": {
    "id": "http://adl.net.gov/expapi/verbs/experienced",
    "display": {
      "und": "experienced"
    }
  },
  "object": {
    "objectType": "Activity",
    "id":
"https://dev.ee.tut.fi/-koro/testixml/form/materiaalimini story/autonkuntoonl ai
tto_3.xml",
    "definition": {
      "name": {
        "fi-FI": "Auton kuntoon laitto"
      },
      "description": {
        "fi-FI": ""
      },
      "type": "http://pkky.fi/xapi/activities/mini story",
    }
  },
  "result": {
    "duration": "PT1M5S"
  },
  "context": {
    "contextActivities": {
      "parent": [
        {
          "id": "https://dev.ee.tut.fi/testi/kurssit/case-kaara",
          "objectType": "Activity"
        }
      ]
    },
    "grouping": [
      {
        "id": "https://dev.ee.tut.fi/testi/kurssit/case-kaara",
        "objectType": "Activity"
      }
    ]
  }
}
```

```

    }
    ],
    "category": [],
    "other": [],
    "id": 26310
  }
},
"timestamp": "2015-06-02T05:53:27Z",
"stored": "2015-06-02T05:53:28Z",
"attachments": []
}

```

On tarjolla muutamia työkaluja, jotka toteuttavat valmiiksi Tin Can -muotoisia oppimateriaaliainioita. Tin Canin tehokkain hyödyntäminen on tällä hetkellä mahdollista ensisijaisesti oppimissisältöjen teknisten kehittäjien tasolla, mutta paketointi on rakennettu sisään joihinkin sovelluksiin. Esimerkiksi Articulate Storyline [33] on työpöytäsovellus vuorovaikutteisten oppimissisältöjen luomiseen, joka tuottaa Tin Can -muotoisia paketteja. Käytännössä lauseet muodostetaan kun käyttäjä tuottaa erilaisia interaktiivisia toteuttavia sisältöjä, jolloin sovellus sitoo tietyn muotoiset lauseet laukaistavaksi sisältöjä käytettäessä.

Sisällöntuottajille on vielä haastavaa määritellä tarkemmin millaisia asioita he haluavat tarkkailla, jolloin sovellukseen ei anneta suoraa mahdollisuutta mittarien tarkempaan valintaan. Tarkkailtavien mittarien valinta riippuu sekä opetushenkilökunnan tarpeista että dataa hyödyntävistä raporteista. Mikäli oppimissisällöt toteutetaan teknisesti alusta alkaen itse, vaatii Tin Can -integraation tekeminen ohjelmakoodin upottamista sovellukseen esimerkiksi JavaScript-pohjaisena selaimella ajettavaksi. On mahdollista kääriä sisältö Tin Can -paketiksi xAPI Wrapperin [56] avulla, mutta monipuolisemmissa oppimisasihioissa lauseille pitää usein määritellä ehdot, jolloin ne laukaistaan. Ehtoja voi toteuttaa myös erillinen aktiviteettien tuottaja (“Activity Provider”, AP), joka voi olla sisältöpaketin ulkopuolinen sovellus. [29]

Koska Tin Can -rajapinta on joustava ja ilmaisuvoimainen, voidaan sitä käyttää myös muiden tapahtumien tallentamiseen ja niiden hyödyntämiseen raporteissa. Tin Canin kautta on teoriassa mahdollista havaita kaikki tarkkailtavaan kokonaisuuteen liittyvät oppimistapahtumat, jotka vaativat interaktiivista järjestelmän kanssa. Suurimpina haasteina on muotoilla eri lähteiden tuottama data yhdenmukaiseksi ja esittää tämä data tarkoituksenmukaisessa muodossa käyttäjäystävällisen raportointijärjestelmän avulla.

2.3.3 Muita standardeja ja toteutuksia

IEEE:n Learning Technology Standards Committee (LTSC) tuottaa erilaisia oppimisteknologioihin liittyviä standardeja. IEEE:n Learning Object Metadata -standardi (LOM) on usein XML-muotoon koodattu malli, joka kuvaa oppimisobjektin tai muun opetuksessa käytetyn digitaalisen resurssin. [34] Standardissa kuvattua määritelmää hyödynnetään esimerkiksi SCORM:in yhteydessä, ja se sisältää yleisluontoisempien metatietokenttien (objektin tyyppi, sisällöntuottaja, omistaja, jakeluehdot) lisäksi esimerkiksi ”pedagogiset attribuutit”, jotka kuvaavat oppimis- ja vuorovaikutustavan. [34]

Learning Tools Interoperability (LTI) on IMS Global Learning Consortiumin kehittämä standardi, jonka tarkoitus on mahdollistaa oppimisalustojen integrointi ulkoisten työkalujen kanssa. Työkalu voi olla esimerkiksi sisällöntuotantoväline, esitystyökalu tai vuorovaikutteinen oppimateriaali. Oppimisalustat tarjoavat usein rajapinnan jonka avulla ulkoisia työkaluja voidaan integroida järjestelmään, mutta jokaisen eri työkalun integrointi voi olla työläs tehtävä. LTI tarjoaa standardoidun mallin rajapinnan kanssa kommunikointiin, jolloin yhdistäminen yhdenmukaistuu. [35]

3. MATERIAALIMALLIN JA SISÄLLÖNTUOTANTOYMPÄRISTÖN TOTEUTUS

Tässä luvussa esitetään materiaalimallin ja sisällöntuotantotyökalujen tuotantoprosessit ja näiden tuottamista varten tehdyt teknologiavalinnat. Luku etenee projektin kulun mukaan – ensin malli suunniteltiin yleisellä tasolla, jonka jälkeen mallia lähdettiin määrittelemään yksityiskohtaisemmin soveltuvien teknologiavalintojen perusteella. Mallin teknisen määrittelyn jälkeen toteutettiin integraatio oppimisalustaan ja määritettiin sisällöntuotantoon liittyvät prosessit ja roolit. Lisäksi kartoitettiin sisällöntuotantotyökaluja ja verkko-oppimisympäristöjä, jotka mahdollistavat vastaavanlaisen sisällön tuotannon. Näiden perusteella päädyttiin toteuttamaan mallille web-pohjainen sisällöntuotantoympäristö Ilmiöpohjaistin, jonka toiminta kuvataan luvun lopussa.

Materiaalimalli perustuu hankkeen alkupuolen aikana määriteltyyn pedagogiseen malliin, jossa ilmiölähtöinen tarkastelu on ennalta ohjattua. Tavoitteena on mallintaa ilmiöpohjaiset opintokokonaisuudet rakenteisilla dokumenteilla, joiden tuottaminen ja ylläpito olisi sisällöntuottajan hallittavissa. Mallin rakenne ja toteutusteknologiat vaikuttavat sisällöntuotantoprosesseihin, opiskelijan näkemään sisältöön sekä opiskelijan toiminnasta kerättävään dataan. Opiskelijan työskentelyä pohjustetaan tarinoiden ja skenaarioiden avulla, joilla luodaan yhteys oppimistavoitteiden ja arjen tilanteiden välille. [3]

3.1 Projektin osapuolet ja aikataulu

Materiaalimalli ja -ympäristö toteutettiin Tampereen teknillisen yliopiston matematiikan laitoksen, Pohjois-Karjalan koulutuskuntayhtymän (PKKY) sekä Arcusysin yhteistyöprojektissa. Hanketta koordinoi PKKY, joka asetti projektin tavoitteet, tuotti oppimateriaalien sisällöt sekä suunnitteli oppimateriaalin visuaaliset elementit. Arcusys kehittää Liferay-alustaan pohjautuvaa Valamis 2.0 –oppimisalustaa, jonka tarkoitus oli toimia pilotissa aineistoa hyödyntävänä oppimisympäristönä. Teknisessä kehitysvaiheessa Arcusys mahdollisti toteutetun, ulkoisella palvelimella sijaitsevan materiaalin kommunikation järjestelmän LRS:n kanssa ja rakensi raportointinäkyvät tuotetun materiaalin perusteella.

TTY:n rooli oli kehittää pedagogiset tietorakenteet käsittävä materiaalimalli PKKY:n hyödynnettäväksi niin, että opiskelijan toiminta syötettyjen sisältöjen yhteydessä tallentuu oppimisalustan LRS-tietokantaan Tin Can –rajapinnan kautta. Tähän vastuualueeseen liittyy diplomityössä esiteltävät tuotokset, eli toteutettu opiskelijadataa keräävä materiaalimalli sekä siihen liittyvä sisällöntuotantoympäristö Ilmiöpohjaistin.

Hanke voitiin jakaa kolmeen osaan, josta ensimmäinen vaihe oli suunnittelu (syksy 2014), toinen vaihe tekninen kehitys (talvi 2015) ja kolmas vaihe pilotointi (kevät 2015). Pilotointivaiheessa oli kaksi vaihetta, jossa ensimmäisessä tekninen kehitys ja ongelmien etsiminen olivat pääosassa. Tässä vaiheessa opiskelijat pystyivät käyttämään ympäristöä ja oppimisaihiota, ja oppimateriaalia tuotettiin sekä julkaistiin aihioon opintojakson kuluessa. Toisen vaiheen painopiste oli toteutetun kokonaisuuden viimeistelyssä yleiskäyttöiseksi ensimmäisen vaiheen kokemusten perusteella.

Hanketta vietiin eteenpäin työpajatyöskentelyllä sekä säännöllisillä, viikoittaisilla videoneuvotteluilla osapuolten kesken. Hanke aloitettiin syksyllä 2014, jolloin ensimmäiset työpajat pidettiin.

Työpaja 20.–22.10.2014

- Kolmen päivän mittaisessa aloitustyöpajassa määriteltiin ilmiöpohjaisen oppimisaihion pedagogiset ja tekniset reunaehdot. Ensimmäisenä päivänä lähdettiin rakentamaan pedagogista mallia ja tunnistamaan siihen liittyvät osa-alueet oppimisobjektien tasolla. Opiskelijan prosessin mittaaminen asetettiin tehtävien arvioinnin sijaan etusijalle.
- Toisena työpajapäivänä tutustuttiin Valamis-oppimisalustan ominaisuuksiin ja sen asettamiin reunaehtoihin toteutettavalle mallille.
- Kolmannessa työpajassa käytiin läpi Tin Can –rajapintaa sekä laajennettiin ensimmäisenä päivänä alustettua mallia miettimällä myös yksittäisen oppimisobjektien rakennetta.

Työpaja 16.1.2015, Tampere

- Päivän kestäneessä työpajassa käsiteltiin sisällön syöttämistä malliin. Tätä varten suunniteltiin lomaketta materiaalin syöttämiseen sekä rakenteen määrittämiseen.

Verkko-opintojaksopilotin aloitustapaaminen 26.2.2015, Joensuu

- Aloitettiin pilotoinnin ensimmäinen vaihe ja ohjeistettiin opiskelijat ympäristön käyttöön.

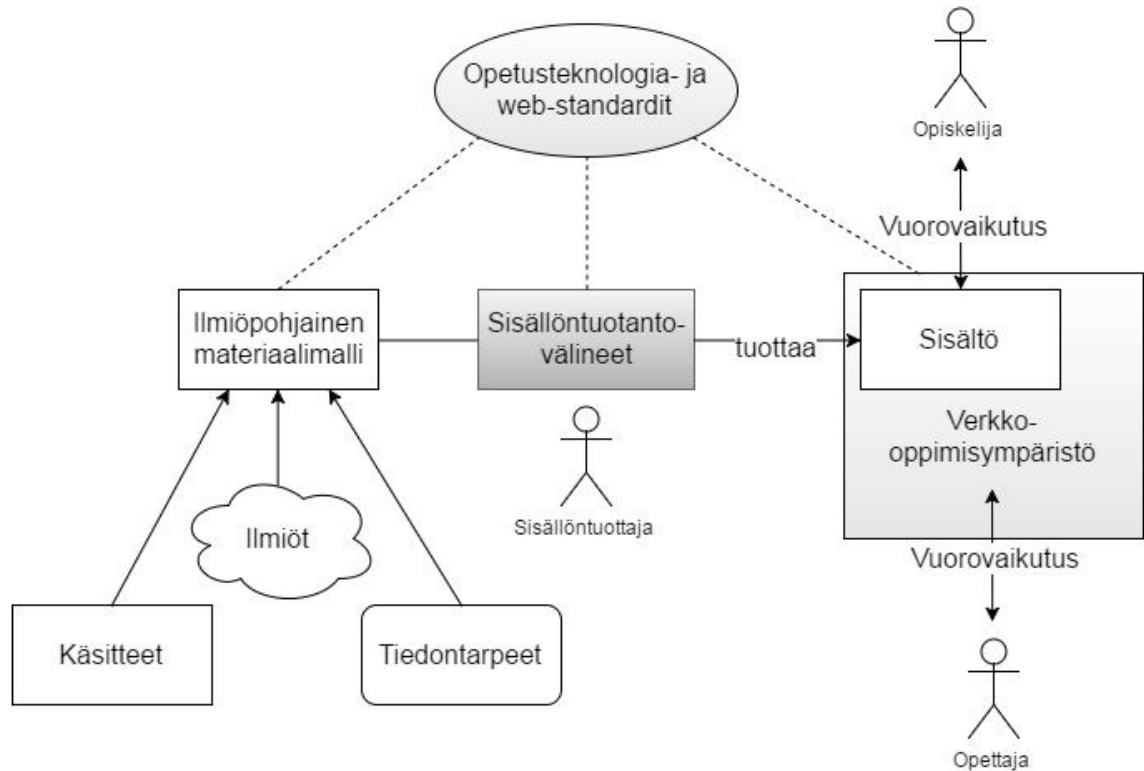
Palautepalaveri 9.4.2015

- Pilotin ensimmäisen iteraation jälkeisessä tapaamisessa käytiin läpi kokemukset ja asetettiin niiden perusteella viimeiset tavoitteet sisällöntuotantoympäristölle ja mallille.

3.2 Materiaalimalli

Malliksi voidaan kutsua yksinkertaistettua kuvaa todellisuudesta. Mallin avulla pyritään yleispätevään esitykseen tarkasteltavasta kohteesta, josta esitetään tarkastelun kannalta oleelliset asiat. Yleispätevät mallit muodostuvat yleensä käsitteistä ja niiden välisistä suhteista. [37]

Ilmiöpohjaisen opetuksen yhteydessä malli sisältää sekä ilmiöiden ja käsitteiden sisällöllisen mallinnuksen, että myös kuvauksen näihin liittyvistä tiedontarpeista palautteen mahdollistamiseksi.



Kuva 3. Korkean tason kuvaus materiaalimallin roolista ilmiöpohjaisen oppimisympäristön toteuttajana.

Materiaalista pyrittiin saamaan mahdollisimman uudelleenkäytettävä ja ylläpidettävä. Materiaalin dokumentit päätettiin tästä johtuen toteuttaa XML-muotoisina sisältökokoisuuksina, mikä antoi mahdollisuuden joustavaan julkaisuprosessiin, sisällön uudelleenkäyttöön sekä sitoutumattomuuteen ainoastaan yhteen sisällöntuotantotyökaluun ja julkaisualustaan.

Malli tulee kuvata niin, että sitä hyödyntävät sisällöntuotantovälineet ja oppimisympäristöt pystyvät käsittelemään sen perusteella kuvattua sisältöä oikealla tavalla. Koostettua mallia ja muita järjestelmiä sitoo erilaiset opetusteknologiaan ja web-tekniikoihin liittyvät standardit, jotka mahdollistavat niiden yhteistoiminnan. Sisältökokoisuuksien hyväksyttävä rakenne voidaan määrittää XML Scheman avulla [11], mikä mahdollistaa niiden toteuttamisen eri sisällöntuotantovälineiden avulla. Sisällön mallintamisen lisäksi malliin sisällytettiin tiedontarpeet paketoimalla materiaalimalli Tin Can –muotoon, mikä mahdollistettiin oppimisen analytiikan ratkaisuiden soveltaminen palautteen antajana opiskelijan toiminnasta.

Materiaalimallin rakenteellista perustaa lähdettiin toteuttamaan lähtökohdasta, jossa ilmiöiden sitomista todelliseen maailmaan lähdettiin toteuttamaan **tarinoiden** kautta. Tarinan avulla rakennetaan opiskelijalle mielikuva kokonaisuudesta ja asiayhteyksistä, ja sen avulla pohjustetaan oppimistilannetta kohti ilmiölähtöistä tarkastelua ja skenaarioiden hahmottamista.

Oppimateriaalissa on mallinnettava sisällön tasolla “käsitemaailma” ja “käytännön maailma”. Tässä työssä “käsitemaailman” mallintaminen lähtee liikkeelle opetussuunnitelmasta, jossa oppimistavoitteiksi on asetettu erilaisten opetussuunnitelmasta lähtöisin olevien käsitteiden ja ilmiöiden hallinta. Oppimistavoitteet kuvataan usein selkeyden vuoksi puumaisessa muodossa, mutta todellisuudessa rakenteesta muodostuu verkkoomainen, sillä käsitteillä on yhteyksiä toisiinsa oppiaineiden välillä sekä niiden sisällä. Ilmiöiden ja käsitteiden välisten suhteiden rakentaminen on merkityksellistä myös ilmiöpohjaisessa oppimisessa, jotta voidaan todeta opiskelijoiden hallitsevan opetussuunnitelman mukaiset osaamistavoitteet.

Tarinoista voidaan irrottaa esimerkiksi ajallisesti pienempiä hetkiä, eli tapahtumia tai “skenaarioita”. Kun käsitellään perinteisiä, lineaarisesti eteneviä tarinoita, niiden mallintaminen eroaa oppiainekohtaisen materiaalin mallintamisesta siinä, että ne ovat usein puurakenteisia ja hajotettavissa pienemmiksi alaosioikseen. Skenaarioissa tarina toimii vastaavasti auttamalla hahmottamaan asiayhteyksiä ja auttamalla opiskelijaa saamaan tilanteesta kokonaiskuva. Tarinaa voidaan käyttää ongelmanasettelussa ja projektin tehtäväksiantona tai toimintakuvauksena. [3]

Mallin taustalla vaikuttaa olemassa oleva opetussuunnitelman mukainen oppiaine- ja kurssirakenne, jonka perusteella keskeiset sisällöt tuotetaan. Tarinoiden yhteydessä havaitaan ilmiöitä ja käsitteitä, jotka liittyvät esimerkiksi tiettyyn matemaattis-luonnontieteelliseen aihealueeseen. Eräänlaisina “piilorakenteina” käytetään taustalla sekä matemaattis-luonnontieteellistä käsitteistöä että varsinaiseen opintosuunnitelmaan liittyviä, oppiaineittain jaoteltuja kursseja. Opiskelijalle materiaalin rakenne näkyy navigoinnin tasolla tarinan muotoon koostettuna, oppiainerajat ylittävänä hierarkkisena rakenteena. Taustalla rakenne ei kuitenkaan ole täysin hierarkkinen vaan verkkoomainen, sillä sisällöt voivat linkittyä toisiinsa tarinan eri tasoilla.

3.3 Kerättävän datan tunnistaminen

Monet verkko-oppimisympäristöt sisältävät erilaisia toimintoja, joiden avulla oppimista pyritään havaitsemaan. Opiskelijoiden toiminnasta ympäristössä kerääntyy lokitietoja, mutta niiden hallinta ja analysointi voi olla haastavaa. Lisäksi on tärkeää määrittää tarkoituksenmukaiset mittarit, joiden perusteella data kerätään. Oppimisen analytiikan (*learning analytics*) menetelmät ja ratkaisut pyrkivät vastaamaan tähän ongelmaan auttamalla huomioimaan opiskelijan toiminnasta signaaleja, jotka ovat merkityksellisiä oppijoille sekä opettajille. Kerätyn datan on korkeakoulutasolla (*academic analytics*)

todettu olevan hyödyllistä myös organisaation hallinnon tasolle, jossa sitä voidaan hyödyntää laajemminkin päätöksenteon tukena. [38]

Oppimisen analytiikalle on monia erilaisia määritelmiä. LAK2011-konferenssissa se määriteltiin teknisellä tasolla olevan opiskelijoista ja oppimiskontekstista kerättävän datan mittausta, keräämistä, analyysia ja raportointia. George Siemens laajentaa määritelmää sekä ”älykkään” (prosessoidun) ja opiskelijoiden generoiman datan käytöksi, että myös malliksi informaation sekä sosiaalisten yhteyksien havainnointiin oppimisen enustamiseksi ja ohjaamiseksi. [40]

Oppimisen analytiikan menetelmät on todettu hyödyllisiksi verkko-opetuksen kehittämisessä sekä sen laadun parantamisessa. [39] Verkko-oppimisessa näkymättömät vuorovaikutustilanteet voidaan datan keräämisen avulla saada näkyviksi, jolloin opettajalle tarjoutuu mahdollisuus tehdä tarpeen tullen interventioita. Opettajalle on taas arviointivaiheessa hyödyllistä saada koosteet esimerkiksi opiskelijoiden aktiivisuudesta. Kokonaisvaltaisempi käsitys oppimisesta ja sen prosesseista voidaan saada pieniä osia yhdistelemällä.

Datan keräämisessä voidaan noudattaa periaatetta, jossa hyvin hienojakoista dataa kerätään suuri määrä ja käytetään suuri määrä resursseja sen analysointiin. Web-analytiikassa tässä lähestymistavassa on ongelmana ”dataähky” (data overload), jolloin tarkoituksenmukainen informaatio piiloutuu merkityksettömien datapisteiden alle. [55] Hyötynä on mahdollisuus tunnistaa asiat, joita ei osattu materiaalin suunnitteluvaiheessa ennakoida. Toisenlaisessa lähestymistavassa keskitytään analyysivaiheen sijaan mittausten suunnitteluun.

Tin Can -rajapinnan kaltaiset tekniset ratkaisut lähtevät liikkeelle juuri mittarien suunnittelusta, mitä kutsutaan ratkaisun dokumentaatiossa ”oppimiskokemusten suunnitteluksi”. Datan keräämisen suunnittelussa tulee Tin Can -dokumentaation mukaan vastata seuraaviin kysymyksiin [54]:

- Millaisista (oppimis)kokemuksista materiaali koostuu?
- Millaista dataa tarvitaan oppimisen analytiikan toteuttamiseksi?
- Millaista dataa tarvitaan palautteiden laatimiseksi mahdollisille sidosryhmille?
- Mistä muista kohteista kuin oppimisaihiosta tarvitaan dataa?
- Mitä dataa tarvitaan tulevien oppimiskokemusten suunnitteluun?
- Millaista dataa kerätään tällä hetkellä ja miksi; tarvitaanko sitä?

Tarkoituksenmukainen kerättävä data voi olla oppimateriaalista riippuen hyvinkin erityyppistä. Yleisellä tasolla tyypilliset mittauskohteet ovat oppimiseen kulutettu aika sekä kirjautumiskertojen, avattujen resurssien, tehtyjen tehtävien ja tuotettujen artefaktien määrä. Haasteena pidetään näiden mittarien tarkoituksenmukaista hyödyntämistä [41]. Yksinkertainen sähköinen oppimateriaali voi koostua ainoastaan päätelaitteella luettavista dokumenteista, joita oppija käy läpi tietyssä järjestyksessä ja käyttää niihin

tietyn määrän aikaa. Jo yksinkertaisimmastakin tapauksesta voidaan löytää useita opiskelijan toimintaa kuvaavia indikaattoreita, vaikka ainoana mitattavana datapisteenä käytetään dokumentin latausta:

- Läpi käydyt asiat
- Oppimiseen kuluva aika (mittaus vähemmän luotettavaa)
- Oppimisen polut (esim. missä järjestyksessä materiaali käytiin läpi?)

Tällainen informaatio on hyödyksi koneellisissa tukitoimissa, kuten suosittelijajärjestelmissä ja oppijan ohjaamisessa. Varsinaisesta oppimisesta on tällaisessa tapauksessa kuitenkin vaikeaa saada luotettavaa tietoa. Opettajalle arviointitilanteessa materiaalin läpikäyntiin liittyvä data toimii pikemminkin selkeämpien mittarien tukena.

Selkeämpänä mittarina voidaan käyttää koneellisesti arvioitavia tehtäviä, joiden analysointi on suoraviivaisempaa. Tehtävät pisteutetaan ja asetetaan skaalalle, jolloin niitä voidaan koostaa eritasoisiksi raporteiksi opiskelijoittain. Ongelmatilanteissa opettaja voi havaita solmukohdat, jotka jäävät useilta opiskelijoilta suorittamatta tai johon vastataan väärin. Myös yksittäisten opiskelijoiden menestystä on tällä tavoin mahdollista tarkastella pisteiden perusteella. Verkko-oppimateriaalissa voidaan hyödyntää pisteutettävien tehtävien lisäksi myös muita interaktiivisia elementtejä kuten simulaatioita (esim. PHET) [42].

Datan tehokas ja monipuolinen analysointi vaatii metadatan rikasta hyödyntämistä materiaalissa. [41] Ilmiöpohjaisessa oppimateriaalissa materiaali sijoitetaan useampaan luokkaan oppiaineiden, kurssin ja käsitteiden mukaan. Lisäksi mitattava objekti on sijoitettu tiettyyn osaan yksittäisen ilmiöpohjaisen kurssikokonaisuuden rakennetta. Tämä data on mahdollista syöttää jo mittausvaiheessa, mikä tekee datan analyysivaiheesta vaivattomamman.

3.4 Sisältöobjektit

IEEE:n määritelmän mukaan yksittäistä tuotettavaa, oppimisessa käytettävää sisältöobjektia voidaan kutsua oppimisobjektiksi (*learning object*) [34]. Dokumenttikartan tapauksessa oppimisobjekti voidaan käsittää materiaalin osaksi tai aiheeksi (*topic*) [14]. Materiaalimallin toteutuksessa käsiteltiin yksittäisiä materiaalin osia näiden määritelmiin perustuvilla atomisilla tasoilla. Oppimisobjektit sijaitsevat repositoriossa, jota kutsutaan tässä yhteydessä dokumenttipankiksi.


Käyttäjälle yksittäinen sisältöobjekti voi näkyä joko artikkelimuotoisena HTML-dokumenttina, vuorovaikutteisena tehtävänä tai muuna sisältönä. Siihen voi sisällyttää myös viittauksia muihin resursseihin, kuten kuviin ja tiedostoihin, jotka käsitetään sisältyväksi yksittäiseen sisältöobjektiin.


Oppimisobjekti jakautuu vielä pienempiin, rakenteisiin osiin, jossa kuvataan sen tietosisältö, metadata sekä viittaukset muihin objekteihin tai mediasisältöihin. Oppimisobjektien sisällöstä suuri osa esitetään XSL-muunnosten kautta materiaalia kuluttavalle käyttäjälle, mutta metadataa voidaan hyödyntää muutakin tarkoitusta varten. Rakenteistetusta tiedosta esitetään loppukäyttäjälle näkymä HTML-dokumentteina. Materiaalin visuaalinen ilme laadittiin PKKY:n tuottamien näkymien perusteella CSS-tyyliä avulla muunnokseen, ja sen avulla liitetään myös materiaalin Tin Can -rajapintaan sitovat skriptit dokumentteihin.

3.4.1 Oppimisobjektien rakenne

TYÖTILAN SUOJAUS JA PUHTAANAPITO

***Argh, nyt nää öljyt valuu pitkin pöytä ja lattiaa! Mitä mä teen? No heitetään läjä noita imeytysmattoja tohoni! Onko tilanne tuttu? Veikkaan, että ainakin jossain mittakaavassa on. Sinänsä äskeinen toimintatapa oli ihan oikea ensiapu, kun vahinko pääsi jo sattumaan. Mutta olisiko siihen tarvinnut ihan koko nippua heittää mattoja? Ne nimittäin muuttuivat samalla vaaralliseksi jätteeksi.**







Kotitalouksiin tarkoitetut puhdistusaineet ovat kokonaisuuksia, joissa on eri ainesosia eri likalaatuja varten. Tämän takia vaikka yleispuhdistusaine sopii monenlaisille tahroille, se ei vastaa tehokkuudeltaan teollisuuden tuotteita. Näistä tuotteista on osattava valita oikea tuote oikeaan kohteeseen. Lisäksi teollisuuden kemikaalit ovat vahvempia (joko happoja tai emäksiä) kuin kotitalouden vastaavat tuotteet.


Teollisuuteen on suunniteltu useita arjen toimintaa tehostavia tuotteita ja palveluita, kuten imeytysmatot ja Vipperi(R) -teollisuuspyyhkeet. Tuotteet on toimitettu työaloihin ja käytetyille on erikseen varattu paikka, josta ne noudetaan. Käytetty teollisuuspyyhkeet pestään ja niistä erotellaan ongelmajätteet jatkokäsittelyä varten. Käytettyä teollisuuspyyhettä tulee käsitellä vaarallisena jätteenä. Sopivissa olosuhteissa voi syntyä itsesyttymisen riski.

Työsalien yleisen puhtauden perusta alkaa keskeisten kulkuväylien vaihtomatoista. Näiden tehtävä on estää niin ulkoa tuleva kosteus kuin muu kiinteä aines, kuten lumi, jää ja hiekka sekä työsalista muihin tiloihin kulkeutuvan öljyn ja muun lian.

Työsalin lattialla oleva lumi/jää/vesi aiheuttaa vakavan liukastumisriskin. Mieti tilanne, jossa siirät painavaa tuotetta ja kenkä luistaa lattialle tulleen jääpölyn päältä.




Tätä osiota on ollut tuottamassa Lindström Oy 




Tapaukset

- Käsien pesu




Tiedostot

- Ekokemin sidosryhmälehti (Ekokem Oy)



Ilmiöt

- Vaaralliset aineet
- Itsesytyminen



Tehtävät

- Imeytysmaton käyttö

EKS
TRA

Vuonna 2014 Peittolankadun yksikössä syntyi pelkästään Vipperi(R) -teollisuuspyyhkeistä ja imeytysmatoista talteenotettua vaarallista jätettä 409,5 kg. Vaarallinen jäte on toimitettu käsiteltäväksi Ekokem Oy:lle Riihimäelle.

Kuva 4. Selaimessa esitetty HTML-muotoinen näkymä artikkelimuotoisesta oppimisobjektista.

Yllä olevassa kuvassa esitetään yksittäinen artikkelimuotoinen sisältö HTML-muodossa kuten se esiintyy selaimessa loppukäyttäjälle. Artikkelimuotoisiin oppiainesisältöihin

toteutettiin mahdollisuus upottaa ulkoista materiaalia, kuten tiedostoja, kuvia, linkkejä ja Youtube-videoita. Viittaukset artikkelin alasisältöihin eritellään sisältötyypeittäin. Alla esitetään vastaava XML-muotoinen sisältöobjekti:

```
<?xml version="1.0" encoding="utf-16"?>
<?xml-stYLESHEET type="text/xsl" href="../../phenomenon.xsl"?>
<content id="7" type="scenario" created="2015-22-01" updated="2015-26-02"
updated_by="Ville Tuominen" author="Ville Tuominen" category="fy"
course="fyp1">
  <title>Työtilan suojaus ja puhtaanapito</title>
  <ingress>"Argh, nyt nää öljyt valuu pitkin pöytiä ja lattiaa! Mitä mä
teen? No heitetääs läjä noita imeytysmattoja tohon!" Onko tilanne tuttu?
Veikkaan, että ainakin jossain mittakaavassa on. Sinänsä äskeinen toimintata-
pa oli ihan oikea ensiapu, kun vahinko pääsi jo sattumaan. Mutta olisiko sii-
hen tarvinnut ihan koko nippua heittää mattoja? Ne nimittäin muuttivat sa-
malla vaaralliseksi jätteeksi.</ingress>
  <text>Kotitalouksiin tarkoitettut puhdistusaineet ovat kokonaisuksia,
joissa on eri ainesosia eri likalaatuja varten. Tämän takia vaikka yleispuh-
distusaine sopii monenlaisille tahroille, se ei vastaa tehokkuudeltaan teol-
lisuuden tuotteita. Näistä tuotteista on osattava valita oikea tuote oikeaan
kohteeseen. Lisäksi teollisuuden kemikaalit ovat vahvempia (joko happoja tai
emäksiä) kuin kotitalouden vastaavat tuotteet.&#xD;
&#xD;
Teollisuuteen on suunniteltu useita arjen toimintaa tehostavia tuotteita ja
palveluita, kuten imeytysmatot ja Vipperi (R) -teollisuuspyyhkeet. Tuotteet on
toimitettu työsaloihin ja käytetyille on erikseen varattu paikka, josta ne
noudetaan. Käytetyt teollisuuspyyhkeet pestään ja niistä erotellaan ongelma-
jätteet jatkokäsittelyä varten. Käytettyä teollisuuspyyhettä tulee käsitellä
vaarallisenä jätteenä. Sopivissa olosuhteissa voi syntyä itsesyttymisen ris-
ki.&#xD;
&#xD;
Työsalien yleisen puhtauden perusta alkaa keskeisten kulkuväylien vaihtoma-
toista. Näiden tehtävä on estää niin ulkoa tuleva kosteus kuin muu kiinteä
aines, kuten lumi, jää ja hiekka sekä työsalista muihin tiloihin kulkeutuvan
öljyn ja muun lian.&#xD;
&#xD;
Työsalin lattialla oleva lumi/jää/vesi aiheuttaa vakavan liukastumisen riskin.
Mieti tilanne, jossa siirrätkä painavaa tuotetta ja kenkä luistaa lattialle
tulleen jääpalan päältä.</text>
  <youtube />
  <images>
    <image_left>pikkukuva_tyosalinsuojauspuhtaanapito.png</image_left>
    <ingressimg>767313927_artikkelikuva_tyosalinsuojaus.png</ingressimg>
  </images>
  <scenarios>
    <scenario
href="https://example.com/material/scenario/ksienpesu_209.xml"
id="209" img="artikkelikuva_ksienpesu.png">
      Käsi en pesu
    </scenario>
  </scenarios>
</content>
<phenomena>
  <phenomenon
href="https://example.com/material/phenomenon/vaarai neet_149.xml"
id="149">
    Vaaralliset ai neet
  </phenomenon>
  <phenomenon href=
```

```

    "https://example.com/material/phenomenon/itsesyttyminen_10.xml"
    id="10">
    Itsesyttyminen
  </phenomenon>
</phenomena>
<concept />
<files>
  <description />
  <link>
    <url>http://example.com/sites/default/files/attachment/ex.pdf</url>
    <title>Ekokeminsidosryhmälähti</title>
    <company>Ekokem Oy</company>
  </link>
</files>
<assignments>
  <assignment
    href="https://example.com/material/assignment/imeytysmatonkaytto
    /story.html"
    id="101">
    Imeytysmaton käyttö
  </assignment>
</assignments>
<related-links>
  <description />
</related-links>
<evaluation />
<extra>Vuonna 2014 Peltolankadun yksi kössä syntyi pelkästään Vipperi (R) -
teollisuuspyyhkeistä ja imeytysmatoista talteenotettua vaarallista jätettä
409,5 kg. Vaarallinen jäte on toimitettu käsiteltäväksi Ekokem Oy:lle Riimi-
mäelle.</extra>
</content>

```

Oppimisobjektin tekstisisältö jakautuu otsikkoon, ingressiin, leipätekstiin sekä lisätietoa tarjoavaan ”extra”-osioon. Tekstissä ei sallita HTML-elementtejä, ja mallissa on vaatimusten perusteella otettu kantaa ulkoasuun esimerkiksi tekemällä rajoituksia dokumenttiin upotettavien kuvien määrään sekä sijoitteluun (ingressing, image_right ja image_left). Tekstimuotoinen sisältö elementeissä sisältää  -entiteettejä, jotka vastaavat rivinvaihtoja.

Muut oppimisobjektin luomisen yhteydessä syötettävät sisällöt ovat artikkelia kuvaavaa metadataa, sen yhteyteen liitettäviä mediaelementtejä sekä linkkejä, joille voidaan antaa kuvaus. Oppimisobjektin sisältämät viittaukset sisältötyypeittäin eriteltyihin alasisältöihin luodaan dynaamisesti tarinarakenteen tuotantovaiheessa (ks. luku 3.8).

3.4.2 Tin Can -paketointi

Oppimisobjektit tulee pakata Tin Can –muotoon, jotta LRS pystyy vastaanottamaan dataa niiden käytöstä. Materiaalissa on Tin Can -raportoinnin kannalta kahdenlaista sisältöä. **Tehtävät** ovat Articulate Storyline -sovelluksen tuottamia, ja sen sisältämät lauseet sekä niiden laukaisijat ovat sovelluksen määrittelemiä. Nämä ovat erillisiä Tin

Can -paketteja, jotka yhdistetään materiaalimalliin. **Artikkelimuotoisia sisältöjä** ovat tarinat ja oppiainesisällöt.

Kommunikointi LRS:n kanssa saadaan aikaan dokumentteihin liitetyn JavaScriptin avulla, joissa on määritelty tarvittavat Tin Can -lauseet sekä logiikka, jonka perusteella ne laukaistaan. Jokaisen oppimateriaaliin sisällytetyn dokumentin yhteydessä käytetään Tin Canin *attempted*-verbiä kuvaamaan sitä, kun oppija avaa sisällön ympäristössä. Lisäksi pyritään mittaamaan selaintasolla kuinka kauan oppija viettää aikaa yksittäisessä sisällössä. Kun oppija poistuu sisällöstä toiseen, laukaistaan *experienced*-verbi.

Alla esitetään esimerkki tavasta, jolla aikaa pyritään mittaamaan ja muuntamaan se Tin Canin hyödyntämään, ISO 8601 –standardin mukaiseen muotoon:

```
$(window).on('beforeunload', function(){
    tincan.sendStatement({
        verb: "experienced",
        context: kurssi.getContext(
            kurssi.CourseAktiviteettiId
        ),
        result: {
            duration: timePeriod(start)
        }
    },
    );
})

// Calculate duration on page (ISO 8601)
function timePeriod(date) {
    var seconds = Math.floor((new Date() - date) / 1000);

    interval = Math.floor(seconds / 3600);
    if (interval > 1) {
        return "PT" + interval + "H" + Math.floor((3600*interval -seconds) /
        60) + "M" + Math.floor(3600*interval*60-seconds) + "S";
    }
    interval = Math.floor(seconds / 60);
    if (interval > 1) {
        return "PT" + interval + "M" + Math.floor(interval*60-seconds) + "S";
    }
    return "PT" + Math.floor(seconds) + "S";
}
```

Oppimissisältöjen loppukäyttäjän näkymään rakennetaan dokumenttimuunnoksen yhteydessä itsearviointilomake, mikä tapahtuu yhdistämällä tarvittava Tin Can -lauseen sisältävä selaimessa suoritettava skripti sekä arviointilomake HTML-muodossa dokumenttiin.

ITSEARVIOINTI

Arvioi miten hyvin hallitsit tämän sisällön		
Tiedän tämän asian	Erinomaisesti	En ollenkaan
Osaan soveltaa tätä työssäni	Erinomaisesti	En ollenkaan
Ymmärrän mitä tällä asialla tarkoitetaan	Erinomaisesti	En ollenkaan
Tämä asia oli mielestäni	Helppoa	Vaikeaa
Yleis"fiilis"		
Vapaa sana:	<input type="text"/>	

Kuva 5. Itsearviointilomake opiskelijalle. Lomake näytetään sisältöjen yhteydessä.

Arviointilomakkeeseen on asetettu asteikko välillä 1-5, jolla opiskelijat arvioivat sisältöön liittyvää tietämystä, sovelluskykyä, ymmärrystä, vaikeustasoa sekä tunnetilaansa. Vapaa sana -kenttä on auki avoimelle palautteelle ja kommenteille. Jokainen opiskelijan syöttämä itsearviointitieto tallennetaan Tin Can -rajapinnan kautta LRS:ään, ja jokaiselle arvioinnin osa-alueelle määriteltiin oma verbinsä. Itsearviointia kuvaamaan ei ollut määritelty yksiselitteisiä ilmauksia ADL:n "de facto" verbilistassa, jonka sisältämiä verbejä suositellaan käytettäväksi ensisijaisina erilaisten LRS:ien raportointitoiminnallisuuksien yhdenmukaistamiseksi. [30] Toteutuksen selkeyttämiseksi täytyi määrittellä sovellukseen sopivat ilmaisut:

- "vibed" - Yleisfiilis
- "evaluated level" - Tämä asia oli mielestäni helppoa/vaikeaa
- "knew" - Tiedän tämän asian
- "understood" - Ymmärrän, mitä asialla tarkoitetaan
- "adapted" - Osaan soveltaa tätä työssäni
- "commented" - Vapaa sana

3.4.3 Materiaalin metatiedot ja luokittelu

Oppiainekohtaista materiaalia voidaan rikastaa metatiedoilla luokittelemalla sitä kategorioihin, jotka ovat hyödyksi materiaalin jatkokäytössä ja siihen liittyvän oppimisdatan analysoinnissa. Metatiedot sisällytetään XML-muotoiseen artikkeliin sekä tiedostolistauksiin.

Materiaalin luokittelu tapahtuu kurssikoodien mukaan, jotka asettuvat matematiikan, fysiikan tai kemian alle. Sisällöntuottajan on luokiteltava oppiainekohtainen materiaali osaksi näitä, ja useamman vaihtoehdon valinta yhdelle materiaalille on myös mahdollista.

Sisältöjen jaottelu tapahtuu sekä materiaalin käyttäjälle että sisällöntuottajalle näkyvimmin ilmiö- tai tarinatasolla, kun oppiaine- ja käsitepohjainen jaottelu on sisältöjen hallintaa ja suodatusta tukevaa metadattaa.

Matemaattis-luonnontieteelliset sisällöt jakautuvat PKKY:n pilottitoteutuksen tapauksessa oppiaineisiin "Matematiikka" sekä "Fysiikka ja kemia". Metatiedon rakenteessa oli kuitenkin mielekkäämpää erottaa viimeksi mainitun kaltaiset monialaiset oppiaineet kahdeksi erilliseksi aineeksi. Myös laajennusmahdollisuus eri oppiaineisiin on olemassa. Ammattioppilaitoksissa opinnot jakautuvat yhteisiin opintoihin sekä alakohtaisiin ammattiopintoihin, mikä voidaan määritellä *type*-attribuutin perusteella.

Oppiaineet jakautuvat edelleen niiden alla oleviin kursseihin. Oppiainejaottelusta muodostetaan erillinen rakenteinen dokumentti `course_categories.xml`, jonka kautta niitä ylläpidetään. Dokumentissa määritellään oppiaineiden alle kurssit, jotka sisältävät sekä nimen että yksilöivän kurssikoodin, jonka perusteella sisällöt viittaavat niihin. Kurssit ovat oppiaineiden lapsielementtejä, joten niiden perusteella voidaan viitata myös oppiainetasolle.

```
<?xml version="1.0" encoding="utf-8"?>
<subjects>
  <subject title="Matematiikka" id="ma" type="yhteiset">
    <course id="map1" title="Peruslaskutoimitukset" compulsory="1" />
    <course id="map2" title="Työelämän matematiikka" compulsory="1" />
    <course id="mav1" title="Talousmatematiikka" compulsory="0" />
    <course id="mav2" title="Matemaattisen tiedon tuottaminen ja analysointi" compulsory="0" />
    <course id="mav3" title="Matemaattinen ongelmanratkaisu" compulsory="0" />
    <course id="mav4" title="Matematiikan perustaidot" compulsory="0" />
    <course id="mav5" title="Matematiikka - AMK-valmentava" compulsory="0" />
  </subject>
  <subject title="Fysiikka" id="fy" type="yhteiset">
    <course id="fyp1" title="Fysiikan perusteet" compulsory="1" />
    <course id="fyv1" title="Kokeellinen fysiikka" compulsory="0" />
    <course id="fyv2" title="Fysiikan syventävä" compulsory="0" />
  </subject>
  <subject title="Kemia" id="ke" type="yhteiset">
    <course id="kep1" title="Kemian perusteet" compulsory="1" />
    <course id="kev1" title="Kokeellinen kemia" compulsory="0" />
    <course id="kev2" title="Kemian syventävä" compulsory="0" />
  </subject>
</subjects>
```

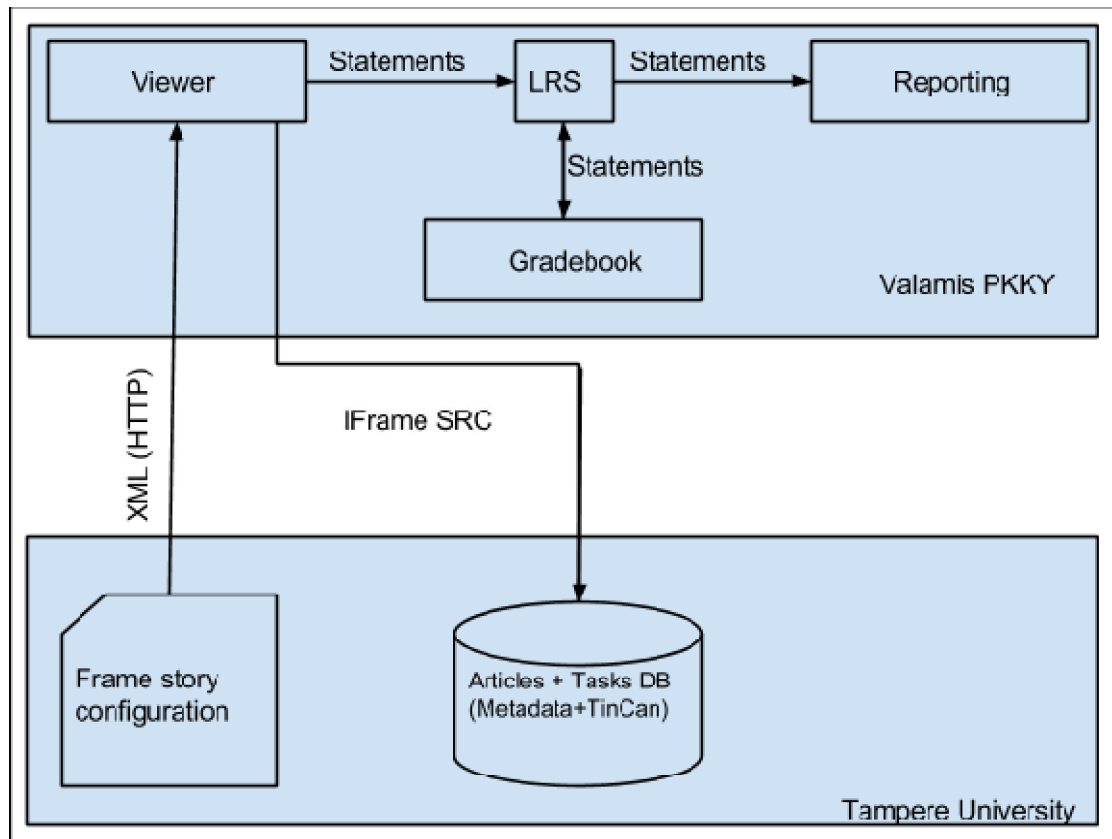
Opiskelija ei näe oppimateriaalia käyttäessään sisältöjen jaottelua oppiaineisiin ja kursseihin. Opettajan näkymissä, kuten raporteissa ja palautetta antaessa kurssi- ja oppiainekohtainen jaottelu voi kuitenkin toimia sisältöjen jäsentäjänä. Sisällöistä ja niiden suorituksista voidaan muodostaa erilaisia näkymiä esimerkiksi opettajan vastuualueen perusteella.

Oppimissisällöt aiheittain luokitteleva sanasto oli tarkoitus saada kontrolloituna asiasanastona, joka on laadittu opetussuunnitelman perusteella. Pilottivaiheessa käyttöön otetussa toteutuksessa mahdollistetaan avainsanojen lisääminen myös sanaston ulkopuolelta, sillä loppuun mietittyä valmista sanastoa ei oltu vielä laadittu. XML-muotoinen sanasto on tulevaisuudessa mahdollista laajentaa rikkaammin kuvatuksi, SKOS-muotoiseksi sanastoksi.

3.5 Kokonaisarkkitehtuuri

Toteutettu oppimisaihio toimii oppimisalustan yhteydessä, jonne opiskelijat voivat kirjautua oppilaitoksen tarjoamilla tunnuksillaan. Materiaalimallia sovelletaan tapaukseen, jossa oppimateriaalikonaisuus on irrotettu alustasta, sillä tavoitteena oli saada malli riittävän riippumattomaksi käytetystä Valamis-oppimisalustasta. Sen sijaan, että materiaali kapseloitaisiin alustan sisälle, se voi sijaita ulkoisella palvelimella ja olla usean eri järjestelmän hyödynnettävissä.

Varsinaisessa materiaalimallissa ei oteta kantaa siihen, kuka materiaalia saa käyttää. Käyttäjähallinnan toteuttaa Valamis, joka myös kehystää materiaalin. Käyttäjähallinnan perusteella saadaan tiedot, jolla yksittäinen opiskelija voidaan yksilöidä Tin Can – lauseessa aktoriksi.



Kuva 6. Yleistason arkkitehtuurikuvaus Valamis-ympäristön ja materiaalin muodostamasta kokonaisuudesta.

Moduulien kuvaus:

- **Viewer** – Valamis-ympäristön sisällönajaja (“content player”) (vrt. SCORM Cloud). Esittää käyttäjille sisältöobjektit ja niiden rakenteen sekä lähettää Tin Can –lauseet. Konfiguroidaan esittämään tarina XML-tiedoston avulla, joka määrittää myös tarinan rakenteen. Konfiguraatio tallennetaan välimuistiin, josta sitä päivitetään säännöllisesti muutosten varalta.
- **LRS** – Valamis-ympäristön Learning Record Store.
- **Reporting** – Erilaiset LRS-datasta tuotetut raportit ja näkymät opettajalle.
- **Gradebook** – LRS-datasta tuotettu opiskelijan henkilökohtainen raportti.
- **Frame story configuration** – Tarinan rakenteen määrittävän XML-tiedoston sijainti.
- **Tietokanta** – Yhden instanssin sisältöobjektitietokanta sekä niiden viittaamat resurssit (kuvat, XSL-muunnos). Konfiguraatio viittaa tietokannan sisältämiin objekteihin.

3.6 Käyttäjien prosessit ilmiölähtöisessä oppimisaihiossa

Sisällöntuotantoprosessit pyrittiin tunnistamaan ja mallintamaan toteutetun mallin perusteella. Ilmiölähtöistä oppimista tukevien järjestelmien muodostaman kokonaisuuteen liittyviin prosesseihin tunnistettiin myös erilaisia rooleja, jotka toteuttavat erityyppisiä sisällöntuotantoprosesseja. Näissä sisällöntuottajaroolit voivat olla osin päällekkäisiä.

Sisällöntuottajat:

- **Raakadatan tuottaja** - Esimerkiksi yritykset, jotka tuottavat pdf- tai HTML-muotoisia dokumentteja. Ei pääsyä oppimateriaalin sisällöntuotantovälineisiin.
- **Oppimateriaalin tuottaja** - Tuottaa uutta, rakenteistaa olemassaolevaa tai syötää sisältöä dokumenttipankkiin.
- **Tarinan rakentaja** - Rakentaa valmiista sisällöistä ilmiölähtöisen oppimateriaalin.
- **Asiasanaston hallitsija** - Ylläpitää kontrolloitua asiasanastoa esim. opetussuunnitelman pohjalta, jonka perusteella materiaalin käsitteet saadaan. Käsitteiden sisällön (kuvaukset) luo oppimateriaalin tuottaja.

Sisällöntuottajien lisäksi materiaalin kanssa vuorovaikuttaa kaksi erillistä roolia:

- **Opiskelija** - Toimii ainoastaan Valamis-ympäristön puolella, pääsy julkaistuun oppimateriaaliin sekä henkilökohtaiseen raporttiin
- **“Opettaja”** - Pääsy Valamis-ympäristössä suoritettavaan oppimateriaaliin, LRS:ään, opiskelijoiden arviointiin sekä raportteihin.

Sisällöntuotanto voi tapahtua organisaation eri tasoilla tai sen ulkopuolella. Tasot voidaan jaotella esimerkiksi oppialoitain tai ainekokonaisuuksittain. Pilottitoteutuksen tapauksessa käsitellään yhtä matemaattis-luonnontieteellisten oppiaineiden kokonaisuutta, jossa roolit toimivat suljetussa ympäristössä. Tässä yhteydessä toimintaympäristö määritellään yhdeksi **sisällöntuotantoinstanssiksi**.

3.7 Sisällön tuottaminen verkko-oppimisympäristöihin

Opettajalla ja sisällöntuottajalla on mahdollisuus soveltaa alustojen ominaisuuksia eri tavoin, mutta toiminnallisuudet asettavat myös rajoja sille, millainen pedagoginen lähestymistapa on mielekkäin. Ilmiöpohjaisen sisällön tuotantotapoja kartoittaessa lähdettiin liikkeelle pedagogisesta lähestymistavasta, ja kartoitettiin sopivimmat ratkaisut tarkastelemalla olemassa olevia oppimisalustoja sekä työkaluja.

Julkaisu-ympäristöllä on merkitystä myös opiskelijoiden kannalta sillä, miten oheistoiminnot tukevat oppimista ja toimintaa oppimisympäristössä. Tärkeässä osassa on raportointiominaisuudet, jotka esittävät opiskelijan toiminnasta kerätyn datan ymmärrettävässä muodossa. Opettajille voidaan tarjota toimintoja, jotka mahdollistavat ajankohtaisista asioista tiedottamisen ja joilla oppimiseen voidaan tehdä erilaisia interventioita. Myös keskustelualueiden kaltaisia vertaisoppimiseen tarkoitettuja työkaluja käytetään.

3.7.1 Moodle

Moodle on diplomityön kirjoitushetkellä suosituin verkkopohjainen oppimisympäristö [26], mihin vaikuttaa sen avoin lähdekoodi ja modulaarinen rakenne, jonka avulla ympäristö voidaan rakentaa organisaation tarpeisiin sopivaksi. Moodle perustuu kursseihin, jotka ovat eri käyttäjien hallinnassa ja joita voidaan luokitella eri kategorioihin.

Moodlen kurssialustaa voidaan käyttää monipuolisesti hyödyntämällä alustan tarjoamaa teknologiaa erilaisten sisältöjen ja tehtävätyyppien luomiseksi sekä opiskelijoiden välisen vuorovaikutuksen tukemiseksi. Sisällön rakentaminen ja hallinta Moodlella tapahtuu lukuisten eri sisältötyyppien avulla. Sisällöntuotannon perustoiminnallisuuksina järjestelmään voidaan esimerkiksi lähettää tiedostoja sekä erilaisia mediasisältöjä, luoda HTML-sivuja sekä perustaa keskustelualueita. Sisällön luominen tapahtuu ympäristöön sisällytettyjen web-pohjaisten lomakkeiden ja WYSIWYG-editorien avulla. [24]

Useissa tapauksissa Moodle-toteutusta hyödynnetään ainoastaan materiaalipankkina ja kurssin aikataulun jäsentelyyn tarkoitettuna alustana, jossa opiskelijoiden vuorovaikutus toistensa ja kurssihenkilökunnan kanssa on vähäistä.

Moodlella on pitkään ollut tuki uudelleenkäytettäville SCORM-muotoisille sisältöpaketeille, mutta Tin Can -tukea ei tarjota suoraan sisäänrakennettuna. Raportointi opiskelijoiden aktiivisuudesta tapahtuu sivulatausten tasolla, mihin oppimisympäristöissä on tarjolla yksinkertaiset työkalut, kuten lokinäkömää tapahtumista. Moodleen on tarjolla kehittäjäyhteisön tuottamia lisäosia, joilla raportointimahdollisuuksia pystyy laajentamaan. [25]

3.7.2 SCORM Cloud

SCORM Cloud [31] ei ole varsinainen verkko-oppimisympäristö, vaan alusta jonka kautta SCORM-standardin mukaisia sisältöjä voidaan tarjota käyttäjille. Se tukee SCORM:n lisäksi Tin Can -muotoisia sisältöpaketteja ja sisältää sisäänrakennetun LRS:n. Ympäristö on nopea ottaa käyttöön, ja sen hinnoittelu tapahtuu käyttäjämäärien sekä käytetyn kapasiteetin mukaan. Ympäristön hyötynä on sen tarjoama rajapinta, jonka kautta paketteja voidaan upottaa varsinaisiin oppimisympäristöihin, kuten Moodleen tai personoidumpaan sisällönhallintajärjestelmään perustuvaan verkko-oppimisympäristöön. Oppimistapahtumia voidaan havainnoida erilaisten raporttinäkymien kautta, jotka antavat yleiskuvan oppimisympäristön käytöstä ja niiden suorituksesta käyttäjittäin. Kaikki oppimistapahtumat ovat nähtävissä LRS-näkymän kautta, joka kerää JSON-muotoisen raakadatan yhteen näkymään ajallisesti järjestettynä.

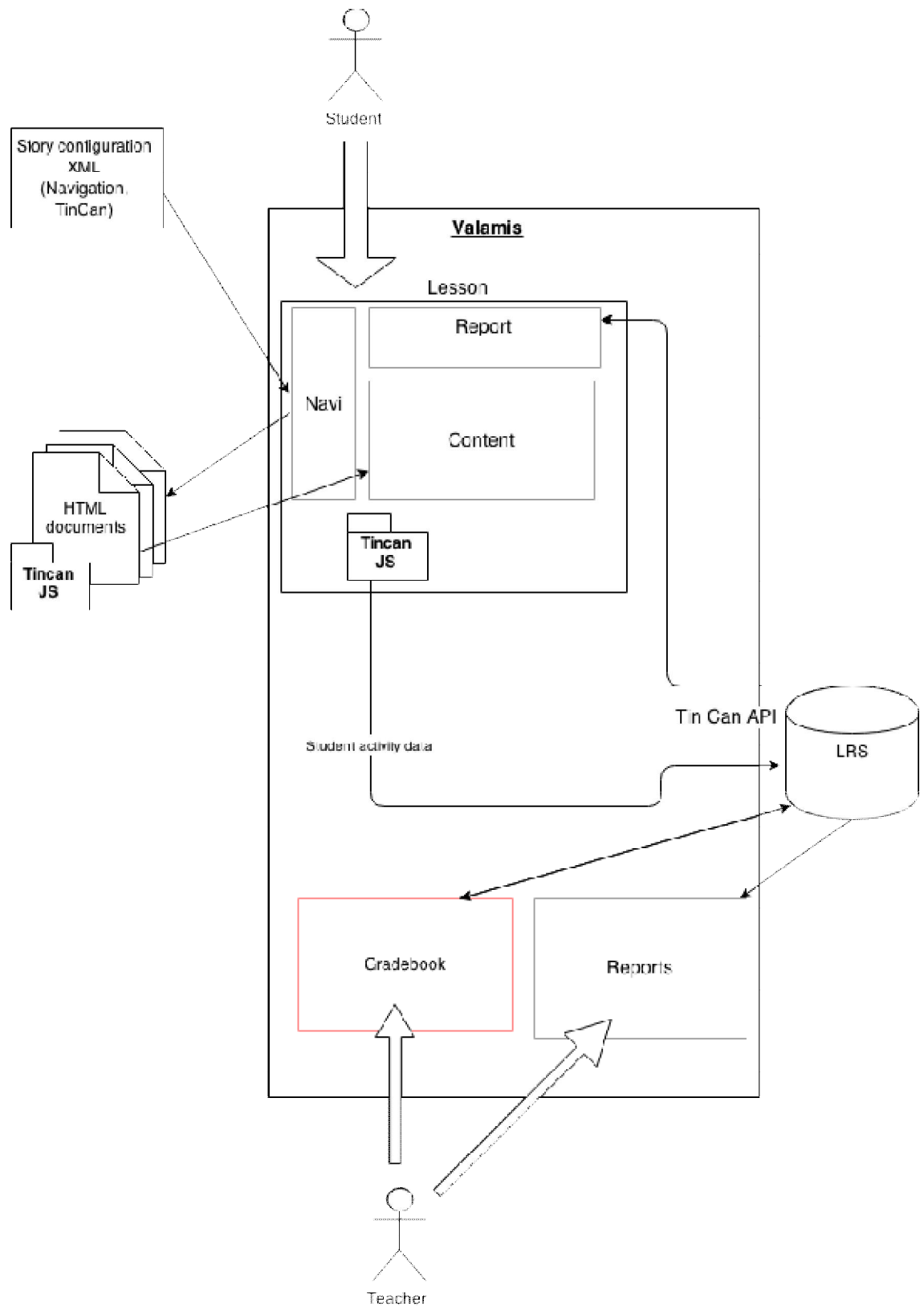
SCORM Cloud on toimiva ratkaisu etenkin Tin Can -muotoisten materiaalipakettien kehittämiseen, sillä niiden lataaminen ja testaus alustalla on yksinkertaista. Projektin aikana alusta soveltui hyvin materiaalipakettien testaamiseen.

Vaikka SCORM Cloudin raportointiominaisuudet eivät välttämättä tarjoa tarpeeksi tarkkaa tai oppimissisältöjen kannalta relevanttia informaatiota, rajapinnan kautta saadaan tuotua raportointidata Tin Can -muodossa, joka voidaan käsitellä haluamallaan järjestelmällä.

3.7.3 Valamis

Valamis on Liferay-julkaisujärjestelmän päälle rakentuva oppimisalusta, joka mahdollistaa SCORM- ja Tin Can -pakettien lataamisen ja sisältää myös LRS:n. Toteutus on Liferay-plugin (”portlet”), jonka asentaminen on helppoa julkaisujärjestelmän hallintanäkymän kautta. Muut Liferay-portletit on käytettävissä myös Valamiksen ohella, jolloin oppimisympäristöä voi laajentaa organisaation tarpeiden mukaan monipuolisia työkaluja hyödyntäen. [32]

PKKY:n ilmiöpohjaisen oppimisasihion kehitysprojektissa päädyttiin toimimaan yhteistyössä Arcusysin kanssa. Valamis-oppimisalustaan toteutettu sisältöpakettien hallinta ei toteutuksen alkuvaiheessa palvellut suoraan prototyypin tarpeita, mutta joustavan yhteistyön myötä Arcusys päätyi kehittämään alustaa käyttämään myös ulkoisia Tin Can –resursseja, jollaisena myös materiaalimalli toteutettiin. Lisäksi projektin aikana ympäristöön toteutettiin materiaalimallin mahdollistaman LRS-datan perusteella esitettäviä näkymiä opettajille.



Kuva 7. Opettajan ja opiskelijan prosessit Valamis-oppimisympäristössä. Vasemmalla materiaali, keskellä Valamis-ympäristön näkymät ja oikealla LRS-tietokanta.

Valamis-ympäristössä opiskelijalla on pääsy oppimateriaaliin, jossa toimiminen tuottaa LRS:ään Tin Can -rajapinnan kautta dataa, joka on osaltaan nähtävissä henkilökohtaisten raporttinäköymien kautta. Opettajalla on pääsy laajempiin raportointinäköymiin, jotka luodaan useampien opiskelijoiden toiminnasta koostetusta datasta. Näköymien avulla tuetaan esimerkiksi arviointiprosessia.

3.7.4 Rakenteisen sisällöntuotannon työkaluja

Sisällöntuotantotyökalut ovat perinteisessä sähköisen oppimateriaalin tuotannossa usein erilaisia työpöytäsovelluksia, kuten tekstinkäsittelyohjelmia. Kokonaisvaltaiset oppimismalustat sisältävät kuitenkin myös niihin upotettuja sisällöntuotantovälineitä, jotka asettavat rajoja sille, kuinka materiaalia lähdetään muotoilemaan, asettelemaan ja järjestelemään. [44]

Materiaalit tulee sitoa olemassa olevaan oppimateriaalin rakenteeseen, jolloin sisällöntuottajan on hallittava tarina kokonaisuutena. Sisällöntuottajilla voi olla erilaisia lähtökohtia, mistä johtuen prosessin kulku voi olla erilainen [45], esimerkiksi:

- Kurssin rakentaminen tarinan pohjalta
- Kurssin rakentaminen opetustavoitteiden pohjalta
- Sisältöjen päivittäminen
- Ilmiöiden ja käsitteiden liittäminen materiaalin rakenteisiin

Sisällöntuotantoprosessia lähdettiin laatimaan näiden lähtökohtien perusteella, jolloin mahdollisimman moni tavoite tulisi huomioiduksi. Tähän liittyen tehtiin kartoitus työkaluista, joilla XML-muotoinen sisältö ja rakenne olisi mahdollista tuottaa.

Työkalujen valinnassa tulee ottaa huomioon käyttäjien tausta sekä materiaalin asettamat vaatimukset. Sisällöntuotantotyökalujen kartoituksessa tutkittiin vaihtoehtoja, jotka vaihtelivat XML-editoreista web-pohjaisiin sisällöntuotantotyökaluihin. Editorissa on käyttäjän näkökulmasta hyvä olla WYSIWYG-editori, tai siinä on muulla tavalla havainnollistettu selkeästi sisällöntuottajan tekemät muutokset. Materiaaliin tulee voida liittää erilaisia mediaelementtejä.

Erilaisiksi vaihtoehtoiksi valikoitiin seuraavat työkalut:

- Eclipse - IDE, johon voi lisätä XML-laajennoksia. Ilmaisia laajennusosia löytyy monia, mutta ovat usein vaikeakäyttöisiä ja rajoittuneita.
- Oxygen – Monipuolinen XML-editori. Editorin haltuunotto vie aikaa, mutta sisältää kattavan dokumentaation. Sisältää Eclipse-pluginin, ja on maksullinen. [16]
- SDL Xopus - Selkeä web-pohjainen WYSIWYG-editori XML-muotoisen, yksittäisen dokumentin muokkaamiseen. Mahdollisuus liittää sisällönhallintajärjestelmään. [43]
- Projektissa toteutettava lomakepohjainen web-työkalu oppimateriaalin julkaisuun

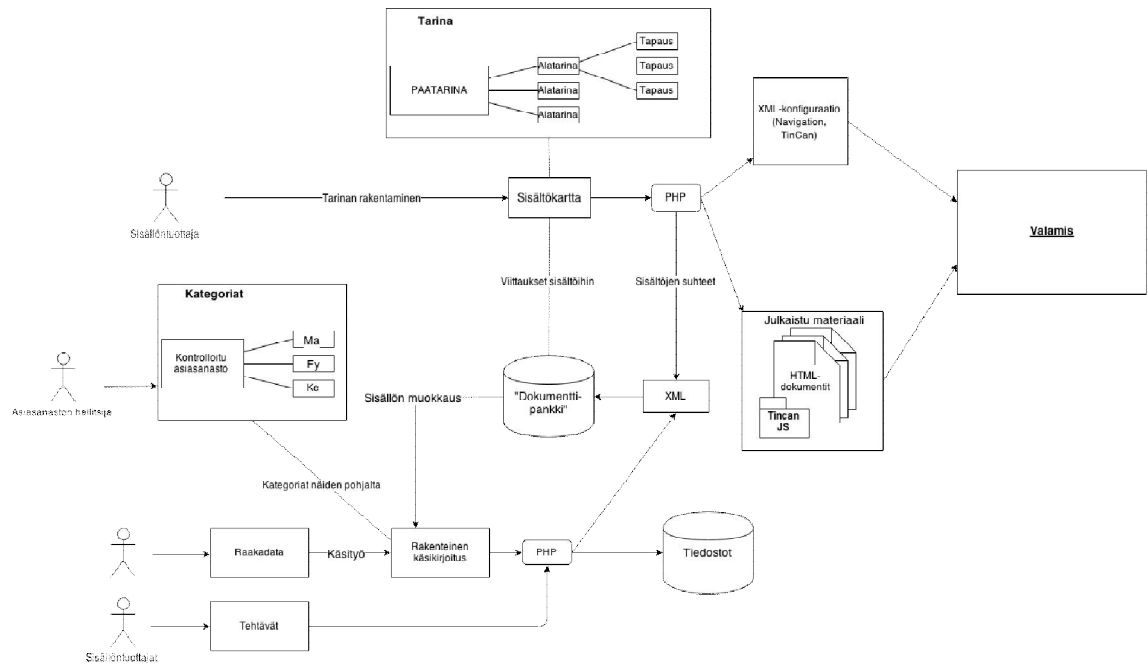
Ilmaisista editoreista ei löytynyt täysin kattavaa ratkaisua sisällön hallintaan, joka olisi voitu ottaa käyttöön sisällöntuottajien osaamis pohjalla järkevässä ajassa. Tästä johtuen materiaalin tuotantoon päädyttiin rakentamaan räätälöity sisällöntuotantoympäristö.

Räätälöidyn julkaisu- ja sisällöntuotantojärjestelmän rakentaminen asettaa rajallisten resurssien takia monia rajoitteita sisällön hallinnan suhteen, mutta yksinkertaistaa sisällön tuottamista ja julkaisua, mikäli organisaation roolit ja sisällöntuotantoprosessit ovat riittävän kartoitettuja. Ei ole mahdollista tuottaa kaikkia ominaisuuksia, mutta mahdollistetaan silti materiaalin uudelleenkäyttö. Kehittyneempi editori on mahdollista ottaa käyttöön myöhemmin, mikäli sisällöntuottajien roolit ja työvaiheet selkiytyvät.

Projektin alkuvaiheessa suunnitelmana oli käyttää ilmiöpohjaisten opintojakson sisällöntuotannon pohjana SKOS-muotoista sanastoa. Sanastojen hallintaan pyrittiin myös löytämään sopiva työkalu, sillä sanaston tuottaminen havaittiin määrittelyvaiheessa selkeästi erilliseksi prosessiksi, joka tehdään opetussuunnitelmatasolla. Ilmiöpohjaisen kurssin sisällöntuottajalla olisi täten mahdollisuus hyödyntää käsitteistöä oppimateriaalin ja tarinan rakentamisvaiheessa. Uusien työkalujen käyttöönotto näytti kuitenkin vaativan liikaa resursseja, jolloin yksinkertaistettu sanasto päädyttiin laatimaan dynaamisesti sisällön syöttämisen ohessa ”avainsanoja” käyttäen.

3.7.5 Sisällöntuotantoprosessi materiaalimalliin

Tässä prosessikuvauksessa sisällöntuotanto suoritetaan erillään Valamis-alustasta, jolloin oppimisympäristö toimii ainoastaan ulkoisessa lähteessä sijaitsevaa materiaalin alustana, LRS:nä ja mahdollistaa käyttäjähallinnan sekä opiskelijoiden arvioinnin.



Kuva 8. Käyttäjien sisällöntuotantoprosessit yhdessä instanssissa.

Oppimateriaalin raakadata voi olla esimerkiksi PDF- ja HTML-muotoisia dokumentteja. Raakadatan rakenteistamiseen tässä toteutuksessa määriteltyyn muotoon tarvitaan välttämättä käsityötä, jolloin tuotetaan sisällön rakenteinen käsikirjoitus. Myös täysin alkuperäistä sisältöä voidaan luoda yllä kuvatulla periaatteella.

Oppimisobjektien sisällöntuotannossa pyrittiin erottamaan erilaisia sisältötyyppejä niiden käyttö- ja tuotantotavan perusteella:

- Tarinat (päätarina, minitarina)
- Skenaariot, ilmiöt, käsitteet – Uudelleenkäytettävät sisällöt
- Tehtävät - Tuotetaan Articulate Storylinellä
- Tiedostot - Ei-rakenteistetut, esimerkiksi yritysten tuottamat tekniset dokumentaatiot, videot sekä simulaatiot.

Näistä sisältötyypeiksi määriteltiin **päätarina** ja **minitarina**, jotka määrittävät tarinan perusrakenteen. Osa-alueet alustaviin minitarinoihin jaoteltu päätarina toimii ”juurena” tarinalle. Pilottitoteutuksessa ”Case Kaara – Ilmiönä auto” päätarina kehystettiin autoon liittyviin ilmiöihin. Auton hankintaan, käyttöön ja huoltoon liittyviä osa-alueita käytettiin minitarinoina.

Minitarinat jakautuvat edelleen niistä kehystettyihin tapauksiin eli **skenaarioihin**, jotka toimivat eräänlaisena rajapintana uudelleenkäytettävien sisältöjen ja tarinoiden välillä, mutta jotka määrittävät kuitenkin uudelleenkäytettäviksi useissa tarinoissa. Tapahtumat ovat kokonaisuuksia, jotka kattavat useita ilmiöitä.

Skenaarioiden alle sijoittuvat **ilmiöt** ja **käsitteet** ovat uudelleenkäytettävän materiaalin sisältötyyppejä. Ilmiöiden sisällä on usein käsitteitä, jotka sisältävät oppiainekohtaisen materiaalin.

Tehtäväpaketit laaditaan Articulate Storyline -sovelluksella, joka kykenee tuottamaan TinCan -standardin mukaisia sisältökokonaisuuksia. Käyttäjä näkee tehtävät asennetuista selainlaajennuksista riippuen HTML5- tai Flash-muodossa ajettavina sovelluksina. Zip-muotoon pakattuja tehtäviä voidaan viedä dokumenttipankkiin, jolloin niistä luodaan materiaalimallin mukainen oppimisobjekti hyödyntämällä Storylinen generoimaa metadataa.

Päätarina	”Case Kaara – Ilmiönä Auto”
Minitarina	Vaaratilanne
Skenaario	Jarrutus ja väistö
Ilmiö	Jarrut
Käsite	Kitka
Tehtävä	Nopeuden muutokset

Taulukko 2. Esimerkkejä ilmiöpohjaisen oppimisaihion sisältöobjekteista.

Tapauksessa, jossa materiaalia ei ole mielekästä rakenteistaa (esim. tietyt tekniset kuvaukset, yritysten tuottamat dokumentit, videot), materiaali ainoastaan viedään järjestelmään jonkin oppimisobjektin yhteyteen. Tiedostot eli esimerkiksi rakenteistamattomat PDF-dokumentit liitetään osaksi siihen liittyvää oppimisobjektia.

Toteutetut oppimisobjektit rikastetaan metadatalalla jatkokäsittelyn ja tehokkaan sisällönhallinnan mahdollistamiseksi. Materiaalien käsitteisiin liittyvä sisältö luodaan kontrolloidun asiasanaston pohjalta, jonka sisällön määrittelee asiasanaston hallitsija. Tällöin materiaalissa käytettäväksi käsitteeksi ei kelpaa sanaston ulkopuoliset termit. Käsitteet tallennetaan omaan XML-kuvaukseensa. Objekteihin voidaan liittää myös muita, vapaamuotoisia kuvauksia, jotka toimivat vastaavasti käsitteistön metadatanä.

3.8 Oppimateriaalin rakenne ja julkaisu

Ilmiölähtöisten kokonaisuuksien hallitsemiseksi on tarpeellista erottaa selkeästi kurssin rakenteen ja sisältöjen muokkaaminen.

Sisällöntuottajalla on mahdollisuus muokata ilmiölähtöisen oppimateriaalin rakennetta dokumenttikarttojen (*document map*) kaltaisten rakenteiden avulla. Ilmiöpohjainen oppimisaihio rakentuu päätarinan (frame story) ympärille, joka koostuu useammasta mini-

tarinasta, jotka sisältävät edelleen skenaarioita. Tämä puumuotoinen rakenne on mielekästä kuvailla DITA-tyyppisellä arkkitehtuurilla, jonka *ditamap*-tyyppinen rakenne sitoo aiheiden kokonaisuudet (*collections*) yhteen. [15] Sisällöntuotannon yhteydessä hyödynnettäviä sisältöjen kokoelmia on kahden tyyppistä - ensimmäinen määrittelee kokoelmat sisältötyypeittäin, toinen käsittää sisältöjen väliset suhteet.

Rakenteen muokkaamisessa alemmalla tasolla oleviin sisältöihin on mahdollista lisätä ylempänä olevia tehtäviä, ilmiöitä ja käsitteitä. Dokumenttikartan päivittämisen yhteydessä materiaalin sisäiset viittaukset päivittyvät dokumenttipankkiin. Näkymässä esitetään rakenne havainnollisessa muodossa rakenteen selkeyttämiseksi sisällöntuottajalle.

Päivityksessä on ajettava skripti, joka käsittelee XML-muotoista dataa ja tuottaa siitä XSL-muunnoksien avulla selaimessa esitettäviä HTML-dokumentteja. Lisäksi dokumentin sisällön sekä sen metatiedot sisältävä XML tallennetaan dokumenttipankkiin, jossa se on uudelleenkäytettävissä. HTML-muotoiset dokumentit sijaitsevat HTTP-palvelimella, ja opiskelijoilla on pääsy niihin Valamis-ympäristön kautta (ks. kuva 5, Valamis-prosessikaavio).

Rakennetta muokatessa luodaan Valamis-ympäristössä käytettävä XML-muotoinen konfiguraatiotiedosto, jossa määritellään Tin Can -lauseet sekä alustetaan materiaalin navigointijärjestelmä. Konfiguraatiotiedostoa käytetään tämän lisäksi myös apuna raporttinäkymien määrittämisessä, sillä siitä on saatavilla helposti koko materiaalin rakenne.

3.9 Sisällöntuotantoympäristö

Tämän diplomityön yhteydessä toteutettiin sisällöntuottajien tarpeisiin ympäristö **Ilmiöpohjaistin**, jossa materiaalin rakenteistamista tuetaan selainpohjaisella käyttöliittymällä ilman kolmannen osapuolen editoreita, mutta joka mahdollistaa silti materiaalin jatkojalostuksen. Ympäristön käyttöliittymä sisältää lomakepohjaiset näkymät oppimateriaalin syöttämiseen sekä oman puunäkymänsä oppimateriaalin rakenteen määrittämiseen.

Ilmiöpohjaistin-ympäristö on materiaalimallin tapaan irrallinen Valamis-oppimisympäristöstä, mutta se tuottaa sisältöobjektien lisäksi tarinakohtaiset konfiguraatiotiedostot Valamis-ympäristöä varten. Konfiguraatiota hyödynnetään navigointipuun rakentamisessa sekä sisältöön liittyvän metadatan koostamisessa raportteihin.

Yksittäinen Ilmiöpohjaistin määritellään tässä yhden suljetun **sisällöntuotantoinstanssin** toteuttajaksi. Instanssi voi sisältää useita eri oppiaineita, kursseja, tarinoita ja sisältöjä, jotka voivat linkittyä keskenään. Taustalla on ajatus oppiainekokonaisuuksien tai ammattiaineiden luokitteluun kategorioihin, joihin liittyy omat kurssikokonaisuudet ja joita hallitsevat eri toimijat [45].

3.9.1 Tekninen toteutus

Käytännössä Ilmiöpohjaistin on palvelimelle asennettava web-sovellus, jonka käyttöön vaaditaan nykyaikainen selain. Teknisesti ympäristö on toteutettu PHP-skriptikielellä, jonka avulla rakennetaan näkymät ja toiminnot materiaalin lisäämiseen, muokkaamiseen ja julkaisuun tuotantoympäristössä. Toteutuksessa olennainen osa oli datan käsittely, jossa hyödynnettiin PHP:n DOM-kirjastoja sekä XML-elementtien manipulointia helpottavaa SimpleXML-kirjastoa. [53]

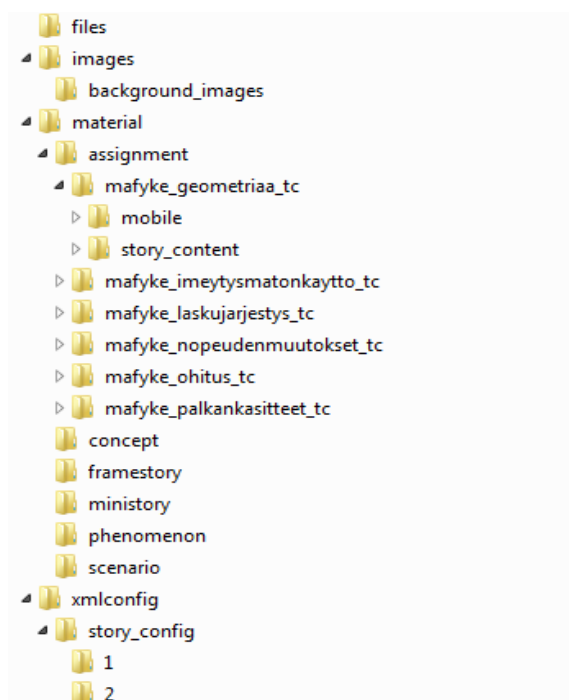
Ilmiöpohjaistimen jakelupaketti sisältää sisällöntuotannossa tarvittavat PHP- ja JavaScript-tiedostot. Lisäksi mukana ovat laukaisimet määrittävät Tin Can –konfiguraatiot sekä asettelun ja ulkoasun toteuttavat XSL- ja CSS-tyylitiedostot, jotka ovat vaihdettavissa ja muokattavissa. Alla olevassa taulukossa esitetään ympäristön käyttämät, palvelintasolla suoritettavat komponentit (näkymät ja vastaavat kontrollerit):

Näkymä	Kuvaus	Kontrolleri
form.php	Lomakenäkymä, joka tuottaa oppiainekohteisista sisällöistä (<i>scenario, phenomenon, concept</i>) XML-muotoisen oppimisobjektin. Selainnäkyssä käytetään jQuery DatePicker ja Chosen -kirjastoja.	post.php
form_story.php	Lomakenäkymä tarinamuotoisten oppimisobjektien (<i>framestory, ministory</i>) tuottamiseen.	post_story.php
tehtava.php	Lomake, joka tuottaa zip-pakatuista Articulate Storyline –tehtävistä oppimisobjektin.	zipper.php
kategoriat.php	Aine- ja kurssikategorioiden kuvausten lisäämiseen tarkoitettu lomake.	post_category.php post_subject.php
rakenne.php	Yksittäisen tarinan rakenteen muokkaamiseen tarkoitettu näkymä. Hyödyntää dhtmlxTree –komponenttia puunäkymän muodostamiseen materiaalipankista ja tarinasta.	savestructure.php

edit.php	Luettelo- ja hakunäkymä oppimisobjekteille. Käyttää DataTables –JavaScript-kirjastoa hakutoimintojen ja luettelon järjestelyn toteuttamiseen.	list.php delete.php
editform.php, editform_assign ment.php	Lomakkeet oppimisobjektien muokkaamiseen. Sama lomake kuin uuden sisällön luomisessa, mutta esittää lomakkeessa teksti- ja mediasisällön sekä oppimisobjektin suhteet toisiin oppimisobjekteihin.	post_edit.php

Taulukko 3. Ilmiöpohjaistimen PHP-komponentit, jotka toteuttavat lomakenäkymät ja datankäsittelijät.

Ympäristö ei vaadi tietokantaa, vaan se tallettaa syötetyn informaation palvelimen hakemistorakenteisiin, jossa materiaali on oppimisalustan hyödynnettävissä ja viitattavissa myös sen ulkopuolelta.



Kuva 9. Yksittäisen instanssin hakemistorakenne hyödynnettävän materiaalin osalta.

Files-hakemistoon tallennetaan materiaalin käyttämät liitetiedostot. **Images**-hakemisto sisältää materiaalissa käytetyt kuvat, joita käytetään materiaalimallissa artikkeliin upotettuna kuvana tai tarinan taustakuvana.

Material-hakemiston sisältö on jaoteltu materiaalimallissa määritellyin sisältötyypein. Se sisältää yksittäiset oppimisobjektit XML-tiedostoina, sekä tehtävien (**assignment**) tapauksessa myös Storyline-tehtävämateriaalin joka koostuu useammasta eri tiedostosta.

Xmlconfig-hakemisto on tarkoitettu konfiguraatioille, jotka määrittävät instanssin dokumenttipankin sisältötyypeittäin, eri tarinoiden konfiguraatiot sekä sanastot (käsitteet ja oppiaine- sekä kurssikategoriat). Tarinakohtaiset konfiguraatiot jaotellaan sisältötunnisteen perusteella alihakemistoihin.

Alla olevassa taulukossa kuvataan Ilmiöpohjaistimen käyttämät, rakenteiset konfiguraatiot:

Tiedosto	Käyttötarkoitus
tree.xml	Ryhmittelee kaikki instanssin sisällöt dokumenttikarttaan sisältötyypeittäin.
[course-id]/tree2.xml	dhtmlxTree-komponentin sarjallistama muoto tarinan rakenteesta, jota käytetään käyttöliittymässä.
[course-id]/structure.xml	Tarinan konfiguraatio Valamis-oppimisalustaan. Sisältää tarinan sisällöt ja niiden väliset suhteet vieruslistaesityksenä. Lisäksi käsittää yksittäisten sisältöjen Tin Can –konfiguraatiot ja muun metadatan (type, category, manifestActivities).
setup.xml	Konfiguraatio editoria varten. Pitää kirjaa sisällön ID-tunnisteista.
course_categories.xml	Määriteltävä kurssikategoriasanasto, joka luettelee oppiaineet, niiden sisältämät kurssit ja näiden tyypit (ammattiopinto/yhteinen opinto, pakollinen/valinnainen).
framestories.xml	Luettelo luoduista päätarinoista.

Taulukko 4. Ilmiöpohjaistimen konfiguraatioon käytetyt XML-kuvaukset.

3.9.2 Sisällöntuottajan käyttöliittymä

Työkalun käyttöliittymä sisältää lomakepohjaiset näkymät oppimateriaalin syöttämiseen ja kurssikategorioiden hallintaan sekä oman puunäkymänsä tarinan rakenteen määrittämiseen.

Käyttöliittymässä hyödynnetään JavaScript-pohjaista dhtmlx-kirjastoa, joka tarjoaa erilaisia hyödyllisiä käyttöliittymäkomponentteja HTML-dokumenttiin upotettavaksi. Li-

säksi käytetään jQuery-kirjastoa sekä tähän pohjautuvaa Chosen-kirjastoa, jonka avulla perinteiset HTML-pohjaiset monivalintoihin liittyvät lomake-elementit muutetaan käyttäjäystävällisemmiksi [49].

3.9.3 Materiaalin lisääminen

Oppimateriaaliobjektin lisääminen tapahtuu lomakkeella, joka rakennetaan materiaalin tyyppiin (tarinasisältö, artikkelisisältö tai tehtäväsisältö) mukaan. Lomake-elementteihin syötetyn tiedon perusteella asetetaan data mallin elementteihin ja attribuutteihin. Työkalu tuottaa tarvittavat Tin Can -paketoitujen oppimisobjektien ja rakenteisten dokumenttien sisältötuottajan syöttämien tietojen perusteella.

Kuva 10. Osa sisällönlouontilomakkeesta.

Ilmiöpohjaistimessa on erillinen lomake tehtävien lataamiselle, millä tuotetaan tehtävästä erillinen oppimisobjekti hyödyntämällä tehtäväpaketin sisältämää metadataa. Articulate Storyline tuottaa valmiiksi tehtävien rakenteen ja Tin Can –lauseet.

Dokumenttipankkiin tallennettua materiaalia voidaan jällempäin muokata. Sisältöobjektit ovat näkyvillä ympäristön kautta näkymässä, jossa niitä voi hakea ja luokitella nimen lisäksi esimerkiksi tyyppin ja kategorian mukaan. Muokkauksen yhteydessä haetaan sisältöobjekti, ja täytetään lomakkeen kentät dokumenttiin tallennetuilla tiedoilla.

Kurssikategorioiden lisääminen ja hallinta tapahtuu myös lomakkeen avulla, joka on erillään sisältöjen hallintanäkymästä.

Kurssien hallinta

▶ Matematiikka

▼ Fysiikka

Fysiikan perusteet	fyp1	<input checked="" type="checkbox"/> Pakollinen
Kokeellinen fysiikka	fyv1	<input type="checkbox"/> Pakollinen
Fysiikan syventävä	fyv2	<input type="checkbox"/> Pakollinen

+ Lisää kurssi

▶ Kemia

▶ Yhteiskuntataidot

Lisää uusi oppiaine

Oppiaineen nimi

Oppiainetunniste (esim. ma, fy, ke...)

Oppiaineen tyyppi

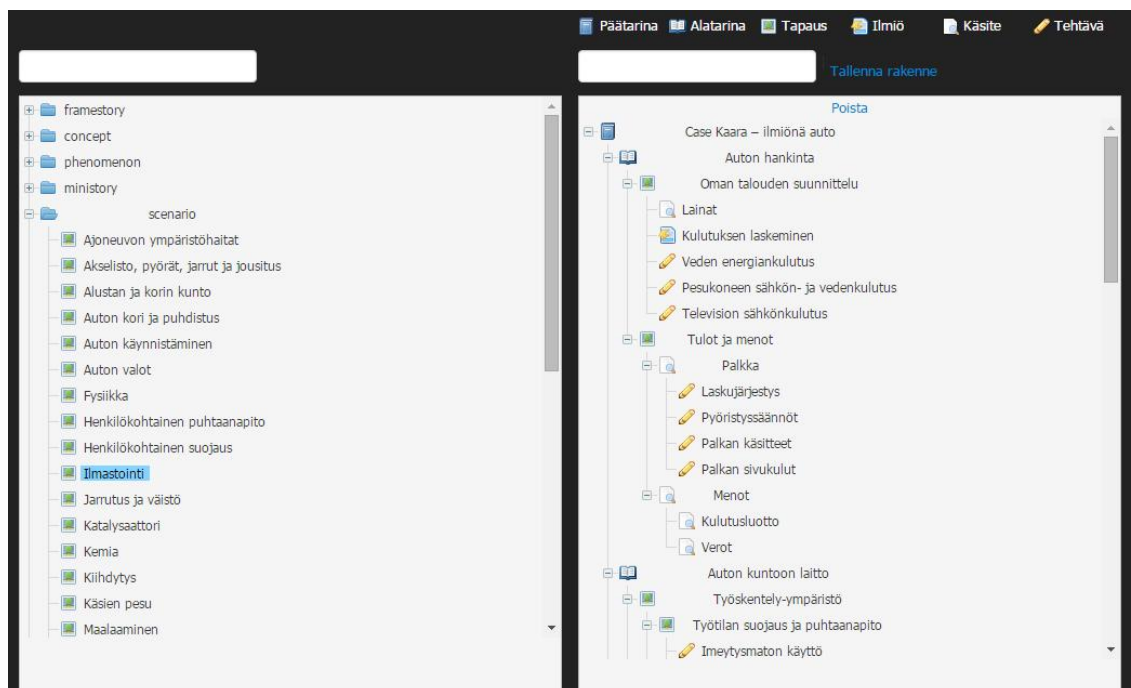
Yhteiset opinnot
 Ammattiopinnot

Kuva 11. Oppiaine- ja kurssikategorioiden lisäämiseen tarkoitettu lomakenäkymä.

Lomakkeen avulla voidaan luoda sisällöntuotantoinstanssiin eri oppiaineita ja sisällyttää niihin kurseja. Kurssi- ja oppiainekokonaisuus talletetaan course_categories.xml

-konfiguraatioon, jonka perusteella kategoriaelementit (ks. kuva 8) rakennetaan sisältö-lomakkeisiin.

3.9.4 Tarinan rakenteistaminen ja julkaisu



Kuva 12. Ilmiöpohjaistimen tarinan rakenteistamisnäkyminen.

Tarinan rakennetta luodessa määritellään sisällöille niiden väliset suhteet. Näkyvässä hyödynnetään dhtmlx-kirjaston Tree-komponenttia, joka visualisoi hierarkkisen datan esimerkiksi hakemistorakenteista tutun puurakenteen avulla. [50] Puunäkymä jaotellaan kahteen osaan – vasemmalla sijaitsevaan dokumenttipankkiin ja oikealla toteutettavan tarinan rakenteeseen. Vasemmanpuoleisesta sisältötyypeittäin luokitellusta materiaali-pankista voidaan viedä drag-and-drop -tyylisesti oppimisobjekteja oikeanpuoleiseen tarinaan, jossa se kootaan hierarkkiseksi kokonaisuudeksi.

Näkymien data kuvataan XML-muotoisissa dokumenteissa. Komponentti lukee tree.xml -tiedostosta dokumenttipankin, jossa oppimisobjektit on jaoteltu sisältötyypeittäin. DhtmlxTree sarjallistaa tarinakohtaisesti tree2.xml -tiedoston, mikä tallentaa tarinan rakenteen käyttöliittymän tasolla. Tästä rakenteesta voidaan määrittää lopullinen hierarkia ja luoda yksittäisiin sisältöobjekteihin viittaukset ylempiin sisältöihin.

Rakennetta tallennettaessa tehdään palvelimelle AJAX-kutsu ja ajetaan PHP-skripti, joka päivittää suhteet sekä oppimisobjektien tasolla, että konfiguraatitiedostossa (structure.xml), jota Valamis käyttää. Skriptin avulla käydään läpi dhtmlxTree-komponentin tuottama puumuotoinen rakenne, joka muunnetaan vieruslistaesitykseksi subscenarios-elementtiin sekä oppimisobjektin sisältötyypeittäin määriteltyihin aliartikkeleihin.

XML-muotoisesta rakenteesta voidaan hakea helposti haluttu data SimpleXML-kirjaston sekä XPath-kyselykielen avulla. Monimutkaisempi PHP:n DOM-rajapinta mahdollistaa uusien XML-dokumenttien luomisen haetun datan perusteella.

Alla esitetään esimerkki konfiguraatiodokumentin rakenteesta tarkastellen yksittäistä elementtiä (käsite-tyyppinen artikkeli "Metalli"). Oppimisobjekteja kuvaavat elementit rakentuvat juurielementin `elements` alle.

```
<?xml version="1.0" encoding="utf-8"?>
<elements xsi:noNamespaceSchemaLocation="elements.xsd"
xml:ns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <!-- ... -->
  <element>
    <subscenarios>
      <child>173</child>
      <child>176</child>
      <child>178</child>
    </subscenarios>
    <id>197</id>
    <name>Metallit</name>
    <type>concept</type>
    <categories>
      <category id="http://pkky.fi/xapi/categories/physics">
        Fysiikka</category>
      <category id="http://pkky.fi/xapi/categories/chemistry">
        Kemia</category>
      <category id="http://pkky.fi/xapi/categories/fyp1">
        Fysiikan perusteet</category>
      <category id="http://pkky.fi/xapi/categories/kep1">
        Kemian perusteet</category>
    </categories>
    <link>https://example.com/material/concept/metallit_197.xml</link>
    <manifestactivities>
      <activity
        id="https://example.com/material/concept/metallit_197.xml"
        type="http://adlnet.gov/expapi/activities/lesson">
        <name>Metallit</name>
        <description lang="fi-FI" />
        <launch lang="fi-FI">metallit_197.xml</launch>
      </activity>
    </manifestactivities>
  </element>
  <!-- ... -->
</elements>
```

- **subscenarios**-elementti koostuu viittauksista rakenteessa alempana oleviin sisältöihin yksilöllisten id-tunnisteiden avulla.
- **categories**-elementti sisältää sisältöön liitetyt kategoriat, joiden URI määritellään kategorian tunnisteiden perusteella. Kurssi- ja ainekohtaiset kategoriat ovat tässä toteutuksessa samalla tasolla, ja URI esitetään.
- **link**-elementti esittää viitattun dokumentin todellisen sijainnin.

- **manifestActivities**-elementti sisältää Tin Can -määritykset, jotka sijaitsivat yksittäisen Tin Can -paketin tapauksessa tincan.xml -tiedostossa [28]. Ilmiöpohjaistimen työkaluilla tuotettuihin sisältöihin Tin Can -elementit tuotetaan koodattujen määritysten perusteella sisältötyypeittäin. Tehtävien tapauksessa elementin sisällön Articulate Storyline on tuottanut valmiiksi.

3.10 Raportoinnin suhde sisällöntuotantoon

Toteutetussa mallissa hyödynnetään kappaleessa 2.3 esiteltyyn Tin Can -rajapintastandardiin pakattua aihiota mahdollistamaan oppimistapahtumien havainnointi. Mallin mukainen sisällön pakointi mahdollistaa standardia tukevan ympäristön rekisteröidä opiskelijan oppimistapahtumat mittareilla, jotka määritellään Tin Can -paketin yhteydessä. Raportit rakennetaan koostamalla LRS:ään kerätystä datasta jäseneltyjä kokonaisuuksia.

Koosteet voivat olla taulukkomuotoisia tai monipuolisempia visuaalisia esityksiä esimerkiksi siitä, millä tavalla opiskelijat ovat suoriutuneet tehtävistä ja kuinka paljon he ovat käyneet läpi oppimateriaalia. Visuaalinen analytiikka on analyttistä päättelyä vuorovaikutteisten visualisointikäyttöliittymien avulla sekä siihen liittyvä monitieteinen tutkimusalue. Se ottaa Keimin ym. [51] mukaan datalähtöisen lähestymistavan sijaan tavoitteeksi datan tehokkaan *käsittelyn* ihmislähtöisesti. Tavoitteena on kehittää visualisointien avulla tekniikoita ja työkaluja tehokkaaseen informaation esittämiseen, yhdistämiseen, sekä tietämyksen ja oivallusten luomiseen datasta.

Tällaisten visualisointien web-pohjaiseen toteutukseen on tarjolla erilaisia vaihtoehtoisia teknologioita. Data tai siitä muodostettu informaatio voidaan esittää visuaalisesti web-standardien mukaan esimerkiksi SVG-muotoisen vektorigrafiikan avulla. [52] XML-merkkaukieleen pohjautuvat SVG-elementit käyttäytyvät selaimessa samalla tavalla kuin tavalliset DOM-elementit. Flash- ja Java-sovellusten tapauksessa sovellukset ovat suljettuja ympäristöjä.

Sisällöntuotannon kannalta voi olla ongelmallista määrittää monipuoliselle monimedia-sisällölle mittareita, sillä sisällöt voivat erota toiminnoiltaan, tarkoitukseltaan ja toteutusteknologioiltaan. Tästä johtuen mittarit on toteutettu malliin niin, että sisällöntuottajilla ei ole mahdollista muokata niiden ominaisuuksia.

Kolmansien osapuolten sovellusten avulla luodut ja ulkopuolisten sisällöntuottajien luomat sisällöt eivät ole useinkaan helposti hallittavissa tarvittavan tarkalla tasolla, vaan niitä pystytään käsittelemään mallin sisällä ikään kuin kapseloituina moduuleina. Articulate Storylinen kaltaisten sovellusten yhteydessä voidaan asettaa tehtäviin esimerkiksi pisteraja, joka vaaditaan tehtävän asettamiseen suoritetuksi. Täydellinen hallittavuus saadaan mahdollistettua ainoastaan mallin sisään luodulle sisällölle, jonka rakenteiste-

tuille osille on asetettu selkeä tarkoitus. Poikkeuksena ovat standardoidut sisällöt, joihin on toteutettu rajapinta toimintoihin tarttumiseksi.

Vaikka toteutuksessa ei anneta sisällöntuottajalle mahdollisuutta muokata tarkkailtavia mittareita tai säätää niiden painoarvoa, on otettava huomioon, että opettajan on hallittava oppimateriaalisällön lisäksi myös toteutuskertojen opiskelijadataa, jota kertyy käyttäjien vuorovaikuttaessa sisällön kanssa. Oppimisobjekteille asetettu metadata on tässä tapauksessa suuressa roolissa. Se voi toimia raporteissa esimerkiksi kurssin, oppiaineen tai käsitteistön mukaisena suodatusmahdollisuutena.

4. TAPAUSTUTKIMUKSEN TULOKSET

Työssä toteutettu tapaustutkimus sisälsi ilmiöpohjaisen oppimisaihion käyttöönoton, kokemukset ja tulokset. Mallin soveltuvuudesta ja sitä hyödyntävistä sisällöntuotantotyökaluista ja –prosesseista saatiin tietoa opiskelijoiden, opettajien sekä sisällöntuottajien tasolla. Teknisen toteutuksen näkökulmasta toteutuksen käyttökelpoisuudesta saatiin tietoa toteutustyön kautta, sisällöntuottajan tarkemmalla haastattelulla sekä vertailemalla valmista mallia olemassa oleviin käytäntöihin.

4.1 Mallin soveltuvuus

Ilmiöpohjaista oppimisaihiota pilotoitiin PKKY:n opiskelijoilla kahdessa vaiheessa. Pilottivaiheen ensimmäisessä vaiheessa testattiin ympäristön teknistä toimivuutta, toisen iteraation aikana pyrittiin keräämään opiskelijoiden kokemuksia.

Oppimateriaalia käytettiin hyödyntäen Valamis-ympäristöä, joka toteutti opiskelijoiden käyttäjähallinnan, LRS-tietovaraston sekä palautekanavana toimivan keskustelufoorumin. Opetuskokeilun ensimmäinen vaihe alkoi 26.2.2015, johon osallistui 7 opiskelijaa. Osallistujat valikoituivat halukkuuden perusteella eri opinto-ohjelmista.

4.1.1 Käyttöönotto

Ennen pilottivaihetta materiaalia tuotettiin ympäristön käytettäväksi Ilmiöpohjaistimen avulla. ”Case Kaara – Ilmiönä auto” –tarina koostettiin sisältämään matematiikan, kemian ja fysiikan ammatillisen oppilaitoksen opetussuunnitelman mukaisia sisältöjä.

Opetuksen aikana tuotettiin myös lisää materiaalia ympäristöön, josta tiedotettiin opiskelijoille Valamis-oppimisalustan foorumin kautta sekä sähköpostilla. Keskustelufoorumia käytettiin tiedotuskanavana toimimisen lisäksi keräämään palautetta käyttäjiltä. Järjestelmä toimi odotetusti, eikä teknisistä ongelmista tullut palautetta opiskelijoilta. Opetuskokeilun aikana opiskelijoilta saatu palaute keskittyi varsinaiseen opetussisältöön teknisen toimivuuden sijaan. Yhtenä teknisenä ongelmana havaittiin kuitenkin opettajien puolelta Storyline-tehtävien toimimattomuus mobiilialustoilla ja tietyissä selaimissa.

Opetuskokeilun toimivuutta ja opiskelijoiden aktiivisuutta voitiin tarkastella Valamis-ympäristöön tallennetun LRS-datan perusteella, johon oli kirjattu opiskelijoiden oppimisaihiossa suorittamat tapahtumat. Opiskelijat olivat tietoisia siitä, että heidän oppimistapahtumiaan tallennetaan tietokantaan. Opettajat pystyivät hyödyntämään dataa

Valamis-ympäristön tarjoamien raporttien kautta ohjaus- ja arviointityössä. Käytetyt mittarit soveltuivat opiskelijoiden suoritusten määrittämiseen ja niistä tulkintojen saamiseen.

Aktiivisuusdataa tarkastellessa huomattiin, että opiskelijat eivät juurikaan käyttäneet itsearviointilomaketta, joka oli tarkoitettu kokemuksellisen datan keräämiseen opiskelijoiden osaamisesta materiaaliin liittyen. Myöhemmissä kurssitoteutuksissa opiskelijaa voidaan motivoida refleктоimaan osaamistaan henkilökohtaisilla palautenäkymillä, jotka koostavat osaamisen raportiksi. Arvioinnin käyttöä voidaan helpottaa vähentämällä lomakkeen mittarien määrää ja tuomalla se selkeämmin näkyväksi. On myös mahdollista asettaa tietyt artikkelit tai aihealueet läpikäydyksi vasta sen jälkeen, kun itsearviointi on suoritettu. [45]

4.1.2 Kokemuksia mallista ja sisällöntuotannosta

Pilotin jälkeen suoritettun haastattelun perusteella sovellettu malli ja sisällöntuotantotyökalut toimivat odotetusti. Käytännössä sisällöntuotantoprosessin kulku eteni niin, että opettajat tuottavat ilmiöiden ja käsitteiden sisällöt tutuilla työkaluilla (esimerkiksi tekstinkäsittelyohjelmilla) ennen sen syöttämistä järjestelmään. Tämän jälkeen materiaali toimitetaan joko sisällöntuottajalle tai syötetään itse järjestelmään. [45]

Sisällöntuotannossa teknologiat mahdollistavat uudelleenkäytettävien oppimisobjektien luomisen. Opettajien ja sisällöntuottajien roolit ovat monesti päällekkäisiä, jolloin uuden opintojakson perustamisen yhteydessä opettajat luovat sisältöä. Sisältö perustuu usein aikaisemmin luotuun informaatioon, jota päivitetään ja järjestellään uudelleen riippuen opetustavoitteista, -välineistä ja -menetelmistä.

Sisällöntuotantoprosessissa lähdettiin liikkeelle varsinaisesta opintojakson sisältämästä opetussuunnitelman mukaisesta sisällöstä, jonka jälkeen laadittiin ilmiöt kehystävä tarina. Rajattua ilmiöpohjaista mallia sovellettiin myös ”väärällä tavalla” käytännön suoritushjeita käsittävien sisältöjen tuotantoon. Liialliset rajoitukset sisällön luomisessa, joiden tavoite oli linjata sisällön suuntaa ilmiöpohjaiseksi oppisisällöksi, koettiin turhiksi ja sisällöntuotantoa vaikeuttavaksi.

Tuotantoprosessi koettiin suoraviivaiseksi ja parannukseksi edelliseen toimintatapaan, sillä usein oppimisalustojen sisällöntuotantovälineet koetaan liian monimutkaisiksi. [45] Ilmiöpohjaistin toteutettiin suoraan PKKY:n vaatimusten perusteella, ja sen toiminnallisuudet vastasivat suoraan heidän tarpeisiinsa.

Malliin rakennettu ilmiöpohjainen oppimateriaalikokonaisuus oli hierarkialtaan syvä, mikä koettiin mahdolliseksi ongelmaksi sisällön löydettävyyden kannalta. Ilmiöpohjaisessa mallissa olennainen rakenteen hallittavuus on ristiriidassa löydettävyyden kanssa. Sisältö on löydettävämpää, kun informaation hierarkia on matala. [46] Tässä tapaukses-

sa sisältöobjekteja tulisi yhdistää suuremmiksi kokonaisuuksiksi, jotka esitetään yksittäisessä oppimisobjektissa.

Mallia ei voitu suoraan asettaa Tin Can –rajapinnan mukaisiin kehyksiin, vaan sen rakentamiseksi tuli tehdä tiivistä yhteistyötä oppimisalustan kehittäjien kanssa. Koska Tin Can API on vielä uusi teknologia eikä esimerkkejä toteutuksista ole kovinkaan paljon, on myös selkeät ja joustavat toimintatavat vielä standardoimatta. Kokemusten perusteella oppimisalustoissa Tin Can –tuki ja dataa käyttävä LRS on keskeneräinen vain osittain suositusten mukaan toteutettu.

4.2 Raporttinäkymät

Eräs työn tavoitteesta oli tutkia, miten palautenäkymät kytketään oppimateriaalin käyttöprosessiin, ja tätä kautta materiaalimalliin. Tin Can -rajapinnan hyödyntäminen osoitautui pilottitoteutuksen aikana toimivaksi tekniseksi tavaksi kerätä dataa opiskelijoiden toiminnasta niin, että se oli mahdollista esittää opettajille hyödyllisessä muodossa. Yksilöidyt ja metadatala rikastetut oppimisobjektit antavat mahdollisuuden ryhmitellä niitä eri osa-alueisiin, joita raporteissa voidaan tarkastella joko yksittäisten opiskelijoiden tai opiskelijaryhmien tasolla. LRS-tietovarastoon kerätty, opiskelijoiden aktiivisuudesta kertyvä data voidaan muuntaa analysoitavaan muotoon visualisointien avulla, jotka koostavat ja esittävät opiskelijoiden aktiviteetit erilaisiin käyttötarkoituksiin kohdenneissa visuaalisissa raporttinäkymissä.

Pilotin aikana tunnistettiin kolme erilaista vaihetta, missä raportointi katsottiin erityisen hyödylliseksi. [45]

- **Kurssin aikana** on tarve saada ajankohtaista tietoa opiskelijoiden toiminnasta. Verkko-oppimisessa näkymättömät vuorovaikutustilanteet voidaan datan keräämisen avulla saada näkyviksi, jolloin opettajalle tarjoutuu mahdollisuus tehdä tarpeen tullen interventioita. Tämä tieto voidaan esittää puumuotoisessa tilannekaaviossa sekä arviointitaulukossa, johon opiskelijoiden suoritukset voidaan koostaa.
- **Arviointivaiheessa** on hyödyllistä saada koosteet oppijoiden suoriutumisesta ja itsearviointinnasta. Tapauksessa voidaan käyttää tilannekaaviota sekä arviointitaulukkoa.
- **Uuden kurssin suunnitteluvaiheessa** pystytään hyödyntämään tehtäviin ja materiaalin osa-alueisiin käytettävää aikaa sekä opiskelijoiden itsearviointia (kuva 6). Lisäksi on hyödyllistä havaita oppimateriaalin solmukohdat (kuva 7).

Pilotissa Arcusys toteutti Valamis-ympäristöön opettajan työkalut, jotka koostavat Tin Can -rajapinnan kautta kerätyn LRS-datan erilaisiksi näkymiksi.

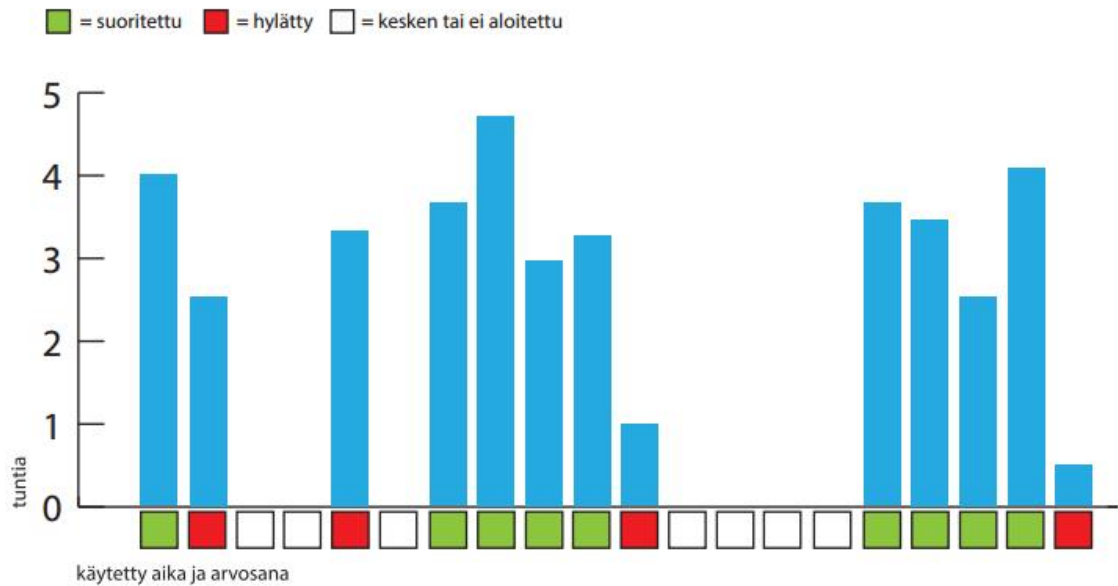
- Taulukkomuotoinen näkymä suoritetuista tehtävistä ja luetuista artikkeleista ryhmiteltynä sisältötyypeittäin.
- Puumuotoinen tilannekaavio yksittäisen opiskelijan läpikäymistä artikkeleista ja tehtävistä. Visualisoi tarinan rakenteen hierarkkisena puuna, ja asettaa niille arvon {suoritettu, kesken, hylätty, ei_aloitettu}. Mikäli solmun kaikkia lapsisolmuja ei ole suoritettu, on myös tämä solmu kesken.

Toteutettujen raporttien lisäksi hankkeen aikana suunniteltiin mahdollisia näkymiä, joiden toteutuskelpoisuutta voitiin arvioida pilotin aikana kerätyn opiskelijadatan perusteella. Seuraavissa aliluvuissa tarkastellaan kolmea suunniteltua näkymää mallin keräämän datan perusteella.

4.2.1 Arviointitaulukko

Kooste opiskelijoiden suorittamasta materiaalista voidaan visualisoida matriisityyppisellä näkymällä eli eräänlaisena taulukkona, jossa esitetään opiskelijat riveinä ja materiaalin osat sarakkeina ryhmiteltynä osa-alueittain - esimerkiksi sisältötyyppien ja oppiainneiden mukaan. Luokitteleva metadata on saatavilla mallin mukaisesta sisällöstä, joka voidaan upottaa Tin Can -lauseen kontekstiin. Taulukon solu voi saada jonkun arvoista {*suoritettu*, *ei_suoritettu*, *ei_aloitettu*}, jotka määräytyvät sillä, onko kyseinen opiskelija tuottanut solua vastaavan oppimisobjektin yhteydessä Tin Can -lauseen. Tapauksessa, jossa oppimisobjektin tila opiskelijan kohdalla on *ei_suoritettu*, ”attempted”-verbi on laukaistu mutta ”experienced”-verbi puuttuu. Taulukkoa on tarve järjestää opiskelijoiden menestyksen ja suoriutumisen mukaan. Mikäli kurssilla on suuri määrä osallistujia ja halutaan tarkastella yksittäistä opiskelijaa, voidaan hyödyntää ”table lens” -tekniikkaa, jossa taulukon valittua riviä korostetaan suurentamalla sitä muihin riveihin verrattuna. [47]

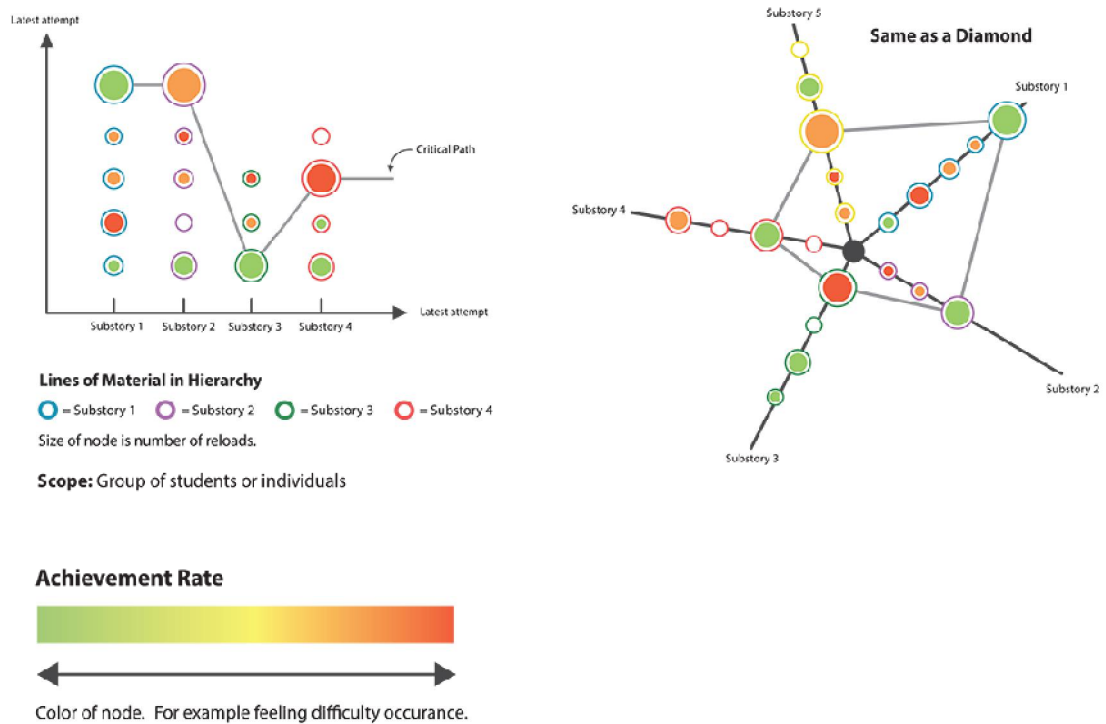
4.2.2 Aikakaavio



Kuva 13. Luonnos aikakaaviovisualisoinnista (PKKY)

Suunnitellussa aikakaaviossa esitetään pylväsdiagrammina opiskelijakohtaisesti tehtäviin käytetty aika. Pylväiden alla on tehtäviin liittyvä arvosana {suoritettu, hylätty, kesken, ei_aloitettu}. Teknisesti ajan mittaaminen selainrajapinnassa tuottaa monia haasteita, eikä se ole jokaisessa tilanteessa luotettavaa. Pilottitoteutuksessa pyrittiin mittaamaan artikkeleissa vietettyä aikaa tavalla, jossa Tin Can -lauseen laukaisija suoritettiin opiskelijan poistuessa sivulta hyödyntämällä JavaScriptin beforeunload-tapahtumankuuntelijaa. Tehtäviin käytetty aika on helpompi mitattava, sillä niihin liittyy useita eri vaiheita, joihin mittauspiste voidaan asettaa.

4.2.3 Kriittinen polku



Kuva 14. Luonnos kriittisen polun visualisoinnista (PKKY)

Kuvassa 12 on esitetty PKKY:n luonnos visualisoinnista, joka pyrkii esittämään oppimateriaaliin liittyvän “kriittisen polun” sekä käyttäjäkohtaisesti että ryhmittäin. Kriittisellä polulla pyritään kuvaamaan opetusmateriaalissa esiintyvät solmukohtat, joiden painoarvo määräytyy yrityskertojen sekä itsearviointitiedosta määritetyn koetun vaikeustason perusteella. LRS-datan perusteella voidaan laskea latauskerrat. Materiaalin koetun vaikeustason voidaan käyttää itsearviointitietoa.

Tin Can -rajapinnan kautta on mahdollista määrittellä myös tehtävän konteksti (esimerkiksi sijainti tai osio oppimateriaalikonaisuudessa), mutta hankkeen toteutuksessa oli helpompaa ryhmitellä sisällöt Valamis-oppimisolustan käyttöön tuotetun XML-konfiguraation perusteella. Pilottitoteutuksessa kontekstina käytettiin ainoastaan yksittäistä kurssia.

Tarinoiden tarkka konteksti voidaan jatkossa määrittellä esimerkiksi Tin Can -lauseissa esiintyvän yksiselitteisen URI:n perusteella, joiden perusteella mitattavaa objektia voidaan luokitella eri tasoihin. [28] Haasteen tälle asettaa materiaalmalli, jossa sisältöobjektilla ei ole välttämättä tarkkaa kontekstia, vaan siihen on mahdollista päästä useampaa eri polkua.

4.3 Sisällöntuotantoympäristön jatkokehitys

Jotta artikkelien sisältö olisi selkeämpää hahmottaa, olisi mahdollista käyttää teknisenä ratkaisuna selainpohjaista WYSIWYG-muokkainta XML-artikkelien editointiin. Tässä tapauksessa artikkeli näkyy sisällöntuottajalle sellaisenaan ilman lomakenäkymää. Esimerkiksi sisällöntuotantotyökalujen kartoituksessa tarkasteltu SDL Xopus [43] on mahdollinen vaihtoehto. WYSIWYG-muokkaimesta riippumatta artikkelien sisältämät metatiedot olisi mahdollista syöttää ainoastaan perinteisempien lomake-elementtien avulla.

Tin Can –rajapinnan eräs vahvuus on siinä, että se mahdollistaa ulkoisten resurssien käyttämisen oppimisobjekteina, joiden käytöstä saadaan palautetta. Kysymys siitä, yhdistetäänkö sisällöntuotantovälineet oppimisolustan toimintoihin, on vaikea, sillä tasapainottelu uudelleenkäytettävän sisällön ja alustariippuvaisuuden välillä voi olla käytännössä hankalaa.

Koska Ilmiöpohjaistin on osaltaan räätälöity Valamis-ympäristöä varten, sen soveltaminen muissa yhteyksissä vaatii pieniä muutoksia. Oppimisolustaan toteutettiin toiminto, jonka avulla opettaja voi liittää sinne tarinamuotoisen materiaalin syöttämällä konfiguraatitiedoston sijainnin. Yksittäisiä oppimisobjekteja voidaan käyttää myös muissa Tin Can –paketteja tukevissa alustoissa, mutta niistä ei luoda valmista tarinamuotoista kokonaisuutta.

On tavallista, että oppimateriaalia muokataan kurssin aikana. Materiaalin päivityksestä ei tullut automaattista ilmoitusta käyttäjille, mutta toiminnallisuus olisi mahdollista toteuttaa oppimisolustan tasolla. Esimerkiksi Valamis päivittää konfiguraatitiedoston tietyin väliajoin, ja mikäli muutos havaittaisiin, voitaisiin uudet lisätyt sisällöt korostaa. Versiointitoiminnallisuus selkeyttäisi prosessia entisestään, ja metadatan avulla olisi mahdollista kuvata esimerkiksi versiokohtaiset muutokset käyttäjille esitettynä.

Tulevaisuudessa toteutettu sisällöntuotantoympäristö sidottaneen vahvemmin Valamis-oppimisympäristöön, jolloin se toteutetaan Arcusysin mukaan todennäköisesti Liferay-portlettina. [53] Mikäli materiaalia on tarkoitus ylläpitää yli oppimisolustan elinkaaren, voi alustan vaihto merkitä materiaalimallin uudelleensuunnittelua. Toteutuksessa pyrittiin muodostamaan mahdollisimman yleiskäyttöinen malli, mutta lopulta päädyttiin sitomaan se sisällöntuotantotyökalujen osalta myös oppimisolustaan. Alustaan sitominen kuitenkin helpottaa sisällöntuotantoa, sillä tällöin sen mahdollisuudet ja rajoitukset tunnetaan.

4.4 Tulosten arviointia

Ilmiöpohjainen opiskelu on erityyppistä kuin oppiainekeskeinen opiskelu, mikä vaatii opintojen suunnittelulta uudentyypistä lähestymistapaa. Pilottitoteutuksessa saatujen kokemusten perusteella toteutettu malli ja sisällöntuotantovälineet vastasivat hankkeessa määriteltyihin ilmiöpohjaisen opetuksen asettamiin vaatimuksiin. Tuloksissa tulee ottaa huomioon, että sovellettu pedagoginen malli ja sen pohjalta laadittu materiaalmalli on ainoastaan yksi, tarinapohjainen lähestymistapa oppimateriaalin ilmiöpohjaisuuteen. Ei ole välttämättä yhtä ainoa tapaa toteuttaa ilmiöpohjaista verkko-opintojaksoa, mutta on selvää, että sellaisen toteuttaminen on mahdollista.

Jotkin sisällöntuotannon asettamat vaatimuksista työkaluille olivat organisaatiokohtaisia (esim. visuaaliset ohjeistukset), eikä niitä voida välttämättä soveltaa jokaiseen tapaukseen. Rakenteisessa sisällöntuotannossa pyritään eriyttämään esitystapa ja sisältö mahdollisimman pitkälle. [9]

Mallin yleistäminen yleisemmälle tasolle voi olla hankalaa opetusorganisaatiokohtaisten tavoitteiden takia. Mikäli korkeamman tason organisaatio, kuten Opetushallitus, määrittäisi ilmiöpohjaisen opetussuunnitelman, voitaisiin malli rakentaa tarkemmin sen varaan. Vastaavaan malliin pohjautuvaa oppimisaihiota on kuitenkin mahdollista lähteä rakentamaan tulevaisuudessa muillakin työvälineillä.

4.5 Suuntaviivoja ilmiöpohjaiselle verkko-oppimateriaalille

Kehitystyöstä sekä pilotista saatujen kokemusten perusteella voitiin koostaa ilmiöpohjaiselle verkko-oppimateriaalille hyviä käytäntöjä, joita voidaan hyödyntää tulevissa toteutuksissa. Tässä yhteydessä tarkastellaan verkko-oppimateriaalin laatua sisällöntuottajien ja opettajien näkökulmasta keskittyen sisältömalliin, sen tuottamiin palautteisiin sekä sisällöntuotantovälineisiin.

Ainoa olennainen teknologia toteutuksen kannalta oli Tin Can API. Toteutusteknologioita olennaisemmiksi tekijöiksi ilmiöpohjaisuuden toteuttajina valikoitui odotetusti mallin toteuttavat rakenteet ja välineet yleisemmällä tasolla.

Tuotettua mallia ja sen pohjalta luotua sisältöä voidaan suunnata suositeltavaan, yleistettyyn suuntaan erilaisin kriteerein. Esimerkiksi Opetushallitus on tuottanut verkko-oppimateriaalin laatukriteerit [21]. Kriteeristöjen lisäksi toteutusta voidaan verrata standardinmukaisiin sisältöihin, jotka toteuttavat vastaavat toiminnallisuudet. Seuraavissa aliluvuissa kuvataan sisältömalliin, prosessiin ja työkaluihin sekä tiedon keruuseen liittyviä, tunnistettuja huomioita.

4.5.1 Sisältömalli

Pilotin aikana tunnistettiin sisältömalliin liittyen seuraavia huomioita:

- Oppimisobjektien tulee olla modulaarisia rakenteita, jotka toimivat itsenäisinä kokonaisuuksinaan.
- Oppimisobjektien tulee olla uudelleenkäytettäviä
- Sisältöjä tulee voida versioida
- Arvioitavien sisältöjen tulee perustua yhteisesti sovittuun opetussuunnitelman käsitteistöön.
- Oppimisobjektit tulee eritellä opetussuunnitelman mukaisiin sekä ilmiöt kuvaaviin luokkiin
- Metadatan on hyödynnettävä mahdollisimman paljon sisällön kuvaajina

Modulaariset rakenteet olivat DITA-periaatteen [15] mukaan toteutettuja, tarpeeksi lyhyitä mutta itsenäisesti toimivia kokonaisuuksia. Toteutustapa vauhditti sisällöntuotantoa ja koettiin selkeäksi. Lisäksi se mahdollisti palasten uudelleenkäytettävyyden useissa eri yhteyksissä yhden instanssin sisällä.

Käyttöön otettuun malliin ei toteutettu versiointimahdollisuutta. Ilmiöpohjaistimia voi olla tällä hetkellä monta instanssia, mikä voi vaikeuttaa informaation hallintaa. Mikäli mallin käyttö laajenee, kehittyneempi sisällönhallinta menisi tällä perusteella yhä enemmän kohti DITA-perustaista mallia, joka sisältäisi esimerkiksi henkilö- ja kurssikohtaisen sisällön versioinnin. DITA:sta on tarjolla myös opetusmateriaaliin suunniteltu spesifikaatio. [36]

Mallin duaalinen rakenne osoittautui pilotissa toimivaksi. Rakenteellisesti toteutettu malli jakaantuu kahteen eri tyyppiseen sisältöobjektiin. Kurssit kehystettiin pilottitoteutuksessa tarinoilla, mutta samalla periaatteella voidaan luoda myös muita monialaisia opintokokonaisuuksia.

Metadatan hyöty tuli ilmi oppiaine- ja kurssikohtaisissa palautenäkymissä, joiden toimivuus perustui juuri sisältöobjektien rikkaaseen metadataan. Sisällöntuottajien tulisi olla sitoutuneita täyttämään riittävä määrä tällaista informaatiota. Uutena haasteena voi tosin tällöin ilmetä kasvavan metadatan hallinta.

4.5.2 Sisällöntuotantoprosessit ja –työkalut

Olenneisimmat havaitut sisällöntuotantoprosesseihin ja –työkaluihin liittyvät huomiot:

- Sisältöjen aukot tulee voida hahmottaa
- Oppimisobjektien tulee sijaita keskitetyssä dokumenttipankissa tai repositoriossa, josta ne ovat poimittavissa kurssiainioihin.
- Sisällöntuotantovälineiden täytyy olla tuttuja käyttäjille tai tarpeeksi helppokäyttöisiä.
- Tuotantovälineet eivät saa rajoittaa liikaa sisällöntuottajien prosesseja.
- Julkaisu tulee mahdollistaa myös muihin järjestelmiin.

Sisällöntuotannon työmäärää voidaan pienentää helpottamalla eri osa-alueiden aukkojen hahmottamista. Ilmiöpohjaista opintojaksoa rakentaessa sisällöntuottaja käsittelee pääasiassa kahta erityyppistä rakennetta: ilmiöpohjaista tarinaa sekä oppiaineiden käsitteistöä. Riskinä on, että molempien rakenteiden samanaikaisesta käsittelestä tulee liian kuormittavaa, ja opiskelija ei koe valmista rakennetta koherenttina. Pilotissa saatujen kokemusten perusteella tarinoiden ja oppiainesisältöjen eriyttäminen auttaa sisältökokoaisuuden hahmottamista, mutta ei ratkaise koko ongelmaa. Jatkokehityksessä kurssisisältöjen taustalla olevaa käsitteistöä voidaan verrata rakennettavan tarinan käsitteistöön, minkä avulla esitettäisiin sisällöntuottajalle. Tämä vaatii erikseen ylläpidettävän asiasanaston, joka pohjautuu johonkin yhteisesti sovittuun kokonaisuuteen, kuten opetussuunnitelmaan.

Sisällön tuotannosta saatujen kokemusten perusteella voitiin päätellä, että vaikka työkalu koettiin helppokäyttöiseksi ja selkeäksi, on sisällöntuotantoprosessi edelleen vahvasti liitoksissa perinteisempiin työpöytäsovelluksiin. Ratkaisuna voidaan tarjota yksinkertaisempaa menetelmää sisällön syöttämiseen. ”Productification”-tyyppistä menetelmää voidaan soveltaa tuttuja työkaluja käytettäessä, jolloin esimerkiksi Excel-taulukkoon syötetyt tiedot voidaan putkilinjaston avulla muuntaa haluttuun muotoon ja viedä järjestelmään. [48] Hierarkioiden ja suhteiden muodostaminen tällä tavalla voi kuitenkin olla haastavaa, jolloin rakennetta visualisoivalla työkalulla on paikkansa tarinan ja sisältöjen suhteiden rakenteistamisvaiheessa.

Toteutuksessa luotiin yhden Ilmiöpohjaistin-instanssin yhteyteen toimivaksi havaittu dokumenttipankki, joka toimii DITA-mallin kaltaisena repositoriona. Jos ilmiöpohjainen opetus yleistyy tulevaisuudessa, voidaan opetussuunnitelman mukaisia sisältökokoaisuuksia luoda keskitettyyn repositorioon. Repositoriosta voidaan tällöin poimia sisältöjä oppilaitosten tarpeiden mukaan, joita voidaan versioida ja yhdistellä toisiinsa.

Mikäli sisällöntuotannossa halutaan tuottaa muissa ympäristöissä toimivia Tin Can -paketteja, tulee julkaisuprosessiin tehdä pieniä muutoksia. Tällöin uudelleenkäytettävä sisältöpaketti julkaistaisiin pakattuna tiedostona, joka sisältäisi vastaavan hakemistora-

kenteen. Tämän lisäksi koko materiaalin kattava navigointijärjestelmä tulisi esittää sisältöobjektien yhteydessä, eikä syöttää sitä erillisenä oppimisympäristöön.

4.5.3 Tiedon keruu ja palaute

- Palautenäkymissä hyödynnetään sisällön metadataa
- Tarkkailtavia mittareita tulee olla vain muutama. Tärkeimmät Key Performance Indicator (KPI) –mittarit on määritettävä.
- Mittareista on voitava tehdä päätelmiä. Visuaalisen analytiikan menetelmät ja työkalut mahdollistavat soveltuvien vuorovaikutteisten näkymien toteuttamisen.
- Kerättyä dataa hyödynnetään opetuksen ja arvioinnin tuen lisäksi myös laajemmin jopa hallinnon tasolla. Kursseja pystytään potentiaalisesti kehittämään

Palautenäkymät olivat olennainen osa ilmiöpohjaisen opetuksen mahdollistamista. Sisältöjen metadatan perusteella pystyttiin ryhmittelemään ilmiöpohjainen sisältö eri oppiaineisiin kuuluviksi, jolloin oppiainekohtainen arviointi ja opiskelijoiden tarkastelu oli mahdollista.

OPH:n verkko-oppimateriaalin laatukriteereissä nostetaan esille sen ”kehittäminen seurannan perusteella”. Käyttäjien palautetta ja käyttötietoa voidaan koota Tin Can API:n avulla, ja tulokset pystytään esittämään palauteäkymässä materiaalin kehittämiseksi. Vaikka Tin Can ei ole varsinainen standardi, siinä määritellään hyviä käytäntöjä, joita kasvava määrä oppimisalustoja tukee.

Olemassa olevat menetelmät mahdollistavat jo lähes millaisen tahansa datan keräämisen. On kuitenkin riski, että dataa kerätään liikaa. [55] Dataa analysoitaessa tuli ilmi, että LRS-data saattaa kasvaa opiskelijoiden määrän kasvaessa sietämättömän suureksi ja redundantiksi. KPI-mittarit määritellään esimerkiksi web-analytiikassa tärkeimmiksi mitattaviksi arvoiksi, joita peilataan tiettyihin tavoitteisiin. Myös oppimissisällöissä olisi määriteltävä vastaavat indikaattorit, jotka kuvaavat parhaalla tavalla tärkeimpiä oppimistapahtumia.

Kursseja pystytään seuraamaan ja kehittämään, mikäli analytiikkatyökalut ovat riittävän laadukkaita. Kerättyä dataa pystytään hyödyntämään opetuksen ja arvioinnin tuen lisäksi potentiaalisesti jopa hallinnon tasolla. [38] Mittarit tulee valita ensisijaisesti sillä tavalla, joka tukee opetusorganisaation päätöksentekoa.

5. YHTEENVETO

Diplomityössä kokeiltiin uudentyyppistä tapaa esittää ja tuottaa rakenteinen oppimateriaali ilmiölähtöisyyden ehdoilla niin, että opiskelijoiden toimintaa pystytään havainnoimaan. Uudenaista oppimateriaalin hyödyntämistapaa sovellettiin myös oppimateriaalin ulkoistamisella oppimisympäristöstä niin, että oppimistulokset tallentuvat ulkopuolella sijaitsevaan Valamis-järjestelmään. Työn tuloksena toteutettiin rakenteinen malli ilmiöpohjaiselle oppimateriaalille sekä sisällöntuotantoympäristö Ilmiöpohjaistin, jossa mallin mukaista sisältöä on mahdollista luoda ja hallita. Sisällöntuotantoprosesseja tarkasteltiin vertaamalla vaatimusten perusteella toteutettua kokonaisuutta valmiisiin malleihin, joista DITA:an perustuvat ovat jalostuneimpia vaihtoehtoja.

Työn tulosten perusteella voidaan todeta, että rakenteisen dokumentaation periaatteiden avulla oppimateriaalia voidaan tuottaa, hallita ja julkaista ilmiöpohjaisen opetuksen edellyttämällä tavoilla. Mallista ei tullut resurssien puutteesta täydellinen, mutta se vastasi opetusorganisaation tarpeisiin. Mahdollista jatkokäyttöä ajatellen koottiin sisällöntuottajien sekä teknisten kehittäjien kokemusten ja teoreettisen tarkastelun kautta suositukset ilmiöpohjaiselle materiaali- ja sisällöntuotantomallille, jonka perusteella olemassa olevaa toteutusta voidaan jalostaa eteenpäin.

Rakenteiseen dokumentaatioon perustuvan oppimisaihion sisällönhallinnan lisäksi tarkasteltiin erityisesti sen paketoimista Tin Can -rajapintaa hyödyntävään muotoon. Teknologia osoittautui erittäin hyödylliseksi ja sen päälle voi jatkojalostaa erilaisia opiskelijoiden toimintaa koostavia raporttinäkymiä. Suurimpina haasteina on määrittää selkeät ja luotettavat mittarit opiskelijoiden menestykselle sekä sitouttaa opiskelijat tuottamaan itsearviointitietoa vapaaehtoisesti. Rikas data mahdollistaa käyttökelpoiset palautenäkymät, ja tulevaisuudessa tätä dataa voidaan hyödyntää laajemminkin opetusorganisaatioiden kehittämisessä. Riippumatta pedagogisista lähtökohdista, datalähtöinen lähestymistapa antaa mahdollisuuden laadukkaaseen oppimissisällön ja -kokemuksen toteuttamiseen.

LÄHTEET

- [1] Pedagogiset mallit. Tietotekniikan opetuksen perusteet. Jyväskylän Yliopisto. (2010). Saatavilla (viitattu 16.5.2016):
<https://koppa.jyu.fi/avoimet/mit/tietotekniikan-opetuksen-perusteet/Opetusmenetelmista-ja-lahestymistavoista/pedagoginen-malli>
- [2] Rongas, A & Laaksonen, R. (2014). Ilmiöopas. Kokemuksia ilmiöopettamisesta - opettajilta toiselle. Hämeenlinnan kaupunki. Saatavissa (viitattu 16.5.2016):
http://wiki.eoppimiskeskus.fi/download/attachments/24873071/ilmioopas2013A_VO2.pdf
- [3] Verkko-opintojen pedagoginen malli. Pohjois-Karjalan koulutuskuntayhtymä. (2015). Saatavissa (viitattu 16.5.2016):
<https://pkky.arcusys.fi/documents/10184/0/Opintojen+pedagoginen+malli>
- [4] Rissanen, Riitta. (2006). Fenomenografia. Luku 5.5. kokonaisuudesta Anita Saaranen-Kauppinen & Anna Puusniekka. 2006. KvaliMOTV - Menetelmäopetuksen tietovaranto. Tampere: Yhteiskuntatieteellinen tietoarkisto. Saatavissa (viitattu 16.5.2016): <http://www.fsd.uta.fi/menetelmaopetus/>
- [5] SCORM Explained. Rustici Software. Saatavissa (viitattu 16.5.2016):
<http://scorm.com/scorm-explained/>
- [6] Korpela, J. (2007). Standardi, mikä se on? Sivu kokonaisuudesta *Datatekniikka ja viestintä*, osiosta *Normit, standardit, protokollat*. Saatavissa (viitattu 16.5.2016): <https://www.cs.tut.fi/~jkorpe/stand.html>
- [7] Pohjanen, L. (2004). Opetusteknologian standardointi: Materiaali- ja kurssimallit. Diplomityö. Tampereen teknillinen yliopisto.
- [8] Berking, P. (2014). Choosing Authoring Tools. ADL Instructional Design Team. Saatavissa (viitattu 16.5.2016): <http://www.adlnet.gov/wp-content/uploads/2013/08/Choosing-Authoring-Tools.pdf>
- [9] O'Keefe, Sarah. (2009). Structured authoring and XML. White paper. Research Triangle Park, NC: Scriptorium Publishing Services. Saatavissa (viitattu 16.5.2016): <http://www.scriptorium.com/structure.pdf>
- [10] About us. OASIS. (2016). Saatavissa (viitattu 16.5.2016): <https://www.oasis-open.org/org>

- [11] Fallside, D. C., Walmsley, P. (2004). XML Schema part 0: Primer Second Edition. W3C. Saatavissa (viitattu 16.5.2016): <http://www.w3.org/TR/2004/REC-xmlschema-0-20041028/>
- [12] Clark, J., & DeRose, S. (1999) XML path language (XPath) Version 1.0. W3C recommendation. W3C. Saatavissa (viitattu 16.5.2016): <http://www.w3.org/TR/xpath>
- [13] Didier, M. et al. (2000). Professional XML. Wrox Press Ltd. 1169 s.
- [14] Eberlein, K. J., Anderson, R. D & Joseph, G. (2010). DITA Version 1.2 Specification. OASIS. Saatavissa (viitattu 16.5.2016): <http://docs.oasis-open.org/dita/v1.2/os/spec/DITA1.2-spec.html>
- [15] Shafie, H. (2015). DITA for the Impatient. Pixware SARL. Saatavissa (viitattu 16.5.2016): <http://www.xmlmind.com/tutorials/DITA/DITA-book.pdf>
- [16] DITA Editor. oXygen XML Editor. SyncRO Soft SRL. Saatavissa (viitattu 16.5.2016): https://www.oxygenxml.com/xml_editor/dita_editor.html
- [17] DITA Open Toolkit. DITA Open Toolkit Project. (2016). Saatavissa (viitattu 16.5.2016): <http://www.dita-ot.org/>
- [18] Isaac, A., & Summers, E. (2009). SKOS Simple Knowledge Organization System Primer. W3C Working Group Note. W3C. Saatavissa (viitattu 16.5.2016): <http://www.w3.org/TR/skos-primer/>
- [19] ONKI-projektin ontologiaeditorien vertailuraportti. Kansalliskirjasto. (2013). Saatavissa (viitattu 16.5.2016): <https://www.doria.fi/bitstream/handle/10024/91429/ONKI-projekti-Ontologiaeditoriraportti-lopullinen-korj.pdf>
- [20] Neven, F., & Duval, E. (2002). Reusable learning objects: a survey of LOM-based repositories. Proceedings of the tenth ACM international conference on Multimedia (pp. 291-294). ACM. DOI: 10.1145/641007.641067
- [21] E-oppimisen laatukriteerit. EDU.fi. (2012). Saatavissa (viitattu 16.5.2016): http://www.edu.fi/verkko_oppimateriaalit/e-oppimateriaalin_laatukriteerit
- [22] XML Essentials. XML Technology. W3C. (2015). Saatavissa (viitattu 16.5.2016): <http://www.w3.org/standards/xml/>
- [23] XML Document Object Model (DOM). Microsoft Developer Network. (2016). Saatavissa (viitattu 16.5.2016): [https://msdn.microsoft.com/en-us/library/hf9hbf87\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/hf9hbf87(v=vs.110).aspx)

- [24] Moodle Docs 3.0. Moodle. (2016). Saatavissa (viitattu 16.5.2016): https://docs.moodle.org/30/en/Main_page
- [25] Plugin types. Moodle Docs 3.0. Moodle. (2016). Saatavissa (viitattu 16.5.2016): https://docs.moodle.org/dev/Plugin_types
- [26] Top LMS Software. Capterra. (2014). Saatavissa (viitattu 16.5.2016): <http://www.capterra.com/learning-management-system-software/?utf8=%E2%9C%93&v=4#infographic>
- [27] Experience API. Advanced Distributed Learning (ADL) Initiative, U.S. Department of Defence. (2013). Saatavilla: <https://github.com/adlnet/xAPI-Spec/blob/master/xAPI.md>
- [28] Statements 101. Tin Can API. Rustici Software. (2016). Saatavissa (viitattu 16.5.2016): <http://tincanapi.com/statements-101/>
- [29] What is an Activity Provider? Tin Can API. Rustici Software. (2016). Saatavissa (viitattu 16.5.2016): <http://tincanapi.com/activity-provider/>
- [30] ADL's Controlled Vocabulary. xAPI Vocabulary Working Group. (2015). Saatavissa (viitattu 16.5.2016): <http://www.adlnet.gov/expapi/verbs/>
- [31] Features. SCORM Cloud. Rustici Software. (2016). Saatavissa (viitattu 16.5.2016): <http://scorm.com/scorm-solved/scorm-cloud-features/>
- [32] Valamis Documentation. Arcusys. (2016). Saatavissa (viitattu 16.5.2016): <http://docs.valamis.arcusys.com/>
- [33] Articulate Storyline. Articulate Global Inc. (2016). Saatavissa (viitattu 16.5.2016): <https://www.articulate.com/products/storyline-why.php>
- [34] IMS Metadata Best Practice Guide for IEEE 1484.12.1-2002 Standard for Learning Object Metadata. Version 1.3 Final Specification. IMS Global Learning Consortium, Inc. (2006). Saatavissa (viitattu 16.5.2016): http://www.imspj.org/metadata/mdv1p3/imsmd_bestv1p3.html
- [35] Learning Tools Interoperability. IMS Global Learning Consortium. (2016). Saatavissa (viitattu 16.5.2016): <http://www.imsglobal.org/toolsinteroperability2.cfm>
- [36] Hunt, J. P & Bernard, R. (2005). DITA for learning content. IBM. Saatavissa (viitattu 16.5.2016): <http://www.ibm.com/developerworks/library/x-dita9a/>
- [37] Tervakari, A., Silius, K., Koro, J., & Alasuutari, A. (2014). Verkkopalvelun laadukkuuden arviointi: Oppimateriaali. Tampere: Tampereen teknillinen yliopisto,

- Intelligent Information Systems Laboratory. Saatavissa (viitattu 16.5.2016): <https://iislab.ee.tut.fi/piiri/content/verkkopalvelun-laadukkuuden-arviointi>
- [38] Baepler, P., & Murdoch, C. J. (2010). Academic analytics and data mining in higher education. *International Journal for the Scholarship of Teaching and Learning*, 4(2), 17.
- [39] Tervakari, A-M., Marttila, J., Kailanto, M., Huhtamäki, J., Koro, J. & Silius, K. 2013. Developing Learning Analytics for TUT Circle. In Ley, T., Ruohonen, M., Laanpere, M. & Tatnall, A. (Eds.) IFIP WG 3.4 International Conference, OST 2012, Tallinn, Estonia, July 30 - August 3, 2012, Revised Selected Papers. Series: IFIP Advances in Information and Communication Technology, Vol. 395. Springer-Verlag, 101-110.
- [40] LAK '11. 1st International Conference on Learning Analytics and Knowledge 2011. Saatavissa (viitattu 16.5.2016): <https://tekri.athabascau.ca/analytics/>
- [41] Duval, E. (2011). Attention please!: learning analytics for visualization and recommendation. Proceedings of the 1st International Conference on Learning Analytics and Knowledge (LAK '11). DOI: 10.1145/2090116.2090118
- [42] Tietoja PhET-projektista. University of Colorado Boulder. (2016). Saatavissa (viitattu 16.5.2016): <https://phet.colorado.edu/fi/about>
- [43] SDL Xopus. SDL. Saatavissa (viitattu 16.5.2016): <http://www.sdl.com/cxc/knowledge-delivery/xml-publishing-xopus/>
- [44] Downes, S. (2005). E-learning 2.0. eLearn magazine. ACM. Saatavissa (viitattu 16.5.2016): <http://elearnmag.acm.org/featured.cfm?aid=1104968>
- [45] V. Tuominen. Pohjois-Karjalan koulutuskuntayhtymä. [Haastattelu] 11.6.2015. Haastattelijana Juho Koro.
- [46] Whinton, K. (2013). Flat vs. Deep Website Hierarchies. Nielsen Norman Group. Saatavissa (viitattu 16.5.2016): <https://www.nngroup.com/articles/flat-vs-deep-hierarchy/>
- [47] Rao, R., & Card, S. K. (1994). The table lens: merging graphical and symbolic representations in an interactive focus+ context visualization for tabular information. Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 318-322). ACM. DOI: 10.1145/259963.260391
- [48] Nykänen, O. (2014). Productification for Collaborative Semantic Modeling. In A. Lugmayr, H. Franssila, & J. Paavilainen (Eds.), Proceedings of the 18th Academic MindTrek Conference 2014: "Media Business, Management, Content &

- Services", 4-6 November, 2014, Tampere, Finland. (pp. 1-6). (MindTrek Conference). New York, NY: The Association for Computing Machinery, ACM. DOI: 10.1145/2676467.2676470
- [49] Chosen (v1.4.2). Harvest. (2015). Saatavissa (viitattu 16.5.2016): <https://harvesthq.github.io/chosen/>
- [50] Tree. DHTMLX Documentation. Dinamenta, UAB. Saatavissa (viitattu 16.5.2016): http://docs.dhtmlx.com/tree__index.html
- [51] Keim, D., Andrienko, G., Fekete, J. D., Görg, C., Kohlhammer, J., & Melançon, G. (2008). Visual analytics: Definition, process, and challenges (pp. 154-175). Springer Berlin Heidelberg. DOI: 10.1007/978-3-540-70956-5_7
- [52] Scalable Vector Graphics (SVG) 1.1 (Second Edition). E. Dahlström, P. Dengler, A. Grasso, C. Lilley, C. McCormack, D. Schepers, J. Watt, J. Ferraiolo, J. Fujisawa, D. Jackson. W3C Recommendation. (2001). <http://www.w3.org/TR/2011/REC-SVG11-20110816/>. Viimeisin versio Saatavissa (viitattu 16.5.2016): <http://www.w3.org/TR/SVG11/>
- [53] SimpleXML. PHP Manual. PHP. (2016). Saatavissa (viitattu 16.5.2016): <http://php.net/manual/en/book.simplexml.php>
- [54] Statement design. Tin Can API. Rustici Software. (2016). Saatavissa (viitattu 16.5.2016): <https://tincanapi.com/statement-design/>
- [55] Schneider, W. (2012). Best Practices Updated. Mixpanel. Saatavissa (viitattu 16.5.2016): <https://mixpanel.com/docs/>
- [56] xAPIWrapper. Advanced Distributed Learning (ADL). (2016). Saatavissa (viitattu 16.5.2016): <https://github.com/adlnet/xAPIWrapper>