

Ilari Waloranta

Fysiikan laboratorioharjoitusten oppimisjärjestelmän toteutus
sisällönhallintajärjestelmällä

Tekijä Otsikko Sivumäärä Aika	Ilari Waloranta Fysiikan laboratorioharjoitusten oppimisjärjestelmän toteutus sisällönhallintajärjestelmällä 39 sivua + 1 liite 1.5.2011
Tutkinto	insinööri (AMK)
Koulutusohjelma	mediatekniikka
Suuntautumisvaihtoehto	digitaalinen media
Ohjaajat	yliopettaja Harri Airaksinen lehtori Jorma Surakka
<p>Insinöörityössä kehitettiin oppimisenhallintajärjestelmä ammattikorkeakoulun fysiikan laboratorioharjoitusten tueksi. Tavoitteena oli toteuttaa järjestelmä, joka tukee opiskelijoita oppimisprosessissa, tarjoamalla yhtenäisen käyttöliittymä oppimismateriaaleille sekä harjoitusten työnannoissa useammalle ryhmän jäsenelle ja ryhmälle samanaikaisesti. Järjestelmä tarjoaa harjoitusten ylläpitäjälle käyttöliittymän harjoitusten ylläpitoon ja yksittäisten oppilaiden suoritusten seurantaan. Sivustolle haluttiin tuki ammattikorkeakoulun käyttäjätunnuksilla kirjautumiseen, mahdollisuus useamman ryhmän jäsenen kirjautumiseen samaan harjoitukseen ja mahdollisuus sivuston jatkokehitykseen ja mukautumiseen fysiikan laboratoriotöiden oppimisvaatimusten mukaiseksi. Työ edellytti ylläpitäjän käyttöoppaan tekemistä, ja vaatimuksena oli selvittää avoimen lähdekoodin ohjelmilla.</p> <p>Koska jatkossa sivua ylläpitävät jatkokäyttäjät, tutkittiin toteutukseen ensisijaisesti sisällönhallintajärjestelmiä, jotka tarjoavat ylläpitotoimiin visuaalisen käyttöliittymän. Projekti-palvelin rajasi palvelimella suoritettavaksi ohjelmistokieleksi PHP:n ja tietokannaksi MySQL:n käytön. Toteutuksessa pyrittiin pitämään ylläpito mahdollisimman selkeänä ja eristämään PHP:llä toteutettu toiminnallisuus ylläpitäjälle näkymättömiin.</p> <p>Toteutuksessa päädyttiin käyttämään Drupal -sisällönhallintajärjestelmää, koska se mahdollistaa monipuolisen muokattavuuden, suuren määrän valmiita liitännäismoduuleita ja laajan dokumentaation. Ammattikorkeakoulun käyttäjätunnuksien CAS-kirjautumispalvelimen tuki löytyi valmiiksi toteutettuna moduulina. Työ toteutettiin Linux-ympäristössä.</p> <p>Työn tuloksena saatiin toimiva fysiikan laboratorioharjoitusten hallinnointijärjestelmä, joka on keväällä 2011 kokeellisessa opetuskäytössä. Saatujen kokemusten perusteella toteutus täyttää sille asetetut vähimmäisvaatimukset. Päätymisen sisällönhallinta-järjestelmän käyttämiseen oli oikea. Se tarjosi valmiita työkaluja, joiden kehittämiseen ei tässä projektissa olisi muuten ollut aikaa.</p> <p>Insinöörityön tuloksia voi käyttää Drupalin mahdollisuuksiin ja toimintaan tutustumiseen ja tuotteen jatkokehitykseen.</p>	
Avainsanat	sisällönhallintajärjestelmä, oppimisympäristö, Drupal, PHP, MySQL, AJAX

Author Title Number of Pages Date	Ilari Waloranta The implementation of the working environment of physics laboratory courses using the content management system 39 pages + 1 appendice 1.5.2011
Degree	Bachelor of Engineering
Degree Programme	Media Technology
Specialisation option	Digital Media Technology
Instructors	Harri Airaksinen, Principal Lecturer Jorma Surakka, Lecturer
<p>The purpose of this thesis was to develop a learning management system as a support for the physics laboratory courses.</p> <p>The objective was to execute a system that supports students in their learning process by offering a solid user interface for the learning material and for the assignments given to several group members and groups simultaneously. The system offers a user interface for the administrator, for the maintenance of the exercises, as well as for monitoring the students' performance in the exercises. The requirements for the site were: the support in registering with the Metropolia login and the possibility for several group members to register for the same exercise, as well as the possibility for further development of the site. The outcome required an admin user guide, as well as the usage of open source software.</p> <p>Since downstream users are going to maintain the site in the future, the research for the execution focused mainly on the properties of content management systems, as they offer a more visual user interface for maintenance. The project server defined PHP as the scripting language and MySQL as the database. The implementation was to be kept as explicit as possible.</p> <p>Drupal content management system was chosen to execute the project, because it enables diverse possibilities for editing, a large number of ready to use modules and extensive documentation. A module supporting the CAS login server for Metropolia user IDs was found as a ready to use set up. The project was executed in a Linux environment.</p> <p>The result of this project is a functioning management system for the physics laboratory course, which is being tested by the course this spring 2011. The decision to use a content management system for this project was the right choice, because it offered ready tools for use and, therefore, adjusted the time schedule.</p> <p>The results of my thesis can be used to learn about the features of Drupal, as well as for the further development of this product.</p>	
Keywords	Content management system, working environment, Drupal, PHP, MySQL, AJAX

Sisällys

Lyhenteet ja termit

1	Johdanto	1
2	Sisällönhallintajärjestelmät	3
2.1	Sisällönhallintajärjestelmät (CMS)	3
2.2	Oppimisenhallintajärjestelmät (LMS)	4
2.3	Sopivan hallintajärjestelmän valintakriteerit	5
3	Drupal -sisällönhallintajärjestelmä	7
3.1	Yleistä Drupalista	7
3.2	Asennusvaatimukset	8
3.3	Drupalin muokkaus ja teemat	9
4	Fysiikan laboratorioharjoitusjärjestelmän toteutus	12
4.1	Projektityöskentely etätyöpalvelimella	12
4.2	Tietokanta	14
4.3	Moduulit	17
4.4	Käyttäjänhallinta, käyttäjien kirjautuminen ja tietoturva	20
4.5	Varmuuskopiointi	27
5	Keskeiset ongelmakohdat, virheenjäljitys ja paikkaus	29
5.1	Ongelmakohdat	29
5.2	Paikkaukset	31
6	Drupal admin -käyttöliittymä	33
7	Yhteenveto	36
	Lähteet	39
	Liitteet	
	Liite 1. Ylläpitäjän käyttöohje	

Lyhenteet ja termit

.htaccess	<i>Hypertext Access</i> . Oletusnimi Apache:n hakemistokohtaiselle asetus-tiedostolle
AJAX	<i>Asynchronous JavaScript and XML</i> . Tekniikka, jonka avulla voidaan siirtää tietoa selaimen ja palvelimen välillä ilman, että koko www-sivua täytyy ladata uudelleen.
Apache	Avoimeen lähdekoodiin perustuva HTTP-palvelinohjelma
API	<i>Application Programming Interface</i> . Ohjelmointirajapinta
Bluetooth	Avoin standardi laitteiden langattomaan kommunikointiin lähietäisyydellä
CAS	Central Authentication Service
CKeditor	avoimen lähdekoodin WYSIWYG-tekstieditori
CMS	<i>Content management system</i> . Sisällönhallintajärjestelmä
CSS	<i>Cascading Style Sheets</i> . Verkkoselaimille tarkoitettu tyylimääräite.
DOM	<i>Document Object Model</i> . Ohjelmointirajapinta, jonka avulla voi muokata XHTML-dokumentteja
Drupal	WWW-pohjainen ohjelmistokehys, sisällönhallintajärjestelmä ja blog-järjestelmä. Ohjelmisto on kirjoitettu PHP:lla ja toimii erilaisissa tietokan-taympäristöissä
GD Library	Avoimeen lähdekoodiin perustuva, C:llä kirjoitettu ohjelma, jolla voidaan luoda ja muokata yleisimpiä kuvaformaatteja
Gedit	GNOME-työpöytäprojektiin kuuluva tekstieditori, joka julkaistaan GNU GPL-lisenssillä
Gimp	<i>GNU Image Manipulation Program</i> . Avoimeen lähdekoodiin perustuva ku-vankäsittelyohjelma, joka on tarkoitettu rasterikuvien muokkaamiseen
GNU	Ohjelmistolisenssi
Hash	Hajautustaulu, hakurakenne, eli avaimia arvoihin yhdistävä tietorakenne
HTML	<i>HyperText Markup Language</i> . Merkintäkieli, jolla kuvaillaan www-sivujen sisällön rakenne
IIS	<i>Internet Information Services</i> . Microsoftin kehittämä palvelinohjelmisto
ImageMagick	Grafiikan käsittelyyn tarkoitettuja ohjelmia ja niiden toiminnallisuuden to-teuttava kirjasto
Inkscape	Avoimeen lähdekoodiin perustuva ohjelmisto vektorigrafiikan muokkaami-seen
JSON	<i>JavaScript Object Notation</i> . Tiedonsiirtomuoto, jota käytetään JavaScript-ohjelmissa

LMS	<i>Learning management system.</i> Oppimisenhallintajärjestelmä
MySQL	Avoimena lähdekoodina jaettava oliopohjainen relaatiotietokantajärjestelmä
Nutilus	GNOME-työpöytäympäristön virallinen tiedostonhallintaohjelma
PDO	<i>PHP data objects.</i> Tietokantaohjelmointimalli, joka mahdollistaa erilaisten tietokantojen käyttämisen PHP-sovellusten tietojen tallentamisessa
PECL	<i>PHP Extension Community Library.</i> Tarjoaa lataushakemiston PHP:n laajennuksille
PHP	<i>Hypertext Preprocessor.</i> Verkko-ohjelmointikieli, joka suoritetaan palvelimella
phpMyAdmin	PHP-pohjainen, selaimen kautta käytettävä MySQL-tietokannan hallintatyökalu
PLS	<i>Physics labs system</i>
POST	Protokolla, jolla siirretään tietoa HTTP-kutsujen välityksellä
PostgreSQL	Avoimena lähdekoodina jaettava olio-relaatiotietokantajärjestelmä
SAML	<i>Security Assertion Markup Language.</i> XML-standardi tietojärjestelmien käyttäjien tunnistamiseen ja valtuuttamiseen liittyvien tietojen jakamiseen tietoverkossa
SCORM	<i>Sharable Content Object Reference Model.</i> Määrittelee, kuinka sisällöt tulisi pakata ja kuvata, kuinka sisältö asentuu ja toimii oppimisympäristössä ja kuinka oppija voi navigoida ja selata kurssin osia
SSH	<i>Secure Shell.</i> Salausta tukeva tietoliikenneprotokolla
WSOD	<i>White screen of death.</i> Ruudulla ei näy mitään
WYSIWYG	<i>What you see is what you get.</i> Tekstieditori, jolla jo muokattaessa sisältö näyttää samanlaiselta kuin lopputulos
XHTML	<i>eXtensible Hypertext Markup Language.</i> HTML:stä kehitetty www-sivujen merkintäkieli, joka täyttää XML:n muotovaatimukset
XML	<i>eXtensible Markup Language.</i> Merkintäkieli, jolla pystyy määrittelemään myös muita merkintäkieliä

1 Johdanto

Insinööriyön tavoitteena on toteuttaa toimeksiantajan tarpeita ja vaatimuksia vastaava oppimisjärjestelmä ja kuvata suunnittelu- ja toteutusprosessi vaiheittain insinööriyöraportissa. Projektin toimeksiantajana on Metropolia Ammattikorkeakoulu, jossa olen itsekin opiskelijana. Projektin lähtökohtana on uudistaa fysiikan laboratoriokurssien oppimisympäristö toiminnalliseksi ja tehokkaaksi vanhaan toimintamalliin verrattuna. Vanhan toimintamallin heikkouksia ovat muun muassa sen irrallinen kokonaisuus, aikataulunhallinta ja oppimismateriaalin sisältö ja jakaminen. Teoriaoppimisen ja käytännön oppimisen yhdistämisen välillä on todettu vaikeuksia muun muassa siinä, että teoriaopetus saattaa olla eri ajankohtana, jopa eri lukukaudella kuin käytännön laboratorioharjoitukset. Täten opiskelijoilla ei ole riittävän tehokkaita lähtökohtia ja työkaluja omaksuakseen oppimismateriaalia perinpohjaisesti. Kurssilla jaettava, laitteistoon liittyvä oppimismateriaali on osoittautunut vanhentuneeksi, ja paperille tulostetut työohjeet eivät ole käytännöllisiä. Projektin suunnittelussa ja toteutuksessa pyrin ottamaan kaikki nämä lähtökohdat huomioon, jotta uusi järjestelmä poistaisi entisen toimintamallin puutteet.

Laboratoriotyöt toteutetaan Metropoliaassa muun muassa Pascon toimittaman fysiikan suureiden mittaamiseen tarkoitetun testilaitteiston avulla. Pascon järjestelmään kuuluu muun muassa antureita, vahvistimia, värähtelijöitä, sähkömittauspiirilevyjä ja kannettavia datankeruuyksiköjä sekä tietokoneelle asennettava datan analysointisovellus.

Työn toteutuksen vaatimuksina ovat helppo käytettävyys jatkokäyttäjille, tuki CAS-rajapinnalle, mahdollisuus jatkokehitykseen ja tehokas käyttäjienhallinta. Ympäristön tulee olla yhteensopiva palvelimien kanssa.

Insinööriyöraportissa esitellään yleisesti sisällönhallintajärjestelmien (CMS) ominaisuuksia, toteutukseen valittavan ohjelmiston, Drupal 7:n, sisältöä ja itse työosuuden erityispiirteitä. Koska työn tuloksena syntyy järjestelmä, jota olen itse tekemässä suunnittelusta lähtien, haluan raportissani pohtia projektin sisältämää toteutusta suunnittelusta aina valmiin järjestelmän tuotannon loppuun asti.

Insinööriöraportissa esitellään aluksi sisällön- ja oppimisenhallintajärjestelmistä ja verrataan niiden välisiä hyötyjä ja haittoja yleisellä tasolla. Seuraavaksi perehdytään sisällönhallintajärjestelmien ominaisuuksiin, muun muassa perusasetuksiin, rajapintoihin, teemoihin ja muokkaamiseen, ja valitaan niistä fysiikan laboratorioharjoitusjärjestelmään sopivimmat. Tämän jälkeen selvitetään produktion kulku, eli toteutuksen erityispiirteet sisältöineen. Seuraavaksi kerrotaan projektin kohtaamista ongelmista ja niihin soveltuvista paikkauksista. Lopuksi arvioin tekemääni työtä ja pohdin, ovatko asetetut tavoitteet toteutuneet, sekä annan parannusehdotuksia.

2 Sisällönhallintajärjestelmät

2.1 Sisällönhallintajärjestelmät (CMS)

Sisällönhallintajärjestelmä (Content Management System tai CMS) on tietojärjestelmä, joka palvelee koko organisaation sisällönhallintaa sen sijaan, että se olisi keskittynyt pelkästään johonkin yksittäiseen osa-alueeseen, kuten esimerkiksi verkkopalveluiden hallintaan. (Boiko 2005: 82.)

Sisällönhallintajärjestelmän keskeisiä tehtäviä ovat muun muassa tiedon säilyttäminen, verkkosivujen, dokumenttien, mediatiedostojen ja useiden muiden eri sisältö-tyyppien hallinta. Sisällönhallintajärjestelmä voi suorittaa useita eri toimintoja verkkosivustolle. Se voi muun muassa säätää, millä tavoin sisältö näytetään ja kuinka monta kertaa se näytetään jollekin tietylle käyttäjälle, ja hallita, kuinka sisältö on yhteydessä muiden verkkosivun elementtien kanssa. Sisällönhallintajärjestelmä mahdollistaa myös verkkosivujen ylläpitämisen tekniikkaa ja koodausta osaamattomille henkilöille. (BusinessDictionary.)

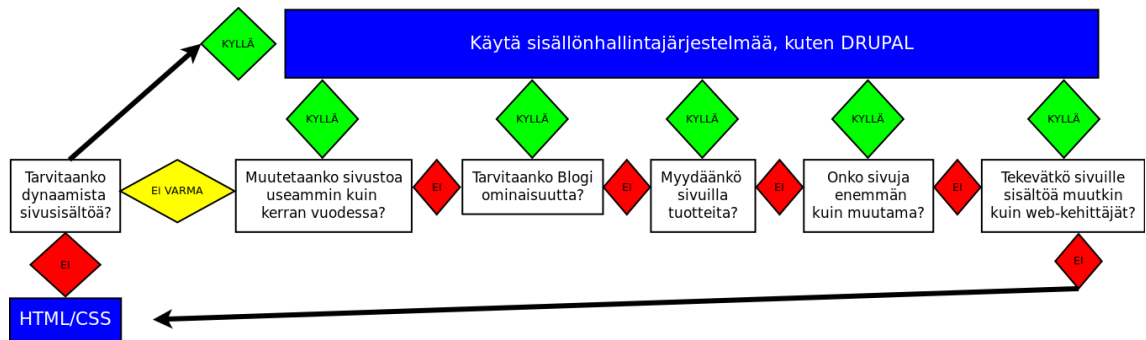
Sisällönhallintajärjestelmätyyppejä on useita, muun muassa seuraavat:

- dokumenttienhallintajärjestelmä
- WWW-sisällönhallintajärjestelmä
- julkaisujärjestelmä
- verkkokauppajärjestelmä
- aineistonhallintajärjestelmä (muun muassa kuva-aineistot, videomateriaali, multimedia).

(Tietotekniikan sanasto.)

Sisällönhallintajärjestelmä on tarpeellinen tilanteessa, jossa sivuston manuaalinen hallinta on monimutkaista ja vaatii enemmän kuin mitä käyttäjällä on ammattiosaamista. Sen hyötynä on muun muassa se, että se sisältää valmiin perustan käyttäjille ja ylläpitäjälle. Sisällönhallintajärjestelmät perustuvat usein valmiisiin moduuleihin. Moduulien ansiosta käyttäjällä on mahdollisuus valita ne ominaisuudet sivustolleen, jotka hän kokee tarpeellisiksi. Myöhemmin on mahdollista hallita sivustoa lisäämällä uusia moduuleita tai poistamalla vanhoja, omien tarpeiden mukaan. Sisällönhallintajärjestelmän

avulla voidaan siirtää rutiininomaisia töitä hallintajärjestelmän hoidettavaksi, jolloin liikenee lisää resursseja itse sisällön tuottamiseen ja ylläpitämiseen ja järjestelmän valmis automatiikka huolehtii muun muassa tietokantamuutoksista. Kuvassa 1 osoitetaan syitä, joiden takia kannattaisi valita toteutusvaihtoehdoksi juuri sisällönhallintajärjestelmä.



Kuva 1. Esimerkki syistä valita sisällönhallintajärjestelmä.

2.2 Oppimishallintajärjestelmät (LMS)

Oppimishallintajärjestelmä (learning managements systems tai LMS) on sovellus, jota käytetään koulutusohjelmien, opetuksen ja verkkokoulutuksien julkaisussa, sisällönhallinnassa, dokumentoinnissa, ylläpidossa ja raportoinnissa. On kahdenlaista tapaa tuottaa materiaalia LMS:iin, muun muassa luonti itse ohjelman sisällä ja tuonti toisella ohjelmalla tehtynä, esimerkiksi SCORM:lla (Shareable Content Object Reference Model). Sopivan LMS:n valinnassa tulee huomioida sen ominaisuudet tarpeen mukaan; monipuolinen LMS mahdollistaa yleensä muun muassa ulkoisen sisällön tuonnin.

LMS:n tulisi suoriutua vähintään seuraavista toiminnoista:

- ylläpidon keskittäminen ja automatisointi
- ylläpidon ja ohjattujen ylläpitopalveluiden tarjoaminen
- oppimisen sisällön koonti ja julkaisu nopeasti ja tehokkaasti
- koulutusta koskevien aloitteiden vahvistaminen skaalauntuvalla verkkopohjaisella tasolla
- siirrettävyyden ja standardien tukeminen
- sisällön personointi ja tiedon uudelleenkäytön mahdollistaminen.

(Ellis 2009: 1.)

Oppimisenhallintajärjestelmän perustehtävänä on hallinnoida verkkokursseja ja tarjota kouluttajalle seuranta- ja raportointityökalut. Järjestelmän avulla opettaja voi seurata jokaisen yksittäisen opiskelijan työskentelyä ja menestymistä ja välittää haluamaansa oppimismateriaalia.

Oppimisenhallintajärjestelmän hyötyjä ovat muun muassa keskustelut ja yhteisöllinen oppiminen ja käyttäjien ja ylläpitäjän vuorovaikutus sisällön kanssa. Järjestelmä antaa tukea oppimiseen liittyviin kysymyksiin ja aikataulujen hallintaan.

2.3 Sopivan hallintajärjestelmän valintakriteerit

Valintavaiheessa eri hallintajärjestelmiä verrattaessa tulee kiinnittää huomiota järjestelmän perusominaisuuksiin, kuten hallintaan, seurantaan ja raportointiin sekä siihen, onko järjestelmä helppokäyttöinen. Sen sijaan, että jokaista sivustoa kohti jouduttaisiin luomaan uusi kokonaisuus aina, kun aineistoa halutaan muuttaa, on järkevämpää hyödyntää sisällönhallintajärjestelmien tarjoamaa valmista tekniikkaa. Perinteisen LMS-järjestelmän avulla voidaan organisoida oppimista perustasolla esimerkiksi jakamalla oppimateriaalia verkossa tai järjestämällä kurssiin liittyvä ilmoittautuminen sekä tiedotus. LMS:n heikkoudeksi koituukin se, että käytännössä se on CMS:a suppeampi mahdollisuuksiltaan eikä välttämättä täytä käyttäjien tai ylläpitäjän tarpeita.

CMS:n etu on, että se sisältää valmiiksi integroidut työkalut (moduulit), joiden avulla voidaan luoda oppimateriaalia ja säilyttää sitä tietokannoissa. Siinä on vakiona muun muassa käyttäjienhallinta, sisällönhallinta, valmiita moduuleita, hyvä dokumentaatio ja lukuisia esimerkkitoteutuksia. Sen avulla voidaan myös muokata oppimateriaalin sisällöistä laadukkaita kokonaisuuksia ja toimittaa materiaali käyttäjien saataville. CMS:n heikkoutena ovat puutteelliset työkalut liittyen käyttäjien suoritusten seurantaan ja raportointiin.

Insinöörityön alkuvaiheessa päädyin siihen, että CMS on aikaa säästävä ja järkevä valinta työn toteutukseen, jotta kaikkea ei tarvitsisi luoda ja rakentaa alusta alkaen. Koska LMS on suunnattu ohjattujen opetusmateriaalien näyttämiseen ja hallintaan sekä käyttäjän taitojen osaamisen tason monitorointiin, on sen yleinen muunneltavuus heikompi kuin CMS-järjestelmissä. Valitsin CMS:n työn toteutukseen, koska valmiit LMS-järjestelmät eivät tarjonneet mielestäni tarpeeksi muunneltavuutta eikä käyttötarkoitus

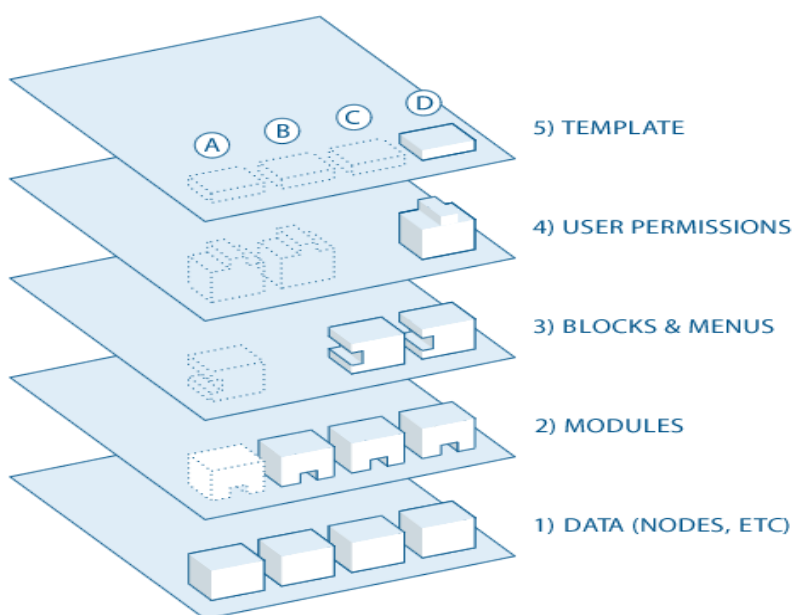
ollut tähän työhön sopiva. Joissain tapauksissa toiminnallisesti tehokkain toteutus voi olla CMS:n ja LMS:n yhdistelmä, jonka avulla voidaan luoda CMS:n ja LMS:n parhaat ominaisuudet yhdistävä kokonaisuus.

3 Drupal -sisällönhallintajärjestelmä

3.1 Yleistä Drupalista

Drupal on avoimeen lähdekoodiin perustuva sisällönhallintajärjestelmä, joka sopii hyvin sekä henkilökohtaiseen että yrityskäyttöön toimintojensa, muunneltavuutensa ja monipuolisen laajennusmoduulivalikoimansa ansiosta.

Drupalin rakenne on jaettu viiteen eri järjestelmätasoon. Alimmainen, eli ensimmäinen taso muodostuu syötetystä sisällöstä. Toinen taso koostuu moduuleista, joita ovat esimerkiksi kommentoinnit, käyttäjäprofiilit ja foorumit. Kolmas taso muodostuu lohkoista (blocks) ja menuista eli valikoista. Lohkoihin voidaan toistaa muun muassa moduulien tulosteet, ja niiden näkyminen sivustoilla on helposti säädeltävissä tulostusoikeuksien avulla. Neljännessä tasossa määritetään käyttäjäryhmien oikeudet, muun muassa mitä kukakin käyttäjä saa nähdä ja tehdä. Ylimmällä, eli viidennellä tasolla on malli (template), joka koostuu pääasiassa XHTML:stä, CSS:stä ja PHP:stä. Näiden Drupal -tasojen avulla voidaan luoda sivustolle rakenne, mutta niiden lisäksi sivusto tarvitsee myös ulko-osun, jota Drupalissa kutsutaan teemaksi. (The Drupal overview, 2010). Drupalin teemoista enemmän luvussa 3.3. Drupalin järjestelmärakenne on esitetty kuvassa 2.



Kuva 2. Drupalin järjestelmärakenne (The Drupal overview, 2010).

Drupal API (Application Programming Interface) on rajapinta, joka mahdollistaa tietojen siirron erillisten sovellusten välillä (SuomiSanakirja.fi). Drupal tarjoaa kehittäjille hyvin dokumentoidun ohjelmointirajapinnan (Drupal API), joka määrittelee, miten ohjelmafunktioita käytetään. Drupal API on joukko PHP-komentoja yhdistettynä Drupal-funktioiksi.

3.2 Asennusvaatimukset

Tässä luvussa keskitytään pelkästään Drupal 7 -version vaatimuksiin, koska työn toteutus tehtiin tällä versiolla.

PHP-palvelimeksi suositellaan Apache v.1.3 tai v.2.x kaikilla käyttöjärjestelmillä. Koska suuri osa Drupalin kehityksestä ja testauksesta toteutetaan Apache-palvelimella, juuri Apache on testatuin valinta. Drupal-ydin voidaan myös asentaa Microsoftin IIS 5-, IIS 6- tai IIS 7 -palvelimille, mutta IIS on tarkempi PHP-asetusten suhteen ja täten monimutkaisempi toteutus. Windows IIS-palvelinta käytettäessä menetetään Drupalin paikkasominaisuus (patch). (Drupal system requirements 2011.)

Tietokantapalvelimeksi suositellaan ensisijaisesti MySQL:a, mutta myös PostgreSQL tai SQLite ovat soveltuvia vaihtoehtoja. Tässä työssä käytettiin MySQL:a. Drupal 7 vaatii MySQL:n version 5.0.15 tai korkeamman sekä PDO-tietokanta laajennuksen PHP:lle. PDO on lyhenne PHP Data Objectsista, ja sen avulla voidaan päästä käsiksi tietokantoihin käyttämällä yhtenäistä API:a. PDO:n versio PECL ei ole yhteensopiva Drupal 7:n kanssa, eikä sitä täten voida käyttää. (Drupal system requirements 2011 ja What is PDO? 2010.)

Drupalin eräät ominaisuudet eivät ole tuettuina joissakin halvoissa ja ilmaisissa tietokannoissa. Toimiakseen tietokannoissa on täten oltava muun muassa seuraavat oikeudet: select, insert, update, delete, create, drop, index ja alter. Tietokantaan voidaan myös antaa käyttäjälle kaikki oikeudet, jolloin edellä mainituista ei tarvitse erikseen huolehtia. (Drupal system requirements 2011.)

Drupal 7:n PHP-versioksi vaaditaan 5.2.5 tai korkeampi, ja PHP:n vähimmäismuistimäärän (memory limit) on oltava 32 MB, suositus on kuitenkin 64 MB. Metropolian PHP tukee 128 MB:a. (Drupal system requirements 2011.)

Drupal 7 vaatii standardit PHP-laajennokset, Hashin ja JSON:n toimiakseen. Kuvakäsittelyn tukena toimii PHP-kirjasto, kuten esimerkiksi GD, jonka PHP 4.3 ja sitä uudemmat versiot sisältävät. Vaihtoehtoinen moduuli on ImageMagick, joka tukee Drupalin perustason kuvamanipulaatioita, mutta integroituu huonommin muihin moduuleihin ja tarjoaa näin ollen heikomman muunneltavuuden. (Drupal system requirements 2011.)

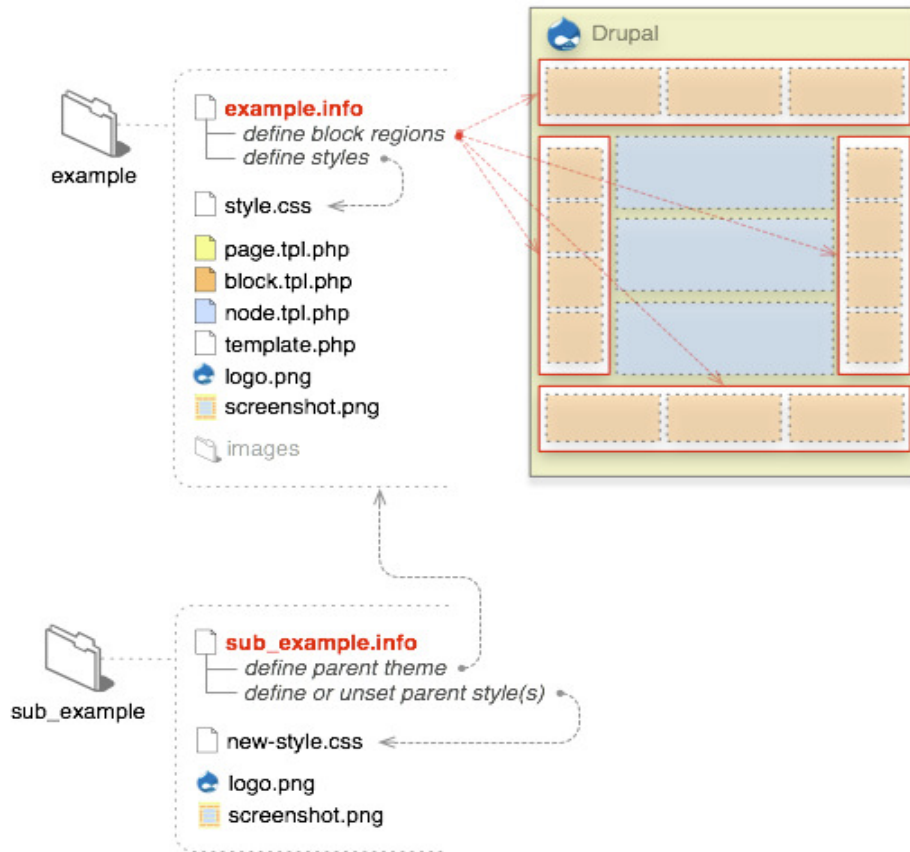
Fysiikan laboratoriotyöympäristö vaatii CAS-palvelimen. Attribuuttien käyttämisen vaatimuksena on käyttää CAS-palvelinta, joka tukee SAML VERSION_1_1 -protokollaa.

3.3 Drupalin muokkaus ja teemat

Drupal teema koostuu ytimen sivupohjista, joilla sivuston toiminta määritetään. Oletusarvoisesti näitä tiedostoja ei muuteta, vaan asetukset ja konfiguraatiot tallennetaan tietokantaan, josta komentosarjat käyvät tarkistamassa parametreja ja tulostavat niiden mukaisen käyttöliittymän. Jos sivupohjia muutetaan, kutsutaan sitä teeman muokkaukseksi. (Drupal Core templates 2011.)

Teemat ovat Drupal-moduuleita, jotka määrittävät, miten Drupal generoi HTML -sisällön (Theming Drupal 6 and 7, 2011). Teemaa muokattaessa on suositeltavaa ottaa pohjaksi valmis Drupal-teema ja muokata sitä. Ytimen mukana tulevien teemojen lisäksi on valittavana vaihtoehtoisia teemoja. Kannattaa käydä läpi aiemmin julkaistut teemat, ennen oman teeman muokkaamisen aloittamista. Sopivan teeman etsiminen kannattaa aloittaa osoitteesta <http://drupal.org/project/themes>.

Drupalin ulkoasu koostuu kolmesta eri osa-alueesta (Region), joista keskimmäinen on sisältö (Content) ja oikean- ja vasemmanpuoliset alueet ovat nimeltään lohkoja (Block). Drupalin sisällön yksikkö on solmu (Node), eli jokainen sivu, blogi tai keskustelu on yksi solmu. (VanDyk&Westgate 2007: 5-6.) Kuvassa 3 on esitetty, mistä elementeistä Drupal-teema koostuu.



Kuva 3. Drupalin teeman rakenne (Drupal Overview of theme files 2011).

Lohkot ovat sisältöön liittyvää tietoa, joka näytetään perinteisesti sivun laidassa. Lohkoja ovat esimerkiksi sisäänkirjautumislohko, keskustelut, otsikot tai kyselyt. Ylläpitäjä voi määrittää, näkyvätkö lohkot jokaisella sivulla vai vain etusivulla. Lohkot voivat lisäksi olla näkyvillä vain tietyille käyttäjille, ja niiden näkymät, kuten valikot, voivat mukautua käyttäjien perusteella. (VanDyk&Westgate 2007: 5-6.)

Teema muokataan kopioimalla valittu teema `sites/all/themes/*`-kansioon ja muokkaamalla moduulimääritteet uudelle nimelle (huom. "sisältöalueet").

Toteutuksessa on käytetty Drupal ytimen mukana tulevaa Bartik-teeman muokattua versiota. Lisäksi on huomioitu geneeristen teemojen sisältöalueet ja projektin ulkoasu, joka on helposti muutettavissa uudelle sivupohjalle. Muokatussa teemassa on vaihdettu vasemman sivupalkin leveyttä ja alaviitteen asemointia. Muut asetukset on tehty käyttöliittymän kautta, joten ne ovat tietokannassa. Sivupohjia voi muuttaa `template.php`-sivupohjasta tai `css/`-kansion tyyli tiedostoista. Toteutuksessa päädyttiin kuvan 4 mukaiseen käyttöliittymään.

The screenshot shows the 'Physics labs' website interface. At the top, there is a navigation menu with 'Home', 'Experiments', and 'Experiments Help' buttons. The main content area is titled 'Päävalikko' (Main menu) and is divided into two columns. The left column, labeled 'Vasen valikko' (Left sidebar), contains a 'Current exercise' section with a link to 'Capacitance-Electrometer', a 'Registered members' section listing two users named 'Ted Test', and an 'Add new member' button. The right column, labeled 'Sisältö' (Content), is titled 'Capacitance, electrometer' and includes a 'General description' section with two bullet points: 'Measure the capacitance of a plate capacitor using a high impedance electrometer.' and 'Estimate the impedance of the electrometer.' Below this is a link to 'Read full experiment instructions'. A 'Detailed help file' section contains a warning that using the help will decrease points by -1 and a 'Use -1 point help' button. At the bottom of the content area, there is a note: 'When you are finished with the experiment, ask for the teacher to check it:' followed by a 'For instructor only' button. The footer of the page contains the copyright information: '© Copyright, Helsinki Metropolia University of Applied Sciences | Physics labs 2011'.

Kuva 4. Fysiikan laboratorioharjoitusjärjestelmän käyttöliittymä.

Käyttöliittymän suunnittelussa haluttiin varata päävalikko staattisille linkeille ja vasen lohko dynaamisille valikoille. Vasen lohko voi sisältää useita valikoita kerralla. Oikea valikko jätettiin käyttämättä, koska se ei ollut tilankäytön kannalta tarpeen ja ylimääräinen valikko olisi monimutkaistanut käytettävyyttä.

Koodiesimerkissä 1 Drupal_add_css-funktiolle annetaan parametriksi oletusteeman asemasta käytettävät tyylimäärittelyt.

Koodiesimerkki 1. Drupal API CSS -linkki.

```
<?php
drupal_add_css('sites/all/themes/custom_css/custom_expfirst.css'
?>
```

Käytetyt, Drupal-teeman yliajavat CSS-tyylit on tallennettu sites/all/themes/custom_css kansioon.

4 Fysiikan laboratorioharjoitusjärjestelmän toteutus

4.1 Projektityöskentely etätyöpalvelimellä

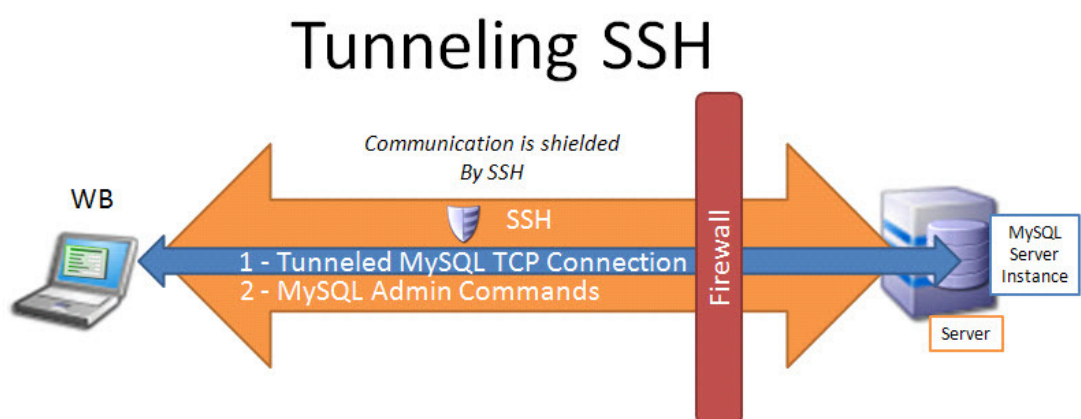
Fysiikan laboratorioharjoitusjärjestelmän toteutus tehtiin Ubuntu Linux -käyttöjärjestelmällä, ja todettiin, että Linux osoittautui erinomaiseksi työkaluksi tämän-tyyppiseen työskentelyyn ja ohjelmointiin.

SSH-tunneli ohjaa liikenteen "projects.metropolia.fi"-palvelimelta "shell.metropolia.fi"-osoitteeseen, josta se tulee SSH-tunnelin kautta työpisteelle. Työpisteellä tunneli näkyy osoitteissa localhost:1522 ja localhost:15443. Näin kaikki liikenne on SSH-salattua Metropolian verkosta työpisteelle. Koodiesimerkissä 2 SSH-tunneli SSH-porttia 22 ja MySQL-porttia 443 varten muodostetaan Linuxin komentoriviltä.

Koodiesimerkki 2. Linux Bash -komentotulkin suoritusjono SSH-tunnelin muodostamiseksi.

```
# /bin/sh
sudo ssh -f user@shell.metropolia.fi -L 1522:projects.metropolia.fi:22 -N
sudo ssh -f user@shell.metropolia.fi -L 15443:projects.metropolia.fi:443 -N
```

Kuva 5 esittää periaatekaaviona, kuinka MySQL-tietokantaa hallinnoidaan SSH-tunnelin läpi.



Kuva 5. SSH-tunneli tietokantapalvelimelle (MySQL Workbench 5.2 and SSH Mini-FAQ 2009).

Projektikansiosta tehtiin pikakuvakkeet Nautilus-tiedostonhallintaan SSH-verkkolevynä, jolloin ne olivat paikallisesti käytössä, samoin kuin koneeseen liitetyt kiinteät mediat. Tämä nopeutti huomattavasti ohjelmointia verrattuna tilanteeseen, jossa muutokset täytyisi ladata palvelimelle erillisellä SSH-ohjelmalla.

Ohjelmointieditorina käytettiin yksinkertaista oletustekstieditoria geditia, joka tunnistaa PHP-koodin ja osaa värittää tekstin sen mukaiseksi. Geditin avulla saatiin myös rivinumeroit näkymään ja sarkaimen leveydet määriteltä. Nämä ominaisuudet osoittautuvat hyödyllisiksi. Kuvat käsiteltiin Gimp-kuvanmuokkausohjelmalla. Vektorigrafiikka editoitiin Inkscapeilla, josta materiaali vietiin pdf:ksi. Selaimena käytettiin Firefox-versiota 3, joka päivitettiin versioksi 4, joka julkaistiin kesken projektin. Yhteensopivuustarkistuksia varten sivua testattiin Windowsin iExplorerilla VirtualBox-virtuaalikoneen kautta. Vian selvitykseen käytettiin Firefoxin lisäosaa Firebug. Kaikki työkalut olivat ilmaisia ja ladatavissa suoraan pakettienhallinnasta, lukuun ottamatta virtuaalikoneen Windows XP:a.

MySQL-tietokannan editointiin käytettiin phpMyAdmin-verkkopohjaista MySQL-käyttöliittymää. PhpMyAdminista sai vietyä tietokannan varmuuskopiot eri muodoissa, joista suositeltavin oli pakattu muoto, sillä se kutistaa tiedostokoon kymmenesosaan.

Projektikansion varmuuskopiointi toteutettiin manuaalisesti, koodiesimerkin 3 kuvaamalla Linux Bash-komennoilla.

Koodiesimerkki 3. Varmuuskopiointi Linux-komentotulkissa.

```
tar -cvf folder
gzip tarredFile
cp tarredFile tarredFileBackup
```

Työpisteen varmuuskopiot tehtiin differentiaalisesti rsync-komennolla. (Riikonen 2009: 12.)

4.2 Tietokanta

Tietokannaksi valittiin riippumattomien taulujen toteutus, koska taulun jakamisesta relaatioliitoksiin ei laskettu olevan etua.

Valittua tietokantarakennetta verrattiin vaihtoehtoon, jossa laboratorioharjoituksen nimikenttätietue olisi tallennettu omaan primääriavaintauluun, jolloin toiseen tauluun jäävä osuus olisi koostunut yksinomaan tapahtumakirjanpidosta. Tämän rakenteen etu on, että siinä saadaan säilytettyä harjoituksen nimen eheys tapahtumataulussa ja helpotettua tietokantahakua, esimerkiksi tapauksessa, jossa halutaan listata kaikki harjoitukset. Harjoituksen nimen eheyden säilyttäminen tapahtumataulussa olisi vaatinut, että opiskelijasivulla harjoitus olisi yksilöity, esimerkiksi ID-tunnisteella. Tämä puolestaan olisi aiheuttanut sen, että harjoituksia olisi hallittava tietokannassa.

Yksi tämän työn lähtökohdista oli ohjelmakoodiosion eristäminen ylläpitäjän hallintarajapinnasta. Relaatiotietokannan käyttäminen olisi vaatinut, että harjoitussivu sisältäisi harjoituksen ID-tunnisteen. Tunniste tosin voitaisiin piilottaa laittamalla sivulle erillinen "block", joka sisältää ohjelmakoodin, esimerkiksi Computed field –moduulilla. (Drupal Computed field 2006.)

Tunnistetta varten voitaisiin rakentaa käyttöliittymä ylläpitäjälle, koska uutta harjoitussivua tehtäessä sivu olisi kuitenkin tullut yhdistää harjoituksen nimeen. Sopiva ratkaisu tähän olisi admin-sivulla tai harjoitussivulla, vieraalla näkymättömällä osiolla oleva erillinen harjoitustenhallintakäyttöliittymä. Oppilaalle ladattavassa (ei välttämättä tulostuvassa) sisällössä puolestaan tulisi näkyä oikean harjoituksen ID-tunniste. Tämä voisi tosin aiheuttaa helposti sekaannusta, ja sellaisessa tapauksessa, jossa ylläpitäjälle tapahtuisi ylläpitovirhe, harjoituksen rekisteröinti ei välttämättä toimisi.

Ylimääräinen taulu lisäisi tietokantaan kirjoitettavan ylimääräisen tiedon määrää ja monimutkaistaisi tarpeettomasti rakennetta. Relaatiokannan käyttäminen myös muuttaisi tietokantahakujen rakennetta. Koodiesimerkki 4 edustaa tyypillistä hakua relaatiokannasta, jossa experiments.id- ja tapahtumakirjanpito.oper_id-taulujen välillä vallitsee viiteavainrelaatioyhteys.

Koodiesimerkki 4. Esimerkki staattisesta kyselystä relaatiotietokannasta.

```
<?php
SELECT `firstname`, `surname` FROM tapahtumakirjanpito, experiments
WHERE experiments.id = tapahtumakirjanpito.oper_id
AND experiments.name = "exp2";
?>
```

Relaatioiden käyttö ei automatisoi tietokannan ja sitä käyttävän asiakkaan rajapinnan toimintaa. Relaation tehtävä on vain yksilöidä vanhempi tauluun tallennettava taulu ja huolehtia taulujen välisestä viite-eheydestä. (Ratol.fi: Avaimet.)

Valitussa ratkaisussa tietokantaan kirjattavan harjoituksen nimi haetaan sivun URL:n aliaksen nimestä (Drupal pathalias -moduuli), jonka merkkijonosta on poistettu sivun osoite. Tämän polun nimi muodostetaan sisällön nimestä, autoalias-moduulin tekemästä Internet-osoitteen kanssa yhteensopivasta muunnoksesta. Seuraavassa koodissa juurikansion URL-polku haetaan PHP \$_SERVER-funktiolla. Siitä poistetaan Drupalin oletussisältöön viittaava osuus, jolloin saadaan pelkkä autoalias-moduulille määritelty polun nimi, eli projektin nimi. Koodiesimerkissä 5 Internet-osoitepolku tallennetaan suoritusjonolla.

Koodiesimerkki 5. Projektin nimen tallentamiseen URL-osoitteesta käytetty suoritusjono.

```
<?php
$linkA = $_SERVER['REQUEST_URI'];
$pageadd=str_replace("/?q=content/expfirst-","", $linkA);
?>
```

Ongelmana tässä on se, että osoiteriviltä haettavan tiedon nimien tulee olla Internet-osoitteen kanssa yhteensopivia (ei välejä tai erikoismerkkejä). Käyttäjille näytetään polun nimestä tallennettu harjoituksen nimi, joka ei ole yhtä luettava kuin tietokannasta haettu, erikseen tallennettu nimi.

Drupal ytimessä on Path-moduuli, jota hallitaan moduulivalikosta. Pathauton etu on, että se on hallittavissa samalta sivulta kuin sivun sisältö. Jos sivustolta haluttaisiin tallentaa eri nimi, kuin minkä sivun URL:n alias antaa, se ei olisi suoraan mahdollista.

Tämä periaatteessa mahdollistaisi osoiteriviltä syötetyn GET-parametrilla siirretyn osoitteen, joka puolestaan mahdollistaisi MySQL -injektion syötön tietokantaan tai ainakin väärin harjoitusten pääsyn tietokantaan. Tämä on kuitenkin estetty tarkistamalla, että käytetty osoite on validi ja että se löytyy myös Drupalin tietokantaan syötetystä sivustolistasta.

Työssä lähdettiin siitä, että harjoituksille ei tarvittaisi erillistä hallintaa ja ettei ole sellaista mahdollisuutta, etteivät tulokset tallentuisi tietokantaan eikä uusille harjoitus sivuille tarvitsisi laittaa yhtään ohjelmakoodia.

Drupal database API

Työssä käytettiin Drupal database -rajapintaa. PHP:ssä ohjelmoija määrittelee kaikki asetukset itse tai tekee niille asetustiedostot, joita kutsutaan. Drupal database -rajapinnan käyttöä varten ei tarvitse määritellä erikseen sisältösivulle, koska ne on määritelty jo Drupalin asetuksissa, mutta siihenkin on tarvittaessa mahdollisuus. Rajapinnan funktioiden parametrit syötetään taulukkomuodossa. Drupal API helpottaa muun muassa seuraavien ominaisuuksien käyttöä verrattuna tavalliseen PHP:hen:

- tukee useampia tietokantapalvelimia helposti
- mahdollistaa kehittäjille tuen monimutkaisempien toimintojen kehittämiseen, kuten transaktioihin
- tarjoaa ohjatun käyttöliittymän kyselyjen dynaamiselle luomiselle
- valvoo turvatarkastuksia ja muita hyviä käytäntöjä
- tarjoaa puhtaan käyttöliittymän moduuleita sivuston kyselyiden keskeyttämiseen ja muokkaukseen.

(Drupal Database API 2011.)

Oletuksena PHP:ssä tietokantayhteys luodaan määrittelemällä kaikki muuttujat erikseen, kuten koodiesimerkissä 6.

Koodiesimerkki 6. Esimerkki staattisesta PHP–MySQL-tietokantahausta.

```
<?php
$host = 'localhost';
$user = 'root';
$pass = 'password';
$database = 'database';
$connect = @mysql_connect ($host, $user, $pass);
$conn = mysql_connect($host, $user, $pass) or die
mysql_select_db($database);
$result = mysql_query('SELECT * from table');
mysql_close($conn);?>
```

Drupal API yhdistää sarjan PHP-funktioita yhdeksi Drupal-funktioksi ja sama haku Drupalissa on koodiesimerkin 7 kaltainen.

Koodiesimerkki 7. Esimerkki Drupal–MySQL-tietokantahausta.

```
<?php
$table = 'table';
$result = db_query("SELECT d, title FROM {$table}");
?>
```

Olenaisinta on, että parametrit on eritelty erilliseksi taulukoksi, joka mahdollistaa esimerkiksi useamman tietokannan päivityksen samoilla arvoilla.

Drupalissa on myös tietokantahakujen lisäksi tietokantarakenteen manipulointiin tarkoitettuja funktioita. Seuraavasta osoitteesta löytyy Drupalin tietokantamallin muokkausrajapinnan dokumentaatio:

(<http://api.drupal.org/api/drupal/includes--database.inc/group/schemaapi/6>).

4.3 Moduulit

Drupal-moduulit ovat erillisiksi paketeiksi koostettuja funktioita. Drupal sisältää moduulienhallinnan, jonka kautta voidaan ottaa tai poistaa moduuleita käytöstä ja hallita niiden asetuksia ja käyttöoikeuksia.

Moduuli voi sisältää muuhun rajapintaan integroituja ominaisuuksia, kuten generiset moduulit, jotka ovat ladattavissa Drupalin sivuilta (Drupal Modules) (<http://drupal.org/project/Modules>). Kirjoitushetkellä valmiita Drupal 7:lle tarkoitettuja moduuleita oli 1 383 kpl, ja määrä lisääntyy koko ajan. Moduulien versionumeroissa esimerkiksi versio 7.x.1.1 tarkoittaa versiota 1.1 Drupal ydin 7.x:lle.

Työssä hyödynnettiin seuraavia Drupal ytimeen kuulumattomia moduuleita:

- Content Access
- Administration
- CCK
- CAS
- Chaos tool suite
- Development
- Example modules
- Computed Field
- Block reference
- IMCE
- Pathauto
- CKEditor
- IMCE Wysiwyg API bridge
- WYSIWYG
- Views
- Webform.

Paketit löytyvät Drupal-juurikansion alta osoitteesta: "sites/all/modules/". Lisäksi ytimen moduuleista tulee erikseen kytkeä päälle ainakin PHP filter, jotta sisällön tekstikentän tyyppiä voitiin valita PHP-suoritus.

Drupaliin lisätystä toiminnoista pyrittiin tekemään moduuleita, ja sivustoa varten tehtiin kolme moduulia:

- adduserform (käyttäjänlisäys)
- databwriter (tietokantatoimintoja)
- experiment admin (ylläpitäjäsivuston toiminta).

Moduulit asennetaan sites/all/modules/moduulinimikansioihin, minkä jälkeen moduuli näkyy Drupalin moduulihallinnassa. Moduuli koostuu seuraavista tiedostoista:

- moduulinimi.info (Drupalin moduulihallinnassa näkyvät tiedot)
- moduulinimi.module (funktiot).

Uuden moduulin funktiot ovat käytettävissä heti, kun se on otettu Drupalin moduulihallinnasta käyttöön ja sille on annettu riittävät oikeudet. Moduulit nopeuttavat ohjelmointia huomattavasti, koska testaamiseen ei tarvita muuta kuin tallennus ja selaimen uudelleenlataus.

Osassa moduuleista on erillinen ohjelmakirjasto, esimerkkinä CKEditor (WYSIWYG-editori), jonka asiakaskoneella ajettava Javascript-pohjainen Open Source -käyttöliittymä on erillisessä kansiossa. CKEditorin tuotesivu löytyy osoitteesta <http://ckeditor.com>. Drupal-moduuli puolestaan on palvelimella ajettava rajapinta, joka mahdollistaa editorin käytön Drupalissa. (CKEditor - WYSIWYG HTML editor 2009.)

Esimerkkien tapauksessa Drupal 7:ään saa "examples"-moduulin, jossa kaikki esimerkit on suunnattu versiolle 7 ja ovat ajettavissa suoraan Drupal hook_menu -rajapinnan kautta. Koodiesimerkissä 8 on Drupal examples form-moduuli, joka hyödyntää hook_menu -rajapintaa.

Koodiesimerkki 8. Esimerkki Drupal hook -menun toiminnasta.

```
<?php
$item['examples/form_example/tutorial/10'] = array(
  'title' => '#10',
  'page callback' => 'drupal_get_form',
  'page arguments' => array('form_example_tutorial_10'),
  'access callback' => TRUE,
  'description' => t('Tutorial 11: Form with file upload'),
  'type' => MENU_LOCAL_TASK,
  'file' => 'form_example_tutorial.inc',
);
?>
```

Tätä form_example_menu()-funktion \$items-muuttujaa (lomake) kutsutaan selaimen osoiteriviltä ~/?q=examples/form_example/tutorial/10, joka palauttaa lomakkeen. Erilistä lomakkeen palautusta ei tarvita.

4.4 Käyttäjänhallinta, käyttäjien kirjautuminen ja tietoturva

Fysiikan laboratorioharjoitusjärjestelmä asetti useita vaatimuksia käyttäjänhallinnalle.

Päävaatimukset olivat seuraavat:

1. Metropolian CAS-rekisteritunnuksen omaavan henkilön tulee kyetä kirjautumaan todennetuksi käyttäjäksi automaattisesti.
2. Ryhmän jäsenten tulee kyetä kirjautumaan harjoitukseen, johon liitetään koneeseen kirjautujan CAS-tunnus.
3. Järjestelmä ei saa sekoittaa Metropolian CAS-kirjautumista, joka on pakollinen selaamisen avauksen yhteydessä. Tästä on huolehdittu muun muassa siten, että olemassa voi olla vain yksi CAS-haltija kussakin harjoituksessa eikä käyttäjä voi kirjautua CAS-järjestelmästä ulos PLS:n (Physics Labs Systemin "kirjaa termi kaikkiin kohtiin") kautta.
4. Ryhmän jäsen ei voi kirjautua kahta kertaa samaan harjoitukseen. Koska CAS-nimi tulee tietokannasta, ei CAS-käyttäjä voi avata kirjautumissivua uudestaan, ellei istunnossa ole tarkistustietoa voimassa siitä, että suoritus on aloitettu kolmen tunnin sisällä (session voimassaoloaika).
5. Ryhmän jäsenten syötössä tarkastetaan että etu- ja sukunimi on syötetty ja että tarkistus on hoidettu Drupalin lomake-API-tietueella "'#required' => TRUE,".

Lomakkeessa, joka sisältää ryhmän jäsenten lisäyksen, on SQL-injektioriskiä vähennetty rajoittamalla käyttäjänimikenttien enimmäispituutta. Suurempiin tarkistuksiin ei ole koettu olevan tarvetta otettaessa huomioon käyttäjäkohderyhmän.

CAS-käyttäjät kirjautuvat automaattisesti todennetuiksi käyttäjiksi, minkä perusteella on rajoitettu, ettei ilman tunnusta pääse kirjautumaan harjoitukseen eikä kirjoittamaan tietokantaan. Kirjautuneelle CAS-käyttäjälle voi asettaa ylläpitäjän roolin. Silloin valtuutetulla selaimella CAS-järjestelmään kirjautunut käyttäjä pääsee suoraan sivustolle ylläpitäjän roolissa, erikseen kirjautumatta.

Huomioitavaa on, että jos Drupal-käyttäjänä on jo saman tunnuksen omaava käyttäjä, kuin minkä CAS-järjestelmä palauttaa, tulee virheilmoitus. Täten CAS-järjestelmään kuulumattomien käyttäjien nimet tulee valita sellaisiksi, ettei riskiä samannimisen CAS-käyttäjän kirjautumiseen ole.

Käyttäjäoikeudettomilta suljetut sivut on suojattu käyttämällä sivujen muodostukseen sessiotunnistetta. Ennen sivun PHP-HTML-generointia tarkastetaan, että käyttäjällä on tarvittava rooli. Näin ollen sisältöä ei pääse väärinkäyttämään, vaikka joku tietäisikin sisällön osoitteen.

Valikoiden näkyvyyttä on hallittu käyttämällä lohkojen "roles" -> "Show block for specific roles" -määritettä ja Näytä lohko "Pages on which this PHP code returns TRUE (experts only)". (Show block on specific pages 2008). Koodiesimerkkiä 9 on käytetty Experiments menu -lohkon näkyvyysasetuksiin. Olennaista on, että näkyvyysetuusrvon määrittäminen on jaettu erilliseen funktioon.

Koodiesimerkki 9. Lohkon näkyvyysasetus.

```
<?php
// "Pages on which this PHP code returns TRUE (experts only)"
$match_b = "false";
$match_b = adduserform_getexperiment();
return $match_b;
?>
```

Koodiesimerkissä 10 lohkon kutsumafunktio palauttaa totuusarvon. Funktiossa on määritelty sallitut arvot tietokentässä, joita verrataan Drupal -rajapinnan "arg"-funktion palauttamaan sivutyypin arvo -polun osaan. Jos sivu kuuluu määriteltyihin sivutyyppeihin, funktio palauttaa arvon "TRUE".

Koodiesimerkki 10. Lohkon näkyvyysehdon kutsuma `adduserform_getexperiment()`-funktio.

```
<?php
// Prints "Experiment-menu only if nodetype is correct && experiminet is not yet
selected.
function adduserform_getexperiment() {
  // Only show if $match is true
  $match = false;
  // Which node types
  $types = array('experiments_index', 'experiment_front_page', 'register-group');
  // Match current node type with array of types
  if (arg(0) == 'node' && is_numeric(arg(1))) {
    $nid = arg(1);
    $node = node_load($nid);
    $type = $node->type;
    if (isset($_SESSION['experimentnlock'])) {
    }else{
    $match |= in_array($type, $types);
    }
    }elseif ($match == "0"){
      $match = "FALSE";
    }else{
      $match = "TRUE";
    }
  }
  return $match;
}
?>
```

CAS (Central Authenticate Server)

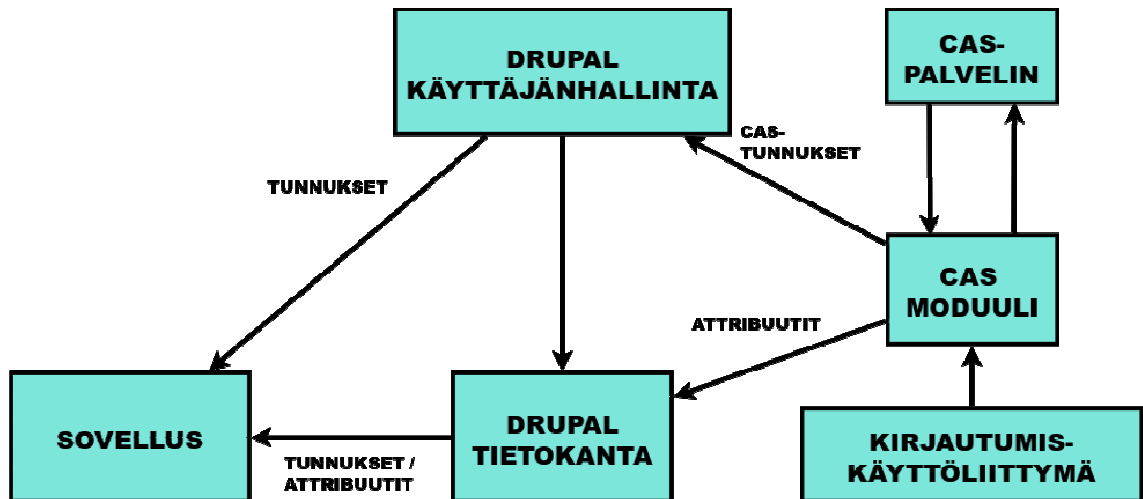
Tässä työssä CAS (Central Authenticate Server) -palvelua on käytetty opiskelijatietojen hakemiseen Metropolian CAS-palvelimelta. Alkuvaiheessa ammattikorkeakoulun opiskelijat joutuvat kirjautumaan CAS-palvelimelle, kun he aukaisevat selaimen. Tätä rekisteröitymistietoa on hyödynnetty siten, että kun CAS:iin kirjautunut opiskelija aloittaa työnsä, hänen nimensä lisätään automaattisesti laboratorioharjoituksen ryhmän jäseneksi. Lisäksi muut kirjautuneet ryhmän jäsenet ovat tietokannassa yhdistettyinä CAS:iin kirjautuneeseen opiskelijaan, koska CAS:iin voi olla kerrallaan kirjautuneena vain yksi opiskelija, jolloin ryhmän lisäjäsenet joudutaan pyytämään kirjautumaan erikseen. Tästä on se hyöty, että sivujen käyttäjistä ja ryhmän jäsenistä jää aina varmistettu yksilötunniste ja täten sivujen väärinkäyttömahdollisuudet pienenevät.

Metropolian CAS-palvelimen kanssa kommunikointiin on käytetty asiakasrajapintana Drupalin CAS-moduulia (CAS 2006) ja phpCAS-kirjastoa (phpCAS 2011). Kun moduuli on asennettu, tunnistaa Drupal CAS-järjestelmään kirjautuneen käyttäjän ja kykenee lisäämään sen Drupal-käyttäjäksi. Näin voidaan tarvittaessa esimerkiksi rajoittaa sivujen käyttöä ulkopuolisilta, koska CAS-kirjautuminen vaaditaan vasta, kun käyttäjä aloittaa harjoituksen suorittamisen.

Oletuksena Drupalin CAS-moduuli palauttaa vain CAS-käyttäjätunnuksen, koska etu- ja sukunimi on haettu erikseen CAS-attribuuteilla (ks. Vähimmäisvaatimukset). Metropolian projektipalvelin palauttaa tarvittaessa opiskelijanumeron, mutta tarpeettomana sitä tietoa ei tallenneta. Metropolian palvelimella tiedot ovat attribuuteissa "fullName" ja "ownWinhaStudentRoleID". "fullName"-attribuutti palautetaan yhtenäisenä merkkijonona, mutta tietokantaan ne on tallentamista varten jaettu etu- ja sukunimeen. Attribuuttien haku on lisätty suoraan CAS-moduulin `cas_login_check()`-funktioon.

Toteutuksessa on käytetty erillistä tietokantataulua, johon tallennetaan jokaisen kirjautuneen CAS-käyttäjän nimi ja käyttäjätunnus. Näin nimi päästään hakemaan missä vaiheessa tahansa käyttäjätunnuksen perusteella. Toinen vaihtoehto olisi ollut lisätä se CAS-moduulin `cas_user_operations_create_username($uids)`-funktioon, jolloin se olisi saatu tallennettua vaihtoehtoisesti Drupalin users-tauluun. Suuremmilla muutoksilla se olisi voitu saada integroitua CAS-käyttöliittymään.

Attribuuttien haku tulee suorittaa siinä vaiheessa, kun Drupalin CAS-moduuli suorittaa kirjautumisen, sillä se sulkee kirjautumisen jälkeen yhteyden, jolloin attribuutteja ei enää päästä hakemaan. Kuvassa 6 on esitetty fysiikan laboratorioharjoitusjärjestelmässä käytettävän CAS-järjestelmän toimintaperiaate.



Kuva 6. CAS-järjestelmän toimintaperiaate.

Koodiesimerkin 11 koodi hakee CAS "fullName"-attribuutin ja CAS-käyttäjätunnukset sekä kirjoittaa ne tietokantaan samalle riville.

Koodiesimerkin 11. CAS-attribuuttien tallentamiseen käytetty ohjelmakoodi.

```
<?php
// cas.module:n cas_login_check() -funktio
// Get CAS attributes
foreach (phpCAS::getAttributes() as $key => $value) {

    if ($key == 'fullName'){
        $cas_full = $value;
    }
    // if ($key == 'ownWinhaStudentRoleID'){
    // $student_id = $value; }
}

if (isset($cas_full)){

    $cas_useri = phpCAS::getUser();

    $table = "aaa_cas_names";
    db_insert($table)
        ->fields(array(
            'cas_full' => $cas_full,
            'cas_name' => $cas_useri
        ))->execute(); //Drupal execute
}
?>
```

cas.module-tiedoston muutoksista on mahdollista tehdä paikkaustiedosto, joka voitaisiin asentaa helposti, jos CAS-moduulia joudutaan päivittämään. Tällä hetkellä muutokset joudutaan kuitenkin lisäämään käsin.

Linkitettyt tiedostot

Laboratorioharjoituksen käyttöliittymässä tuli olla latausmahdollisuus käyttäjille avoimille tiedostoille ja tiedostoille, joiden avaamisesta kirjataan suorituspisteitä vähentäviä miinuspisteitä. Ylläpitäjällä tuli olla lisäksi mahdollisuus päivittää tiedostoja. Avoimille ja suljetuille tiedostoille tehtiin sisältöalueelle oma tiedostokenttä (Drupal ytimen Field-moduuli), jotta tiedostojen vaihto olisi ylläpitäjälle helppoa samalta sivulta, kuin mistä sivun sisältöä muokataan. (Working with field module (Drupal 7 and later) 2010.)


Kuva 7 osoittaa harjoitussivutyypissä käytettyjen kenttien tyypit.

Label	Name	Field	Widget	Operations	
+ Title	title	Node module element			
+ Body	body	Long text and summary	Text area with a summary	edit	delete
+ File	field_file	File	File	edit	delete
+ Closed file	field_filec	File	File	edit	delete

Kuva 7. Sivutyypin kentät.

Kuvan 8 mukaiset kaksi file-kentän tyyppiä näyttävät seuraavanlaisilta, kun suljetun kentän lisäasetuksista on poistettu ylimääräiset muokkausasetukset.

File

File information	Operations
<p>+  Experiment_Capacitance.doc (25 KB)</p> <p>Description</p> <p>Read full experiment instructions</p> <p><small>The description may be used as the label of the link to the file.</small></p>	<p>Remove</p>

Add a new file

Files must be less than **8 MB**.
Allowed file types: **pdf doc docx xls jpg gif exe xlsx txt**.

Closed file

 [Experiment_Capacitance-Electrometer_close.pdf](#) (117.9 KB)

Kuva 8. Harjoitussivun tiedostonhallinta.

Suljettujen tiedostojen hakemiston tuli olla suojattu, jotta käyttäjä ei pääsisi avaamaan tiedostoa suoraan osoiteriviltä. Suojaus toteutettiin Drupalin Private-kansioilla (Private local files served by Drupal), joiden tiedostot ovat ladattavissa vain Drupalin API-linkkimuunnoksen kautta, jolloin vain järjestelmällä on niihin lukuoikeus. Tiedoston sijainti suojattiin ja miinuspuoleisten vaatima tiedoston avaustapahtuman rekisteröinti toteutettiin taulukolla, jonka POST-lähetä suorittaa Drupal API drupal_goto-funktion ja kirjoittaa tapahtuman tietokantaan.

Suojausta suunniteltiin aluksi myös Apache-palvelimen .htaccess RewriteCond -määreellä, jolla sivuston liikenne voitaisiin ohjata kokonaan toiseen osoitteeseen, mutta tämä ei olisi suojannut kansiota. Suojaus olisi voitu tehdä oikeuksia käyttäen, mutta ylläpitäjän oikeuksien puute esti tämän. (Module mod_rewrite.)

Koodiesimerkin 12 sisältö .htaccess-tiedostossa uudelleenohjaa HTTP-pyyntöt uuteen osoitteeseen. Koodi konvertoi käyttäjälle monimutkaisemmat "get_product.php?id=" -osuudet helppolukuisempaan "products"-muotoon.

Koodiesimerkki 11. Esimerkki Apachen osoitteen uudelleenohjauksesta.

```
RewriteEngine on RewriteBase /
RewriteRule ^products/([0-9]+)/?$ /get_product.php?id=$1 [L]
```

Tiedostolatauksen aloituksen olisi voinut toteuttaa myös PHP header -funktiolla, jonka suoritus lähettää käyttäjälle http-otsikon. Tässä tapauksessa se sisältäisi tiedoston latauslinkin ja tiedostotyyppin määrittelyn. (PHP header 2011.)

4.5 Varmuuskopiointi

Drupal-sisällön suositeltava varmuuskopiointi sisältää muutokset, jotka on tehty muokkaamalla tiedostoja tai lataamalla uusia tiedostoja hakemistoihin. Nämä tiedostot sisältävät hakemistot ovat yleensä /drupal/sites/, jonka alla ovat kaikki puhtaaseen asennukseen lisätyt moduulit, teemat ja kirjastot, asetukset sekä sivuille ladatut tiedostot. Lisäksi tarvitaan /.htaccess-tiedosto turvallisuusasetuksia varten sekä kolme ensimmäistä riviä /index.php-tiedostosta, jos halutaan ottaa virheenjäljitys käyttöön. Muut tiedostot voidaan korvata suoraan esimerkiksi puhtaalla Drupal 7:n asennuksella. Nämä

tiedostot eivät yleensä muutu itsekseen, ja virheellisen asetuksen pääsee aina korjaamaan muokkaamalla tiedostoa, joten peruslevyrikon varalle tehtävää tiedostojen varmuuskopiointia ei välttämättä tarvitse suorittaa. Varmuuskopiointi on kuitenkin suositeltavaa, koska työmäärään suhteutettuna varmuuskopioinnin vaatima tila on häviävän pieni ja edullinen.

Sivuston kautta tehdyt muutokset ja sivun sisältö ovat tallennettuina tietokantaan. Tietokannan varmuuskopiointi on tärkeää, koska kaikki ylläpitäjän hallintasivujen kautta tehtyjen muutosten sisältämä informaatio sijaitsee siellä. Jos sivuille syötetään sellaista ohjelmakoodia, joka estää sivuston toiminnan täysin, voidaan sivusto palauttaa yksinkertaisesti palauttamalla tietokanta. On suositeltavaa, että tällaisessa tilanteessa käydään korjaamassa tietokantaeditorilla toiminnan estävä vika vain siihen tietokannan kenttään, jossa vika on.

5 Keskeiset ongelmakohdat, virheenjäljitys ja paikkaus

5.1 Ongelmakohdat

Insinööriyön teossa esiintyneitä ongelmia kartoittaessa keskityttiin Drupalin ominaisuuksiin ja työn toteutukseen liittyviin ongelma-kohtiin sekä niiden paikkausmahdollisuuksiin.

Projektin alussa työskentelyä hidastutti Drupalin monipuolisuus ja Drupal API:n eroavaisuudet PHP-syntaksiin verrattuna. Projektin edetessä kävi kuitenkin selväksi, että Drupalin kehityksen korkean alkukynnyksen jälkeen sen ohjelmistokehitysrajapinta ja sisällönhallintatyökalut toimivat tehokkaasti. Missään vaiheessa ei tullut tilannetta, että Drupal ei soveltuisi joiltakin osin tämän työn tarkoitukseen.

Drupal-versioksi valittiin versio 7, joka oli julkaistu vasta muutama kuukausi aikaisemmin. Drupal 7:n dokumentaatio ei ollut vielä kovin kattava, mutta se kehittyi nopeasti. Tällä hetkellä Drupalin dokumentaatio koskee versioita 5.x, 6.x ja 7.x, jotka eivät ole kaikilta osin yhteensopivia. Varsinkin versioon 7 on tullut huomattavia muutoksia rajapinnan toimintaan.

Tässä työssä oleelliset eroavaisuudet koskivat tietokanta- ja taulukko-API:n toimintaa. Drupal.org:n dokumentaatiossa on eroteltu hyvin eri Drupal-versioita koskevat eroavaisuudet. Koska Drupal 7 on uusi julkaisu, esimerkkitoimitusten löytämiseksi on usein haettava tietoa myös muista lähteistä. Suurin osa erittelemättömistä dokumenteista koskee Drupalin versioita 6.x, koska niiden kirjoittamisen aikaan Drupal versiota 7 ei ollut vielä julkaistu.

Suuren osan PHP-osaamisesta joutui opettelemaan uusiksi Drupal API:n myötä. Tavallinen PHP toimi, mutta pidemmän päälle oli kuitenkin parempi opetella Drupalin omat toimintatavat.

Väärä merkki PHP-sivusivustollisissa voi aiheuttaa virheilmoituksen tai saattaa estää koko Drupalin käynnistymisen. Tätä tilaa kutsutaan WSOD:ksi, joka on lyhenne lauseesta (White Screen Of Death). Tällöin ainoa keino palauttaa sivusto toimintakykyiseksi on

poistaa virheellinen arvo tietokannasta. Virheellinen sisältö löytyy "block"-taulusta. Virheellisen taulun voi myös palauttaa varmuuskopiosta. Suositeltava ohjelmointitapa onkin käyttää moduuleita, koska virheellistä moduulia voi muokata ulkoisella tekstieditorilla. WSOD:n voi aiheuttaa myös PHP-palvelimen sivuston käyttöön varatun muistin loppuminen. (The White Screen of Death 2010.)

WYSIWYG-editorin käyttöönotossa oli joitakin ongelmia. WYSIWYG:n kehitys aikaisemmista versioista oli keskittynyt vain yhteen WYSIWYG-moduuliin, eli CKEditoriin. Ohjeissa sanottiin CKEditorin vaativan Wysiwyg API:n, mutta se ei silti toiminut Wysiwyg API:n kanssa. Ongelma ratkesi lopulta ottamalla Wysiwyg API pois käytöstä. Toimimalla näin menetettiin Drupalin WYSIWYG-editorin hallintaan liittyviä ominaisuuksia, mutta CKEditor toimi. Myöhemmin kokeilin päivittynyttä Wysiwyg API:a, joka toimikin CKEditorin kanssa.

Työtä hidasti palvelimen asetusten epäyhteensopivuus ja pääkäyttäjän oikeuksien puute. Ongelmia oli ensin hakemisto-oikeuksien kanssa, joille piti lisätä Apache-palvelimen pääkäyttäjätunnuksen suoritusoikeus phpMyAdmin -käyttöliittymää varten. Lisäksi Apachen asetukset olivat myöhemmin muuttuneet palvelimen ylläpidon seurauksena, mikä esti phpMyAdmin-toiminnan. Tapauksessa jouduttiin kääntymään ylläpidon puoleen.

Lisäongelmia ilmeni siinä, että PHP:lla oli pääkäyttäjän oikeudet kansioihin ja Drupal muutti käyttäjätiedostojärjestelmän kansioden oikeuksia, jotta se pystyisi säätelemään niiden oikeuksia käyttäjähallintatasolla. Kansioita jouduttiin myöhemmin siirtämään, ja siinä jouduttiin turvautumaan uudestaan pääkäyttäjän oikeuksien haltijaan. Asiaankuumattomia kansioita muuttui Apache-tunnuksen omistamaksi. Ylläpidon ansiosta Apachen konfiguraatioon tehtiin Drupalin toimintaan vaikuttavia muutoksia, minkä jälkeen vain käyttäjätiedostojärjestelmän kansiot olivat Apache-tunnuksen omistuksessa.

Tiedosto-oikeuksia yritettiin muuttaa seuraavalla koodiesimerkin 13 PHP:lla ajettavalla UNIX-komennolla. Se ei kuitenkaan toiminut.

Koodiesimerkki 12. Yritys muuttaa tiedosto-oikeuksia PHP-suorituksella.

```
<?php
`chmod -R a+w drupal/sites/default/files`;
?>
```

CAS-rajapinnan attribuuttituki palvelinpuolella valmistui vasta, kun käyttäjähallinta oli jo osittain valmis. Mahdollinen attribuuttituen lisääminen oli kuitenkin huomioitu suunniteltaessa, joten suuremmilta muutoksilta vältyttiin.

Aikaulu osoittautui tiukaksi. Osa ylläpitäjäsivuston toteutuksesta jouduttiin siirtämään myöhemmäksi.

Drupal-virheenjäljitys asetetaan päälle laittamalla koodiesimerkin 14 mukainen sisältö juurihakemiston index.php-tiedoston alkuun.

Koodiesimerkki 13. Drupal-virheenjäljityksen päällekytkeminen.

```
<?php
error_reporting(E_ALL);
ini_set('display_errors', TRUE);
ini_set('display_startup_errors', TRUE);
?>
```

5.2 Paikkaukset

Drupal-paikkaukset ovat käyttäjien tekemiä paikkauksia Drupal ytimeen ja sen moduuleihin. Paikkaukset ovat aina tapauskohtaisia, eikä niitä tule asentaa, ellei siihen ole erityistä syytä. Paikkaukset löytyvät aina foorumeilta. Drupalilla on julkaisujärjestelmä paikkauksille. Drupalin virallisilla sivuilla ei julkaista paikkauksia, ennen kuin ne on validoitu. (Applying patches 2011.)

Drupal-paikkaus käyttää Unix-järjestelmän komentotulkin "patch" -komentoa ja sen "diff"-funktioita. Drupal patch -tiedosto sisältää paikkauksen kohdepolut, ja siksi se tulee asentaa Drupalin juurihakemistoon. Esimerkkikuvassa 9 on esitetty Ckeditorille ke-

hitettävät paikkaukset. Myös julkaisuun vielä hyväksymättömiä paikkauksia voi kokeilla, mutta niiden toimintaa ei ole kuitenkaan testattu riittävästi.

Summary	Status	Priority	Category	Version	Component	Replies	Last updated	Assigned to
CKeditor + Profile2 = Call to undefined function ckeditor_user_customize_form_validate() on uid=1	needs work	normal	bug reports	7.x-1.x-dev	Code	8	2 hours 57 min	
Switching text format to filtered html deletes all "p" tags	needs work	normal	bug reports	7.x-1.x-dev	Code	7	12 weeks 6 days	

Kuva 9. Drupal-virheenkorjausjärjestelmä.

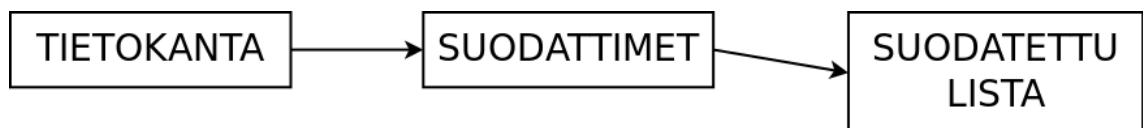
6 Drupal admin -käyttöliittymä

Harjoitusjärjestelmän ylläpitäjän tapahtumamerkintöjen muokkaussivun tekeminen aloitettiin insinöörityössä, mutta se ei ole vielä valmis. Sivulle on tehty oma sivun sisältyyppi, jonka linkki on määritelty näkymään vain ylläpitäjän näkymässä. Sivun lisäksi suojattu myös tarkistamalla, onko käyttäjä ylläpitäjä. Suojaus on tehty koodiesimerkin 15 lauseella, jossa Drupal-funktio on yhdistetty ehtolauseeseen.

Koodiesimerkki 14. Ylläpitäjäoikeuksien tarkistaminen Drupal-funktiolla.

```
<?
php if(user_access('administer')){ }
?>
```

Sivua suunniteltaessa oli otettava huomioon, että tapahtumamerkintöjä saattaa kirjautua tietokantaan vuodessa jopa 1 500 kpl. Jo muutamassa kuukaudessa kertyvä määrä on liian iso tulostettavaksi yhdelle sivulle, joten tuloksia tulee suodattaa. Kuvassa 10 on kuvattu suunniteltujen suodattimien toiminta.



Kuva 10. Admin-suodattimien toimintaperiaate.

Suodatusparametreiksi valittiin

- suorituksen hyväksyminen
- suorituksen ajanjakso
- harjoituksen nimi.

Kuvassa 11 on esitetty visuaalisesti Admin -käyttöliittymän suodatinvallinnat.

Experiment list

Approved state <input checked="" type="radio"/> Unapproved <input type="radio"/> Approved <i>dropdown_approved</i>	Select year 1 day <i>dropdown_year</i>	5 hours Capacitance-Electrometer <i>dropdown_experiment</i>
--	---	--

Suodatetut tulokset

Kuva 11. Admin -käyttöliittymän suodatinvallinnat.

Suotimet ovat valintalomakkeita. Ilman erillistä käsittelijää lomake täytyy päivittää. Se tapahtuu oletuksena käyttäjän painaessa "submit"-painiketta. Ongelmia syntyy tapauksessa, jossa käyttäjä ei paina nappia. Tällöin suotimien vaikutus ei ole validi. Tässä toteutuksessa tämän toteutuminen on varmistettu käyttämällä Ajax (Ajax (Asynchronous JavaScript And XML) -käsittelijää, joka päivittää tapahtumakirjanpidon aina, kun käyttäjä muuttaa lomakkeen arvoa. Drupal tarjoaa valmiin rajapinnan Ajaxille, joten sen käyttöönotto oli suhteellisen helppoa. (Ajax framework.)

Koodiesimerkissä 16 on esimerkki Drupal-lomakkeesta, jossa otsikko päivittyy Ajaxin takaisinkutsulla.

Koodiesimerkki 15. Ajax-takaisinkutsun käyttäminen Drupal-lomakkeessa.

```
<?php
// Tarkistaa onko lomakkeella jo arvo ja asettaa sen oletusarvoksi
$selected = isset($form_state['values']['ajax_selector']['dropdown_experiment']) ?
$form_state['values']['ajax_selector']['dropdown_experiment'] : key($options_year);
$form['ajax_selector']['dropdown_experiment'] = array(
  '#prefix' => '<td valign=top>',
  '#suffix' => '</td>',
  '#type' => 'select',
  // Dynaaminen otsikko
  '#title' => '<div id="dropdown-table-replace">' . $options_year[$selected] . '</div>',
  '#title' => $options_year[$selected],
  '#default_value' => $selected,
  '#description' => t('dropdown_experiment'),
  // Taulukon arvojen haku funktiosta
  '#options' => _expadmin_get_experiment_dropdown_options(),
  '#options' => $options_two,
  '#default_value' => $options_two[5],
  '#ajax' => array(
    // Ajax palautusfunktion nimi
    'callback' => 'expadmin_ajax_selector_callback',
    // Päivittyvä DOM kenttä voidaan asettaa myös takaisinkutsufunktiossa
    'wrapper' => 'dropdown-table-replace',
  ),
);
?>
```

Tarkoituksena on toteuttaa myös tapahtumamerkintöjen päivitys Ajaxia käyttäen. Näin sivulla ei tarvitse olla "submit"-painiketta. Silloin näkyvissä olevan sivun tapahtumamerkintöjen tila vastaa aina tietokannassa olevaa tilaa eikä käyttäjän toimista johtuvaa differentiaalisuutta voi esiintyä. Tietokanta päivitetään takaisinkutsufunktiossa. DOM-rakenteen "wrapper"-tunnisteen määrittäminen voi sijaita takaisinkutsu-funktiossa ja itse ID missä tahansa osassa sivua, ei pelkästään lomakkeessa.

7 Yhteenveto

Insinööriraportin päätarkoitus oli esitellä, miksi fysiikan laboratorioharjoitusten oppimisjärjestelmän toteutuksessa päädyttiin käyttämään sisällönhallintajärjestelmää ja miten sen ominaisuuksia hyödynnettiin.

Etsiessäni projektin vaatimukset täyttäviä ja siihen parhaiten soveltuvia sisällönhallintajärjestelmiä, tutustuin useisiin eri toteutusvaihtoehtoihin, joista parhaaksi valikoitui Drupal. Päädyin Drupaliin erityisesti sen muokattavuuden, hyvän käyttäjähallinnan, monipuolisten valmiiden moduulien ja CAS-tuen perusteella. Drupalin valintaa puolsi myös Drupalin vapaa muunneltavuus ja rajoittamaton käyttö kaupalliseen tarkoitukseen. Insinööriyössä kehitettiin Metropolian fysiikan laboratorioharjoitusten suoritukseen soveltuva järjestelmä, jota käytetään kurssin tukena.

Suunnittelemani ja toteuttamani järjestelmä on osoittautunut hyödylliseksi Metropolian fysiikan laboratoriokursseille. Työssä lähdettiin liikkeelle siitä, että haluttiin uudistaa opintojakson toimintamalli toimivammaksi kokonaisuudeksi, ja mielestäni tämä onnistui niin hyvin, kuin tämän aikataulun puitteissa on mahdollista. Uusi järjestelmä on mahdollistanut tehokkaamman tavan hallinnoida kurssin oppimismateriaalien sisältöä, jakamista ja aikataulujen hallintaa, ja toivon mukaan se on lisännyt myös opiskelijoiden oppimismotivaatiota. Käyttäjien rekisteröity kirjautumisominaisuus on helpottanut opettajan kirjanpitoa, mikä oli myös yksi tekijä miksi, vanhaa toimintamallia haluttiin muuttaa.

Paranneltavaakin toteutukseen vielä jäi. Harjoituksen valvojan salasanan hyväksymisen voisi toteuttaa paremmalla tavalla kuin nykyinen salasanan kirjoitus harjoituspäätteellä. Hyväksymisen voisi tehdä esimerkiksi kauko-ohjaimella, jona voisi toimia esimerkiksi puhelimeen ohjelmoitu sovellus. Tiedonsiirto tapahtuisi bluetooth-protokollalla. Esimerkki merkkijonon lähetyksestä puhelinsovelluksella löytyy osoitteesta <http://linuxgazette.net/153/ziemann.html>. Puhelimiin löytyy myös valmiita, näppäimistökarttaa emuloivia sovelluksia. Kaukosäätimenä voisi käyttää erillistä pienikokoista bluetooth-näppäimistöä, jolloin erillistä sovellusta ei tarvittaisi. Bluetoothin käyttö tiedonsiirtoon vaatisi työpisteiltä bluetooth-tuen. Harjoitukseen kirjautumista voisi rajoittaa sallimalla sen vain kurssille ilmoittautuneille. Tämä vaatisi kurssin ilmoittautumistietojen attribuuttituen CAS-palvelimelta. PLS:ään voitaisi lisätä ominaisuuksia, jotka tu-

kevat PLS:n integrointia Pascon laiteympäristön ja sen Datalogger-ohjelmiston kanssa. Harjoitussivulle voisi lisätä esimerkiksi normaalijakauman tarkistuksen mittaustuloksista. Harjoituksen tekijät ja valvoja voisivat käyttää tietoa arvioidessaan mittausten onnistumista ennen harjoituksen hyväksymistä. Laboratorio-ohjeisiin voitaisiin lisätä videoita Pascon laitteiden käytöstä testitilanteissa ja harjoitusten kulusta. Videoiden julkaisuun Drupal-sivuilla voisi käyttää esimerkiksi Drupal MediaFront -moduulia. Sivuston kohdekäyttäjiltä saatu palaute on arvokasta arvioitaessa sivuston toimivuutta ja kehitystarvetta. Koska testaus tehdään kohderyhmässä ja sivujen luonnollisessa käyttöympäristössä, on testauksen luotettavuusaste usein parempi kuin keinotekoisesti järjestetyissä testaus tilanteissa. Suositeltava palautteen keräämismenetelmä on lomake, jonka lähete tallennetaan tietokantaan ja jonka tulokset ovat sivuston kehittäjien luettavissa.

Toteutustapana sisällönhallintajärjestelmät helpottavat verkkosivustojen hallintaa ja vähentävät ohjelmakoodin muuttamisen tarvetta sivustoa päivitettäessä. Drupal oli valintana ideaalinen, sillä sen moduulirakenne mahdollistaa myös tulevaisuudessa sen helpon muokattavuuden. Kehitystyönä projekti oli hyvin opettava myös henkilökohtaisesti: Drupal oli minulle aiemmin tuntematon sisällönhallintajärjestelmä, ja nyt minulla on huomattavasti enemmän tuntemusta Drupalista ja sen ominaisuuksista, ja voin hyödyntää sitä tulevilla työtehtävilläni.

Insinööriyön tuloksia voi käyttää Drupalin mahdollisuuksiin ja toimintaan tutustumiseen ja fysiikan laboratoriojärjestelmän jatkokehitykseen.

Lähteet

- Ajax framework. Verkkodokumentti. Drupal Association.
<<http://api.drupal.org/api/drupal/includes--ajax.inc/group/ajax/7>>. Luettu 12.4.2011.
- Applying patches. 2011. Verkkodokumentti. Drupal Association.
<<http://drupal.org/patch/apply>>. 19.4.2011. Luettu 26.4.2011.
- Boiko, B. 2005. Content management bible. United States : Wiley Publishing Inc.
- BusinessDirectory. Verkkodokumentti. WebFinance, Inc.
<<http://www.businessdictionary.com/definition/content-management-system-CMS.html>>. Luettu 11.3.2011.
- CAS. 2006. Verkkodokumentti. Drupal Association. <<http://drupal.org/project/cas>>. 16.8.2006. Luettu 20.3.2011.
- CKEditor - WYSIWYG HTML editor. 2009. Verkkodokumentti. Drupal Association.
<<http://drupal.org/project/ckeditor>>. 16.10.2009. Luettu 14.3.2011.
- Drupal Computed field. 2006. Verkkodokumentti. Drupal Association.
<http://drupal.org/project/computed_field>. 21.7.2006. Luettu 8.4.2011.
- Drupal Core templates. 2011. Verkkodokumentti. Drupal Association.
<<http://drupal.org/node/190815>>. 17.3.2011. Luettu 28.3.2011.
- Drupal Database API. 2011. Verkkodokumentti. Drupal Association.
<<http://drupal.org/node/310069>>. 4.2.2011. Luettu 20.3.2011.
- Drupal Modules. Verkkodokumentti. Drupal Association.
<<http://drupal.org/project/Modules>>. Luettu 12.3.2011.
- Drupal Overview of theme files. 2011. Verkkodokumentti. Drupal Association.
<<http://drupal.org/node/171194>>. 2.4.2011. Luettu 14.4.2011.
- Drupal system requirements. 2011. Verkkodokumentti. Drupal Association.
<<http://drupal.org/requirements>>. 24.4.2011. Luettu 26.4.2011.
- Ellis, Ryann K. 2009. Field Guide to Learning Management Systems. American Society for Training & Development.
- Frank, Mike. 2009. MySQL Workbench 5.2 and SSH Mini-FAQ. Verkkodokumentti.
<<http://mikefrank.wordpress.com/2009/12/14/mysql-workbench-5-2-and-ssh-mini-faq/>>. 14.12.2009. Luettu 14.4.2011.
- Module mod_rewrite. Verkkodokumentti. The Apache Software Foundation.
<http://httpd.apache.org/docs/1.3/mod/mod_rewrite.html>. Luettu 23.3.2011.
- phpCAS. 2011. Verkkodokumentti. Jasig.
<<https://wiki.jasig.org/display/CASC/phpCAS>>. 26.3.2011. Luettu 2.4.2011.

PHP header. 2011. Verkkodokumentti. The PHP Group.
All rights reserved. <<http://php.net/manual/en/function.header.php>>. 22.4.2011. Luettu 26.4.2011.

Ratol.fi: Avaimet. Verkkodokumentti. Ratol.fi.
<<http://www.ratol.fi/opensource/mysql/avaimet.htm>>. Luettu 22.3.2011.

Riikonen, Juha. 2009. Tietojärjestelmän varmuuskopiointin suunnittelu, testaus ja toteutus. Satakunnan ammattikorkeakoulu.

Show block on specific pages. 2008. Verkkodokumentti. Drupal Association.
<<http://drupal.org/node/320795>>. 13.10.2008. Luettu 25.3.2011.

SuomiSanakirja.fi. Verkkodokumentti. SuomiSanakirja.fi.
<<http://suomisanakirja.fi/rajapinta>>. Luettu 23.3.2011.

The Drupal overview. 2010. Verkkodokumentti. Drupal Association.
<<http://drupal.org/getting-started/before/overview>>. 21.8.2010. Luettu 13.3.2011.

Theming Drupal 6 and 7. 2011. Verkkodokumentti. Drupal Association.
<<http://drupal.org/theme-guide/6-7>>. 20.3.2011. Luettu 28.3.2011.

The White Screen of Death. 2010. Verkkodokumentti. Drupal Association.
<<http://drupal.org/node/158043>>. 22.9.2010. Luettu 18.3.2011.

Tietotekniikan sanasto. Verkkodokumentti. krimaka.net
<<http://www.krimaka.net/tietotekniikka/tietokone-ja-muut/tietotekniikan-sanasto.html>>. Luettu 11.3.2011.

Van Dyk & Westgate 2007. Pro Drupal Development. United States: Apress.

What is PDO? 2011. Verkkodokumentti. Drupal Association.
<<http://drupal.org/requirements/pdo>>. 3.11.2011. Luettu 28.3.2011.

Working with field module (Drupal 7 and later). 2010. Verkkodokumentti. Drupal Association. <<http://drupal.org/documentation/modules/field>>. 25.10.2010. Luettu 23.3.2011.

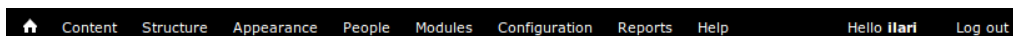
Liitteet

Liite 1. Ylläpitäjän käyttöohje

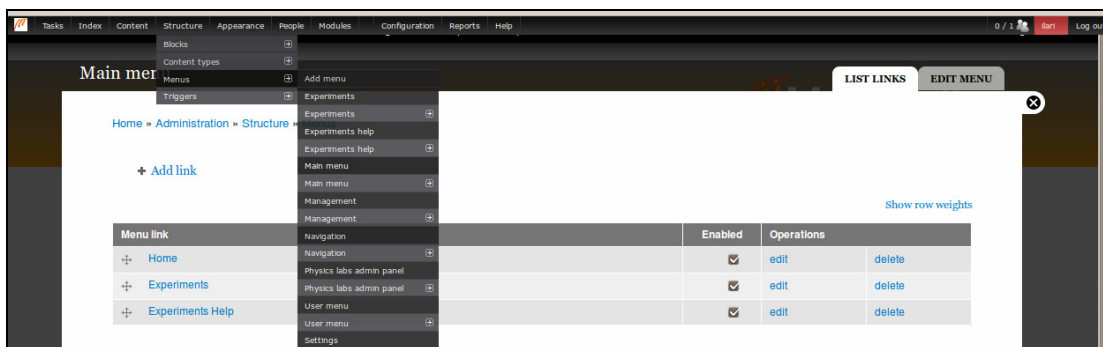


Ylläpitäjän valikoiden käyttö

Ylläpitäjän valikot koostuvat kahdesta valikosta, Drupalin oletusvalikosta ja Administration menu-moduulin lisävalikosta. Drupalin oletusvalikko koostuu vain pikakuvakkeista, Administration-menu sisältää alavalikoita.



Kuva 1. Drupal-oletusvalikko.



Kuva 2. Administration menun lisävalikko.

Uuden harjoituksen luonti

Uutta sisältöä lisätään ylläpitäjän palkista "Content -> Add content".

Erityyppisille sivuille on määritelty omat sivutyypit kuvan 3 mukaisesti.

Add content

[First page content](#)
Home page content. You can make several versions of these, and publish them when necessary.

[Experiments index](#)
Experiments index page

[Experiment page](#)
When creating new experiment, use this.

[Experiments general help index](#)
[Experiments general help](#)
When new general help content page is needed, use this.

Kuva 3. Sisältötyypit.

Harjoitussivun sivutyyppi on "Experiment page".

Uutta harjoitussivua luotaessa on kiinnitettävä huomiota siihen, että autoaliasin moduulin (harjoituksen muokkaussivulla näkyy "URL path settings" välilehtenä) generoima oikopolkumuunnos kuvaa harjoituksen nimeä. Sen "URL alias" -kentän arvon loppuosa tallennetaan tietokantaan, jossa se toimii harjoituksen tunnisteena. Tämä arvo näkyy käyttäjille kunkin harjoituksen sivulla. Kentän arvon on oltava muotoa content/expfirst-Capacitance-Electrometer.


Oletuksena kentän arvo määräytyy sivun otsikon perusteella, joka usein sopii sellaiseen, mutta on syytä tarkistaa. Kuva 4 osoittaa "URL-alias" -välilehden asetukset.

Capacitance, electrometer	<input type="checkbox"/> Automatic alias
Revision information No revision	An alias will be generated for you. If you wish to create your own alias below, uncheck this option. To control the format of the generated aliases, see the URL alias patterns .
URL path settings Alias: content/expfirst-Capacitance-Electrometer	URL alias <input type="text" value="content/expfirst-Capacitance-Electrometer"/>
Authoring information	Optionally specify an alternative URL by which this content can be accessed. For example, type "about" when writing an about page. Use a relative path and don't add a trailing slash or the URL alias won't work.

Kuva 4. Harjoituksen tunniste.

Uuden harjoituksen luonnissa tätä "URL alias" -osoitetta tarvitaan tallennettaessa suljettua tiedostoa harjoitussivulle.

Suljetun tiedoston nimen tulee olla muotoa Capacitance-Electrometer.pdf, jossa Capacitance-Electrometer vaihtuu harjoituksen mukaan. Tiedostoformaatti on PDF, ja muiden tiedostopäätteiden lataus palvelimelle on estetty. Jos nimi ei vastaa harjoituksen "URL alias" -kenttää, ei tiedoston avaus toimi. Tällöin kuitenkin avaustapahtuma kirjataan miinuspuolelle. Kuva 5 esittää suljetun tiedoston hallinnan käyttöliittymää.

Closed file
 Capacitance-Electrometer.pdf (117.9 KB) <input type="button" value="Remove"/>

Kuva 5. Suljetun tiedoston nimi.

Avoimen tiedoston nimessä ei ole rajoituksia.

Tiedostoliitteiden hallinta

Kuva 6 on tiedostoliitteiden hallintasivusta.

The screenshot shows a file management interface with a table of files. The table has columns for 'File Information', 'Display', and 'Operations'. The files listed are 'Normal Distribution.pdf (43.15 KB)', 'Measurement error.pdf (35.21 KB)', and 'Confidence interval.pdf (43.36 KB)'. Each file has a 'Description' field and a 'Remove' button. The 'Display' column contains checkboxes. Annotations in orange boxes point to various elements: 'Painoarvon hallinta, järjestystä muutetaan raahaamalla' points to the '+' icon; 'Tiedoston nimi' points to the filename; 'Määrittelee näytetäänkö linkki sivulla' points to the checkbox; 'Käyttäjille näkyvä tiedoston' points to the description field; and 'Palvelimen asetukset määrittelevät tiedoston suurimman koon. Tiedostotyyppiä voi lisätä Sivutyypin tiedostokenttäase-' points to the file size and type restrictions.

File Information	Display	Operations
<input type="checkbox"/> Normal Distribution.pdf (43.15 KB) Description: Normal distribution	<input checked="" type="checkbox"/>	Remove
<input type="checkbox"/> Measurement error.pdf (35.21 KB) Description: Measurement error	<input checked="" type="checkbox"/>	Remove
<input type="checkbox"/> Confidence interval.pdf (43.36 KB) Description: Confidence intervall	<input checked="" type="checkbox"/>	Remove

Add a new file

Selaa... Upload

Files must be less than **8 MB**.
Allowed file types: **pdf doc docx xls**.

Kuva 6. Tiedostojen hallinta.

Tiedostomuutosten jälkeen on painettava sivun alaosasta "Submit"-painiketta, jotta muutokset tulevat voimaan.

Microsoftin iExplorer 8:lla ei tiedostojen poisto onnistu, eikä juuri julkaistua iExplorer 9:ä ole testattu. Luotettavimmin sivusto toimii Firefox-versioilla 3 ja 4, joiden osalta se on myös parhaiten testattu.

Sisältöversioiden käyttö

Sivua muutettaessa voi käyttää Drupalin versiohistoriaa, joka pitää kirjaa sisältöversioista. Muutoksen kirjaus versiohistoriaan tehdään valitsemalla "Create new revision".

Kuva 7 esittää sisällön versionhallinnan tila-asetussivun käyttöliittymän.

<p>Menu settings Capacitance, electrometer</p> <p>Revision information New revision</p> <p>URL path settings Alias: content/expfirst-Capacitance-Electrometer</p> <p>Authoring information By ilariw on 2011-02-27</p>	<p><input checked="" type="checkbox"/> Create new revision</p> <p>Revision log message</p> <p>Resistor #2 value changed.</p> <p><small>Provide an explanation of the changes you are making. This will help other authors understand your motivations.</small></p>
--	---

Kuva 7. Sisältöversiohistorian käyttöönotto.

Tehdyt versiot löytyvät sen jälkeen sivun ylläpito näkymän kuvan 8 kuvaamalta Revisions-välilehdeltä.

Revisions for <i>Capacitance, electrometer</i>	
View	Edit
Revisions	Track
Revision	Operations
04/10/2011 - 20:54 by ilari Resistor #2 value changed.	current revision
04/09/2011 - 20:23 by ilari	revert delete

Kuva 8. Sisältöversiohistoria.

Sisällön julkaisutila

Sivusisällön voi laatia valmiiksi, mutta vasta kun sivusisältö on asetettu "julkaistu"-tilaan, on sivusto käytössä. Siten samasta sivusta voi pitää useampaa versiota ja julkaista niitä tarpeen mukaan.

Kuva 9 esittää sivusisällön julkaisutilan käyttöliittymää.

Menu settings Capacitance, electrometer	<input checked="" type="checkbox"/> Published
Revision information No revision	<input type="checkbox"/> Promoted to front page
URL path settings Alias: content/expfirst-Capacitance-Electrometer	<input type="checkbox"/> Sticky at top of lists
Authoring information By ilariw on 2011-02-27 00:29:47 +0200	
Publishing options Published	

Kuva 9. Julkaisutila.

Käyttäjien ja salasanojen hallinta

Uusien käyttäjien lisäys ja olemassa olevien käyttäjien muokkaussivu löytyy "Index -> People". Tähän on tarvetta käytännössä vain ylläpitäjien osalta. Drupal luo automaattisesti käyttäjien (opiskelijat) tunnukset CAS-palvelimen palauttamien tietojen perusteella.

Harjoituksen valvojan harjoitusten hyväksymissalasanana löytyy tietokannasta "drupal_aaa_teacherp"-taulusta. Hyväksymissalasanana säännöllinen muuttaminen on suositeltavaa, koska harjoituksen valvoja joutuu kirjoittamaan salasanan harjoitusten tekijöiden ollessa läsnä. Ylläpitäjän käyttöliittymään on suunniteltu tehtäväksi harjoituksen ylläpitäjän salasanavaihdon mahdollistava lomake.

Moduulien päivitys

Moduulien päivitys on syytä tehdä, jos moduuleissa esiintyy ohjelmavirheitä tai on syytä epäillä tietoturvariskiä.

Projektipalvelin ei tue päivitystä käyttöliittymästä PHP:n ftp-tuen puutteen vuoksi. Manuaalisessa päivityksessä on suositeltavaa siirtää vanha moduuli pois uuden alta, jolloin vanhaan moduuliin voidaan palata, jos päivityksessä versiossa ilmenee ongelmia.

Tällä hetkellä ainakin CKEditor sisältää ohjelmavirheitä, joten sen päivitysjulkaisuja kannattaa seurata. CKEditorin ohjelmavirheet eivät kuitenkaan ole tyypiltään käyttöä estäviä.

Moduulien päivitysten tilan näkee osoitteesta:

<http://fysiikka.edu.metropolia.fi/?q=admin/reports/updates>

Lohkojen ja valikoiden sijaintien muuttaminen

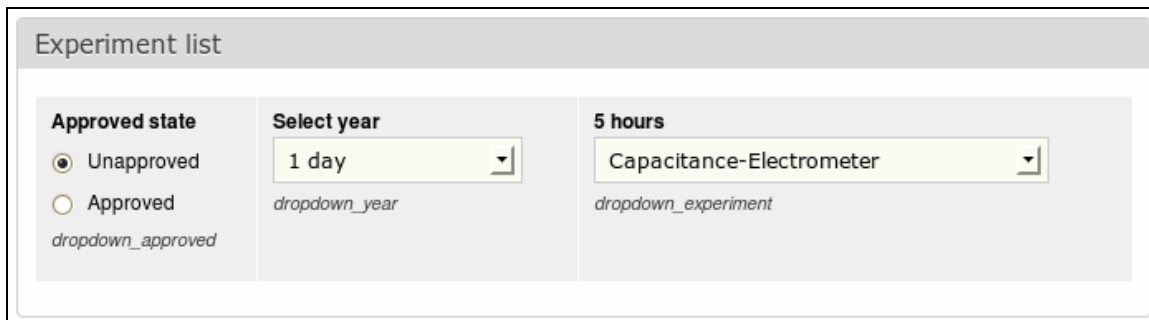
Lohkojen ja valikoiden sijaintia ja asetuksia hallitaan Home » Administration » Structure » Blocks-asetussivulta. Niiden painoarvoa voidaan muuttaa, ja niiden sijaintia teeman määrittämässä sisältökentässä voidaan vaihtaa kuvan 11 osoittamasta käyttöliittymästä.

Content				
+	Main page content	Content	configure	
+	Start experiment button	Content	configure	delete
Sidebar first				
+	Physics labs admin panel	Sidebar first	configure	
+	Experiments	Sidebar first	configure	
+	Experiments help	Sidebar first	configure	
+	Exercise group	Sidebar first	configure	delete

Kuva 10. Lohkojen hallinta.

Harjoitusten ylläpitäjän käyttöliittymä

Ylläpitäjän sivulla on suodatinlista, joka näyttää suodattimien alla listan harjoituksista, jotka täyttävät suodatinlistan määrittelemät arvot.



The screenshot shows a web interface titled "Experiment list". It contains three filter sections:

- Approved state:** Two radio buttons are present. The first is labeled "Unapproved" and is selected. The second is labeled "Approved". Below these is the text "dropdown_approved".
- Select year:** A dropdown menu is shown with "1 day" selected. Below it is the text "dropdown_year".
- 5 hours:** A dropdown menu is shown with "Capacitance-Electrometer" selected. Below it is the text "dropdown_experiment".

Kuva 11. Harjoitusten ylläpitäjän suodatinlista.

Ulkopuolisten pääsyä sivustolle voidaan rajoittaa muokkaamalla Apachen asetuksia tai ottamalla kommenttimerkki pois juurikansion .htaccess-tiedoston "#allow from .metropolia.fi"-riviltä.