



**LAUREA**  
AMMATTIKORKEAKOULU  
*Yhdessä enemmän*

# Seniori365.fi-palvelun lokalisaatio

Hiltunen, Markus  
Paavilainen, Sami

2016 Laurea



Laurea-ammattikorkeakoulu

## Seniori365.fi-palvelun lokalisaatio

Markus Hiltunen, Sami Paavilainen  
Tietojenkäsittelyn koulutusohjelma  
Opinnäytetyö  
Joulukuu, 2016

Markus Hiltunen, Sami Paavilainen

### Seniori365.fi-palvelun lokalisaatio

Vuosi 2016 Sivumäärä 33

---

Tämän toiminnallisen opinnäytetyön tavoite on mahdollistaa Seniori365.fi-palvelun toiminnan laajentuminen muille paikkakunnille. Ensimmäisenä toteutettavana paikkakuntalaajennuksena on Vantaa. Toinen keskeinen tavoite on varmistaa sivuston helppokäyttöinen jatkokehitys luomalla dokumentaatio testausympäristön perustamisesta ja luomalla ohjeet uusien paikkakuntien lisäämiselle.

Työn tilaajana on Laurean aluevastaava Pia Kiviharju. Seniori365.fi-palvelu on verkkosivusto, jonka loppukäyttäjiä ovat seniori-ikäiset ihmiset ja heille palveluita, sekä tuotteita tarjoavat yritykset. Seniori365.fi-palvelun tavoite on auttaa seniori-ikäisiä ja heidän omaisia tai avustajia löytämään heille suunnattuja palveluita, tuotteita ja tapahtumia. Sivusto on luotu espoolaisille käyttäjille, joten se sisältää alunperin vain Espoossa toimivia yrityksiä ja espoolaisille suunnattua sisältöä. Seniori365.fi-palvelulle on ollut kysyntää Vantaalla, joten sivuston laajentaminen tukemaan myös vantaalaisia yrityksiä on ajankohtaista. Samalla luodaan pohja sivuston laajentamiselle. Käyttäjälle tämä näkyy mahdollisuutena etsiä sivustolta sisältöä kuntakohtaisesti.

Sivuston kuntalaajennus perustuu jo aikaisemmin sivustolle tehtyihin kehityssuunnitelmiin ja Laurean liiketalouden opiskelijoista koostuvan suunnittelutiimin asettamiin vaatimuksiin. Päätös testausympäristön perustamisohjeistuksen tekemiseen osaksi opinnäytetyötä tuli projektin suunnitteluvaiheessa, kun huomattiin Seniori365.fi-palvelun teknisessä dokumentaatiossa olevan parannettavaa.

Sivuston tärkein arvioija on asiakas. Projektissa tärkeimmät arvot ovat sivuston käytettävyys ja selkeys käyttäjälle. Lopputuloksen on oltava helppo käyttää, kevyt teknisesti ja helposti laajennettavissa.

Listamme myös projektin aikaina sivustolla ilmenneet osiot, joita voisi parantaa. Lisäksi annamme ehdotuksia, kuinka sivustoa voisi jatkokehittää. Tässä opinnäytetyöraportissa esitään myös kehitystyön aikana havaittuja puutteita, jotka saattavat haitata jatkokehitystä. Tarkoituksena on varmistaa sivuston kehityksen jatkuvuus kertomalla jatkokehittäjille mihin asioihin he voisivat kiinnittää huomiota.

Markus Hiltunen, Sami Paavilainen

### Seniori365.fi Service Localization

Year	2016	Pages	33
------	------	-------	----

---

The main goal of this practise-based thesis is to convert Seniori365.fi website to other municipalities, starting from Vantaa. Other central goal for the project is to guarantee and ease the development of the website. This will be achieved by creating a document for building the testing environment for the website and producing guidelines on how to further expand the website for other municipalities.

Seniori365.fi is a website which main consumers are elderly people and companies who provide them services and products. Idea of Seniori365.fi is helping elderly people to find interesting services, products and events. The website has been originally created for Espoo and it contains content mostly designed for the elderly in Espoo. There has been demand for this service in Vantaa also so the expansion of the website is in order. The base for further expansion will be created as a result. For the end user this will be seen as an option to choose the municipality to search content with.

The expansion is based on already made plans for the website and demands set by the design team which consists of business students of Laurea. Idea for the document of creating the testing environment as a part of the thesis came up in the planning stage of the project, when it was noticed that there were room for improvement in technical documentation of Seniori365.fi.

The most important evaluator for the project is the customer. This can also be seen on the values of the project that are based around the customer. The end result should be easy to use, light to build technically and easy to expand.

We will be listing the things that come up during the project that could use improvement and we will also give our opinions on how the website could be further developed. Goal for this last part of our thesis is to give guidance for the next developers of the website and point out some parts or problems of the website they should look into.

Keywords: Drupal, Seniori365, webdevelopment, localization

## Sisällys

1	Johdanto.....	6
2	Verkkosivukehityksen työkalut ja käsitteet .....	7
2.1	Drupal .....	7
2.2	Localhost.....	7
2.3	XAMPP .....	7
2.3.1	Apache .....	8
2.3.2	MySQL .....	8
2.3.3	PHP.....	8
2.3.4	PHPMyAdmin .....	9
2.4	Git.....	9
2.5	FTP-pääteohjelma .....	10
2.6	Taksonomia.....	10
2.7	Content type .....	10
3	Localhost Paikallisympäristön perustamisohjeet .....	10
3.1	Localhost-paikallisympäristö Web-kehittäjän työkaluna.....	11
3.2	Paikallisympäristö osana opinnäytetyötä .....	11
3.3	Paikallisympäristön ohjelmien asennus ja käyttöönotto .....	11
3.3.1	XAMPP .....	12
3.3.2	SourceTree .....	13
3.3.3	WinSCP .....	16
3.3.4	PHPMyAdmin konfiguraatiot .....	17
3.3.5	Paikalliseen ympäristöön kirjautuminen .....	20
3.3.6	Yleisten ongelmien vianmääritys .....	20
4	Sisällön lokalisointi .....	21
4.1	Tavoitteet ja vaatimukset .....	21
4.2	Kunta kategorian luonti .....	22
4.3	County taksonomian lisääminen sisältötyyppeihin .....	24
4.4	Filtterin lisääminen .....	25
4.5	Kalenteritapahtumien filtteröinti .....	26
4.6	Kunnan valinnan mallintaminen .....	29
5	Sivuston laajentaminen uusille paikkakunnille .....	30
6	Yhteenveto .....	31
	Lähteet .....	32
	Kuviot .....	33

## 1 Johdanto

Seniori365.fi-palvelu on opiskelijoiden toteuttama verkkosivusto, joka on suunnattu Espoon seniori-ikäisille ja heidän omaisilleen tai avustajilleen. Sivustoa voivat käyttää myös heitä palvelevat yritykset, sekä heidän omaisensa ja vapaaehtoistöistä kiinnostuneet.

Sivusto sisältää linkkejä ja yhteystietoja seniori-ikäisille suunnatuista palveluista, tuotteista ja tapahtumista. Sivustolla on lisäksi muun muassa tietoa omaishoitajille, tietoa sivustosta, sekä ajankohtaisista asioista. Sivuston sisältö on pääsääntöisesti suunnattu Espoolaisille ja tällä hetkellä sivustolla ei ole mahdollista tarkastella kuntakohtaista sisältöä. Jatkokehityksen tavoitteena on luoda mahdollisuus tarkastella kuntasidonnaista sisältöä.

Seniori365.fi-palvelun kehityksestä on vastuussa Laurean aluepalvelupäällikko. Hän toimii myös tämän opinnäytetyön toimeksiantajana. Asiakasyritys tälle toimeksiannolle on Laurea.

Kuntakohtaisen kategorisoinnin luonti toteutetaan kolmen hengen ryhmällä, joka koostuu Laurean tietojenkäsittelyn opiskelijoista. Sivuston kehitystyön vaatimukset määritellään yhdessä sisällöstä ja yritys yhteistyöstä vastaavan ryhmän kanssa, joka koostuu viidestä liiketalouden opiskelijasta. Projektin tarkoitus on mahdollistaa kuntakohtaisen sisällön tuottaminen Vantaalle, mutta laajennusmahdollisuus muille kunnille tulee olla olemassa. Projektin aikana pidettiin palavereja ja tilannekatsauksia tekniikkatiimin, sekä koko projektiryhmän ja toimeksiantajan kesken. Näissä palavereissa seurattiin projektin edistymistä, sekä keskusteltiin mahdollisista haasteista ja ratkaisuista.

Opinnäytetyön päätavoitteena on mahdollistaa kuntakohtaisen sisällön luonti Vantaalle Seniori365.fi-palveluun. Tämä tulisi toteuttaa siten, että tulevaisuudessa muidenkin kuntien implementointi palveluun olisi mahdollisimman sujuvaa ja helposti tehtävissä. Tästä syystä tehdään myös ohjeistus uusien kuntien lisäämisestä palveluun.

Sivutavoitteena pyritään edistämään Seniori365.fi:n jatkokehitystä luomalla ohjeet testiympäristön perustamiselle. Näiden ohjeiden tarkoitus on nopeuttaa seuraavien kehittäjien perehdyttämistä osaksi projektiryhmää. Pyrkimyksenä on luoda yksityiskohtaiset ohjeet, jotka ovat riittävän selkeät testiympäristön luomiselle ilman ulkoista perehdytystä.

## 2 Verkkosivukehityksen työkalut ja käsitteet

Tässä osiossa esitellään teknisessä toteutuksessa ja sen testauksessa käytetyt järjestelmät. Projektin aikana käytetyt järjestelmät valittiin käyttöön, koska projektia on aiemmin kehitetty samoin menetelmin ja projektin aikaisemmat dokumentaatiot tukevat seuraavissa kappaleissa käsiteltyjä järjestelmiä.

### 2.1 Drupal

Drupal on selainpohjainen sisällönhallintajärjestelmä. Drupalilla pystytään kehittämään sekä pieniä, että laajempia verkkosivustoja ilman suurempaa osaamista ja tietämystä verkkosivujen teknisestä kehittämisestä. Drupal on suosittu työkalu verkkosivujen kehittämiseen ja sillä on Suomessakin kehitetty tunnettuja sivustoja kuten esimerkiksi YLE:n verkkosivut. Drupal sivustojen yksi vahvuus on jatkokehityksen ja sivuston laajentamisen mahdollisuus.

(Suomen Drupal -yhdistys, 2014)

Drupalin perusosa on nimeltään Drupal core, jolla voidaan tehdä tavallisia sivuja ja blogeja. Drupal on kuitenkin erittäin hyvin laajennettavissa modulaarisesti. Drupaliin on tehty yli 30 tuhatta moduulia käyttäjien toimesta. Näillä moduuleilla pystytään muuntamaan verkkosivusta esimerkiksi verkkokauppa, galleria tai lisäämään projektinhallinta työkaluja sivuille.

(Wikipedia®, 3. helmikuuta 2015)

### 2.2 Localhost

Localhost viittaa paikalliseen tietokoneeseen itseensä, jonka ip osoitteet ovat 127 alkuisia. Työssämme Localhost paikallisympäristö viittaa Apache palvelimen avulla pyörittämäämme testausympäristöön. Verkkosivun tiedostot ja tietokannat sijaitsevat paikallisesti tietokoneellamme ja näin saadaan luotua oikeaa Seniori365.fi-palvelua simuloiva testausympäristö.

### 2.3 XAMPP

XAMPP on ilmainen avoimen lähdekoodin kokonaisuus, joka sisältää Apache HTTP-palvelimen, MySQL-tietokannan, sekä erilaisia PHP ja PERL -skriptejä. XAMPPin helpon asennettavuuden ja graafisen käyttöliittymän vuoksi se on erittäin suosittu ympäristö verkkosivujen kehitykseen. (Apache Friends)

Seniori365.fi-palvelun kehityksessä voisi myös käyttää XAMPPia vastaavaa järjestelmää, mutta aiemman kehityksen dokumentointi tukee parhaiten tätä kokonaisuutta. Lisäksi yleisyytensä vuoksi XAMPPiin löytyy ongelmatilanteissa helposti ja paljon vertaistukea esimerkiksi kirjoittamalla hakukoneeseen mahdollisen virheilmoituksen tai toimenpiteen, joka kokonaisuudella halutaan saavuttaa.

XAMPP on saatavilla Windows-, Linux- ja OSX -pohjaisille tietokoneille. Tässä projektissa XAMPPia käytettiin sekä Windows 7-, että Linux Mint -käyttöjärjestelmillä. Tietokoneen käyttöjärjestelmä ei vaikuta XAMPPin toimivuuteen merkittävästi, sillä Apache ja MySQL -palvelut pääsee käynnistämään graafisesta käyttöliittymästä ja palveluiden konfiguraatiot tehdään vapaavalintaisella tekstinkäsittelyohjelmalla tekstitiedostoon.

### 2.3.1 Apache

Apache on XAMPPin mukana tuleva HTTP-palvelin, jolla tässä tapauksessa paikallinen Seniori365.fi-palvelun testiympäristö sijaitsee. Palvelin käynnistetään XAMPPin mukana tulevasta graafisesta käyttöliittymästä, jonka jälkeen se toimii tietokoneella taustaprosessina. Apache mahdollistaa verkkosivun tarkastelun selaimessa ja käsittelee sivulla tehdyt pyynnöt paikallisesti ilman internet-yhteyttäkin. (<http://www.wpbeginner.com/glossary/apache/>, What is: Apache)

### 2.3.2 MySQL

MySQL on avoimen lähdekoodin tietokannanhallintajärjestelmä, jota käytetään esimerkiksi verkkosivun taustalla olevan tietokannan käsittelyyn. Järjestelmä perustuu SQL-kieleen, joka on useiten käytetty ja standardisoitu tietokantojen käsittelyyn suunniteltu kieli. (Oracle, 2016)

MySQL asentuu osana XAMPP-pakettia ja sitä käytetään Seniori365.fi-palvelu projektissa yhtenä työkaluna paikallisen testiympäristön luonnissa ja kehityksessä. MySQL paikallisessa ympäristössä mahdollistaa valmiin, toimivasta Seniori365.fi-palvelun versiosta kopioidun tietokantataulun upottamisen paikalliseen ympäristöön.

### 2.3.3 PHP

PHP on nettisivujen toteuttamiseen tarkoitettu suosittu ohjelmointikieli. PHP-koodi suoritetaan palvelimella ennen kun se lähetetään määritetylle verkkosivulle. PHP-koodilla kykenee käsittelemään muun muassa palvelimen tiedostoja ja tietokantoja. PHP:n vahvuuksiin lukeutuvat sen helppokäyttöisyys ja monipuolisuus. PHP on käytettävissä myös kaikissa yleisimmissä web-hotelleissa.

PHP vaatii toimiakseen palvelimen, jossa on mahdollista suorittaa PHP-koodia. Muodoltaan PHP-sivu on samanlainen kuin HTML-sivu, koska palvelin käsittelee PHP koodin enne kuin se lähetetään käyttäjälle ja muuntaa sen sivuksi, joka sisältää pelkkää HTML-koodia. (Antti Laaksonen, 2011)



### 2.3.4 PHPMyAdmin

PHPMyAdmin on ilmainen ohjelmistotyökalu, joka on kirjoitettu PHP koodilla. Se on tarkoitettu MySQL- tietokantojen hallinnointiin verkossa. PHPMyAdmin tukee erilaisia toimintoja liittyen MySQL ja MariaDB tietokantoihin. Suosituimmat toiminnot voidaan toteuttaa käyttöliittymän kautta. Kyseisiä toimintoja ovat muun muassa tietokantojen-, taulujen-, sarakkeiden- ja niiden välisten suhteiden hallinta. Kaikki PHPMyAdminin toiminnot on mahdollista tehdä myös SQL lauseilla.

(phpMyAdmin contributors, 2003-2016)

PHPMyAdminin kehitys on 18-vuotta vanha projekti, jonka kehityksen Tobias Ratschiller aloitti. Hän joutui kuitenkin luopumaan projektin kehityksestä 2000-vuonna. Näihin aikoihin PHPMyAdmin oli saavuttanut todella suosituksen aseman PHP pohjaisten sovellusten joukossa ja MySQL:än hallinta työkaluna. Vuonna 2001 kolme kehittäjää Olivier Muller, Marc Delisle ja Loic Chapeaux ottivat PHPMyAdminin kehityksen vastuulle ja rekisteröivät sen Source Forge.net sivustolle. Vuodesta 2015 jatkokehitys on sijoitettu kokonaan GitHubiin.

(phpMyAdmin contributors, 2003-2016)

## 2.4 Git

Gitillä tarkoitetaan modernia versionhallintajärjestelmää, jota voidaan hyödyntää esimerkiksi ohjelmistojen tai verkkosivujen kehityksessä. Seniori365.fi-palvelun kehityksessä on ollut suosituksena käyttää SourceTree-nimistä ohjelmaa, mutta samat toimenpiteet onnistuvat myös toisilla vastaavanlaisilla ohjelmilla. Tässä projektissa versionhallintajärjestelmä näyttää Seniori365.fi-palvelun kehittäjille tiedostopuun, joista sivusto koostuu. Kun puussa oleviin tiedostoihin tehdään muutoksia, muut kehittäjät näkevät Git-järjestelmän kautta mitä tiedostoa muutettiin, mitä tiedostossa muuttui ja kuka muutoksen teki.

Tämä toiminnallisuus auttaa kehittäjiä pysymään mukana verkkosivuilla tapahtuvissa muutoksissa luoden helppokäyttöisemmän ja turvallisen kehitysympäristön, sillä jokainen muutos jättää järjestelmään aikaleimallisen merkinnän. Tällöin kehitystiimi löytää viimeisimmän version jokaisesta tiedostosta, eikä synny tilannetta, jossa kehittäjä tekee muutoksia jo vanhentuneeseen tiedostoon. Lisäksi tiedostojen näkyminen kaikille kehittäjille auttaa pitämään ohjelmiston tai verkkosivun tiedostot keskenään yhteensopivina.

Git-järjestelmä perustuu repositorioihin, joilla tarkoitetaan keskitettyjä tiedostojen tai tietokantojen säilytyspaikkoja. Repositorioita voidaan käyttää myös esimerkiksi tiedostojen, ohjelmistojen tai tietokantojen levittämiseen internetin välityksellä (Vangie Beal). Seniori365.fi-projektissa hyödynnetään BitBucket-nimisellä sivustolla sijaitsevaa repositoriota, jonne otetaan yhteyttä Git-järjestelmällä.

## 2.5 FTP-pääteohjelma

Seniori365.fi-palvelun sivuston tiedostot sijaitsevat verkkohotellin palvelimella, jonne voidaan ottaa yhteys käyttämällä siihen tarkoitettua tiedostonsiirto-ohjelmaa. Seniori365.fi-palvelu projektissa on aiemmin käytetty tähän WinSCP-nimistä ohjelmaa.

Ohjelma näyttää palvelimella olevat verkkosivun taustalla toimivat tiedostot ja tiedostopuun, sekä mahdollistaa tiedostojen kopioinnin palvelimelta paikalliselle tietokoneelle tai toisin päin, sekä tiedostojen muokkaamisen palvelimella.

Ohjelmaksi on alun perin valittu WinSCP, sillä sen pystyy käynnistämään esimerkiksi Laurea-ammattikorkeakoulun opiskelijakäyttöön tarkoitetuilla tietokoneilla ilman erillistä ohjelmisto-asennusta. Lisäksi se on graafisen käyttöliittymänsä vuoksi melko helppokäyttöinen.

## 2.6 Taksonomia

Taksonomia (eng. taxonomy) tietotekniikan kohdalla tarkoittaa jaottelua, mahdollisuutta tunnistaa ja nimeämistä. Taxonomy on Drupalin core moduuli eli se tulee Drupalin asennuksen mukana. Käytännössä taksonomian avulla pystytään ryhmittelemään haluttu sisältö niille annettujen termien ja ehtojen mukaan ryhmiin. Taksonomia muun muassa helpottaa valikoiden organisointia, sekä mahdollistaa sisällön hallinnan kategoroinnin avulla.

(Drupal, 22.8.2016)

## 2.7 Content type

Drupalissa content type tarkoittaa sisältötyyppiä. Käytännössä kun drupal sivulle lisätään sisältöä pitää määrittää sen sisältötyyppi. Oletuksena drupalin mukana tulevat sisältötyypit ovat basic page eli normaalisivu ja article eli artikkeli. Sisältötyyppejä voi luoda myös itse, milloin päästään määrittelemään tarkemmin minkälaisesta sisällöstä on kyse.

(Drupal, 3.11.2016)

## 3 Localhost Paikallisympäristön perustamisohjeet

Projektitöiden jatkuvuuden takaamiseksi on oltava olemassa tarvittavat dokumentaatiot ja perehdytysuunnitelmat. Seniori365.fi:n kohdalla oletettavaa on, että työryhmä vaihtuu ja uudet projektiryhmän jäsenet tarvitsevat perehdytystä. Tavoitteena on tehostaa jatkokehittäjien implementointia osaksi tiimiä ja tukea heitä Seniori365.fi:n jatkuvuuden varmistamiseksi.

### 3.1 Localhost-paikallisympäristö Web-kehittäjän työkaluna

Paikallisympäristö on testaukseen ja verkkosivujen kehittämiseen tarkoitettu paikallinen versio verkkosivustosta. Tässä ympäristössä verkkosivustolle suunnitellut muutokset voidaan testata turvallisesti. Tämä ympäristö tarjoaa myös mahdollisuuden harjoitella verkkosivukehitystä ja rakentaa verkkosivustot paikallisesti ennen niiden siirtämistä verkkoon.

Paikallisympäristö toimii paikallisella serverillä, joka luodaan esimerkiksi Xampp:in avulla. Lisäksi paikallisympäristö tarvitsee tietokannan. Tietokantana voidaan käyttää esimerkiksi MySQL tietokantaa. Tiedostot joita paikallisympäristö käyttää löytyvät omalta tietokoneelta, mutta haluttaessa jatkokehittää verkkosivua jo olemassa olevat tiedostot pitää usein siirtää kyseisen verkkosivuston palvelimelta omalle tietokoneelle. Seniori365.fi:n tapauksessa käytetään XAMPP pakettia ja rakennetaan paikallisympäristö Apache serverille, sekä käytetään PHPMyAdminia, jonka avulla hallitaan SQL tietokantaa.

### 3.2 Paikallisympäristö osana opinnäytetyötä

Localhost-paikallisympäristö on välttämätön työkalu osana projektia. Tämän kehitysympäristön avulla voidaan testata suunnitellut toteutukset ja muutokset Seniori365.fi-palvelun verkkosivuille. Projektin alussa Seniori365.fi-palvelun siirtämisessä paikallisympäristöön ilmeni kuitenkin ongelmia, jotka johtuivat suurilta osin ohjeistuksen puuttesta. Nämä ongelmat viivästyttivät opinnäytetyön verkkosivujen kehitysvaiheen eli Seniori365.fi-palvelun sisällön lokalisoinnin aloittamista.

Seniori365.fi kaltaisten palveluiden kannalta on erityisen tärkeää, että uudet projektin jäsenet saataisiin tehokkaasti osaksi projektiryhmää ja hyvin perehdytettyä. Seniori365.fi-palvelun projektiryhmä on vaihtunut useasti ja sen dokumentoinnissa on parantamisen varaa. Ohjeistus päätettiin aloittaa siitä, miten paikallisympäristö asennetaan. Tämän ohjeistuksen tavoite on nopeuttaa ja tehostaa mahdollisten sivuston jatkokehittäjien siirtymistä osaksi projektiryhmää ja näin ollen helpottaa Seniori365.fi-palvelun kehittämistä.

Ohjeista pyritään luomaan selkeät ja tarvittavan yksityiskohtaiset. Ohjeistus on kuvitettu ja sen tavoitteena on olla niin selkeä ettei aiempi kokemus paikallisympäristön asennuksesta ole välttämätöntä.

### 3.3 Paikallisympäristön ohjelmien asennus ja käyttöönotto

Paikallisympäristön perustaminen aloitetaan lataamalla siihen tarvittavien ohjelmistojen asennusmediat. Tässä ohjeessa on mainittu Seniori365.fi-projektissa toimivaksi todettuja ohjelmistoja, mutta työ on mahdollista tehdä myös vaihtoehtoisilla ohjelmistoilla, mikäli työssä

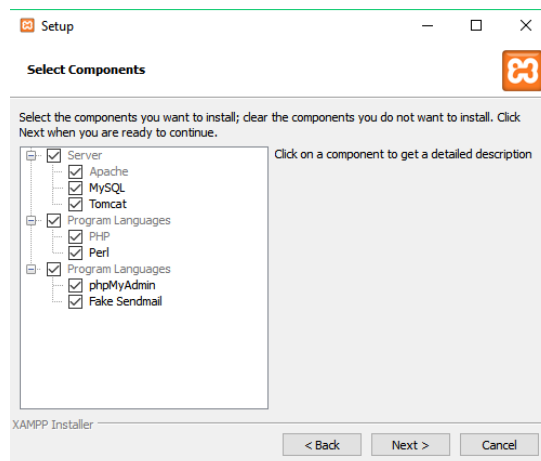
käytettävä käyttöjärjestelmä ei tue suositeltuja ohjelmia. Suurin osa listatuista työkaluista on kuitenkin saatavilla myös Linuxille tai Macille joko suoraan, tai käyttämällä niitä erillisen yhteensopivuuskerroksen tai emulaattorin (kuten Linuxissa WINE, tai Macissa Boot Camp) kautta. Projektissa käytetyt erikseen asennettavat ohjelmat:

- XAMPP (WIN / Linux / OSX)
- Sourcetree (WIN / OSX)
- WinSCP (WIN)

Suosituksena on myös ottaa käyttöön vapaavalintainen koodille sopiva tekstieditori, esimerkiksi Sublime Text tai Notepad++.

### 3.3.1 XAMPP

Ohjattu asennus aloitetaan käynnistämällä ohjelmiston kehittäjän verkkosivuilta ladattu asennusmedia. Asennuksessa voi huoletta valita kaikki ehdotetut komponentit.

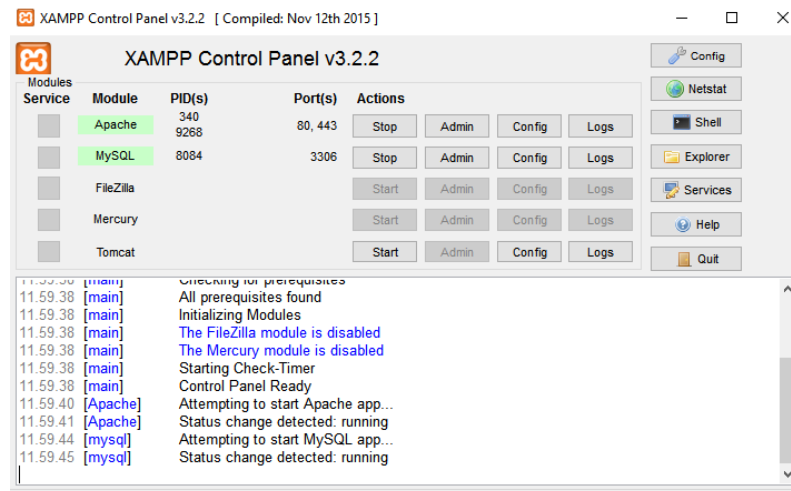


Kuva 1: Asennettavat XAMPP-komponentit

Kun XAMPP on asentunut, käynnistetään XAMPP -ohjauspaneeli mieluiten järjestelmänvalvojana (Run as Admin). Avautuvasta ohjauspaneelistä käynnistetään moduulit **Apache** ja **MySQL** painamalla niiden vierestä **Start**. Käynnistyessään oikein, moduulin nimi samaa vihreän taustan. Windows 10 -käyttöjärjestelmällä on todettu, että Apachen ja Skypen käyttö samanaikaisesti aiheuttaa virhetilanteen, sillä molemmat yrittävät oletuksena käyttää porttia 80 ja SSL-porttia 443.

(Drupal, 22.9.2016)

Toisinaan, kun XAMPPin käynnistää moneen kertaan uudelleen, myös MySQL saattaa jättää prosessinsa taustalle jumiin estäen moduulin käynnistytksen. Normaalisti tähän auttaa viimeistään tietokoneen käynnistäminen uudelleen.



Kuva 2: XAMPP-ohjauspaneelista käynnistettävät moduulit

Toimivuus voidaan testata kirjoittamalla internet-selaimen osoitepalkkiin **localhost** ja painamalla enter. Mikäli selain ohjautuu "Welcome to XAMPP" -sivulle, on asennus onnistunut eikä vaadi vielä tässä vaiheessa lisämäärytyksiä.

### 3.3.2 SourceTree

Ennen SourceTreen asennusta, on luotava käyttäjätunnus <https://bitbucket.org> -sivustolle. Seniori365.fi-palvelun live-version repositorio sijaitsee täällä. Lisäksi tarvitaan kutsu olemassaolevalta repositorion käyttäjältä, jotta tiedostoja voi tarkastella ja ladata.

Asennuksen aikana SourceTree pyytää käyttäjää kirjautumaan sisään Atlassian-tunnuksella. Tähän voidaan käyttää aiemmin luotuja Bitbucket-tunnuksia, sillä molemmat palvelut ovat samalta valmistajalta. SourceTreen mahdolliset kysymykset etäpalvelimista ja SSH-avaimesta voidaan ohittaa.

Tämän jälkeen SourceTree todennäköisesti ilmoittaa Git-asennuksen puutteesta, jolloin valitaan SourceTreen oman upotetun Gitin lataamisen. Sama tehdään tarvittaessa myös **Mercurial**-asennukselle.

SourceTree: Git not found



We were not able to locate a Git install on your system already. There are 3 ways we can deal with this:

Download an embedded version of Git for SourceTree alone to use.

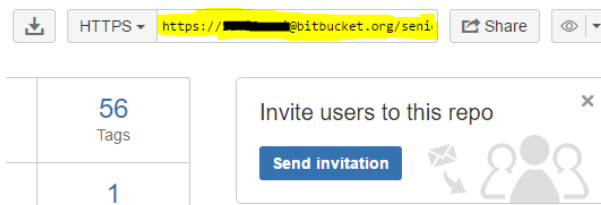
Browse to the location of Git on your system (if we missed it)

Check again (should you wish to download and install a full version of Git yourself)

I don't want to use Git

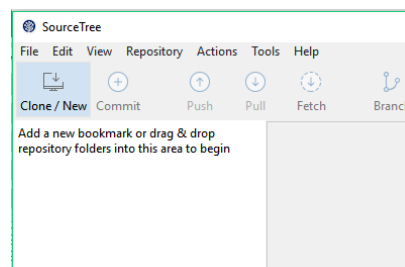
Kuva 3: Git-lataus SourceTreessä

Asennuksen valmistuttua, navigoidaan Bitbucketissa Seniori 365 kansioon main-site ja kopioidaan oikeassa yläkulmassa oleva **HTTPS-osoite**.



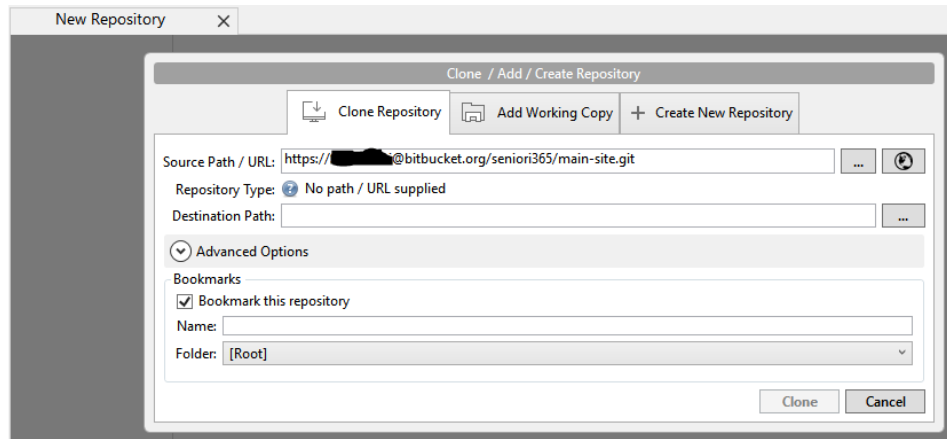
Kuva 4: Repositorion osoite

Tämän jälkeen SourceTreessä klikkataan vasemmasta yläkulmasta **Clone / New**.



Kuva 5: Repositorion tuominen SourceTreehen

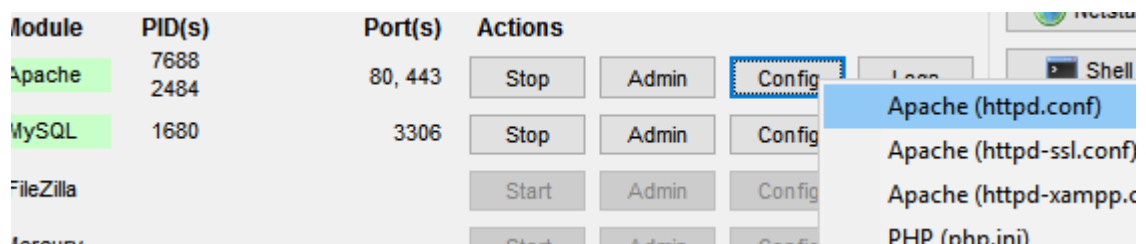
Avautuvassa ikkunassa, ensimmäisessä välilehdessä sijaitsee kenttä **Source Path / URL**. Täähän liitetään aiemmin kopioitu **HTTPS-osoite**.



Kuva 6: Repositorion tuonnin määrittäminen

Destination Pathiksi voidaan asettaa vapaavalintainen kansio, josta tiedostot halutaan löytää. Mikäli SourceTree pyytää jälleen kirjautumaan sisään, käytetään aiemmin luotua Bitbucket-tunnusta, jolla on oikeudet päästä Seniori365.fi-repositorioon. Tämän jälkeen valitaan **Clone** ja tiedostot latautuvat määritettyyn kansioon. Tämän jälkeen SourceTree pyytää nimeä ja sähköpostia, joka näytetään muille Seniori365.fi-repositorion käyttäjille osana sujuvaa versi-onhallintaa.

Lopuksi tehdään vielä muutoksia **XAMPPin** konfiguraatioon. **Apache**-moduulin vierestä valitaan **Config** ja otetaan muokattavaksi **Apache (httpd.conf)**.



Kuva 7: Apache-juuren määrittäminen

Avautuvasta tiedostosta tulee etsiä kohta **DocumentRoot**, joka osoittaa oletuksena XAMPPin asennuskansiossa olevaan polkuun. Aseta sen sijaan molempiin kohtiin poluksi aiemmin ladattu **main-site** -kansio. Alla olevassa esimerkkikuvassa kansio sijaitsee E: -asemalla.

```
#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "E:\Localhost-kamat\main-site"
<Directory "E:\Localhost-kamat\main-site">|
..
```

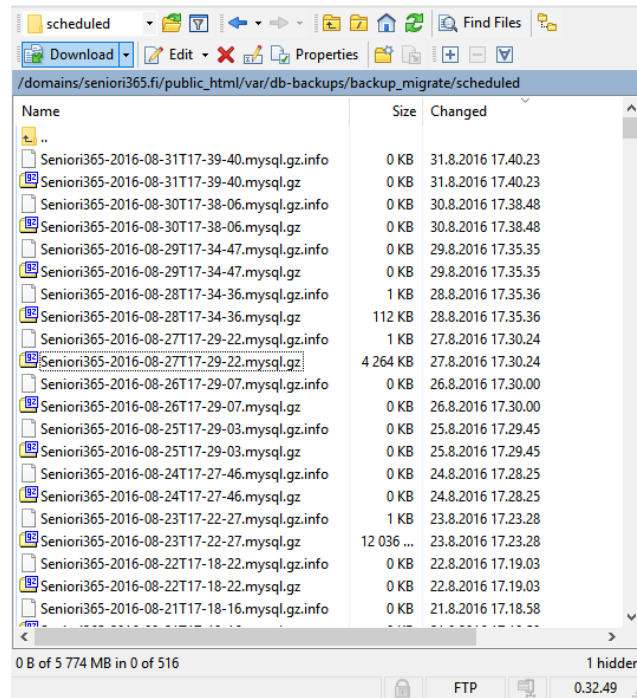
Kuva 8: Esimerkki Apache-juuren määrittämisestä

### 3.3.3 WinSCP

Mikäli käytössä on WinSCP -ohjelman portable-versio, voidaan sovellus avata suoraan kuvakeesta ilman erillistä asennusta. Tämä on hyödyllistä esimerkiksi silloin, kun yhteys palvelimeen täytyy ottaa esimerkiksi Laurea-ammattikorkeakoulun tietokoneelta, joihin normaaleilla opiskelijatunnuksilla ei voida tehdä ohjelmistoasennuksia. Muutoin ohjelma asennetaan noudattamalla ruudulla näkyviä ohjeita. Kun WinSCP käynnistetään ensimmäistä kertaa, ohjelma pyytää ensin kirjautumaan sisään. Tietoturvasyistä tässä dokumentissa ei paljasteta palvelimen kirjautumistietoja, mutta ne ovat saatavilla Seniori365.fi-palvelun projektiryhmälle perustetussa pilvitalennustilassa.

Tarkoituksena on tässä vaiheessa hakea palvelimelta viimeisin varmuuskopio sivuston SQL-taulusta. Varmuuskopiot löytyvät palvelimen kansioista **./public\_html/var/db-backups/backup-migrate/scheduled/**. Tiedostot voi myös etsiä joko painamalla ALT + F7 tai klikkaamalla ikkunan oikealta puolelta "Find files." Tällöin haku tehdään tiedostopäätteen mukaan termillä **\*mysql\*** (tähdet mukaan lukien).





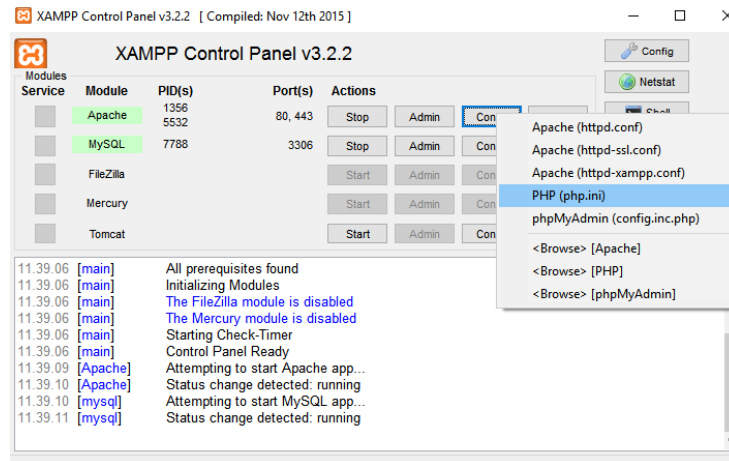
Kuva 9: Polku, jossa Seniori365.fi-palvelun varmuuskopiot sijaitsevat

Nopeinta on järjestää tiedostot muokkauspäivän mukaan ja ladata palvelimelta uusin **.mysql.gz** -päätteinen tiedosto, jonka koko on noin **30 000kb tai suurempi**. Tällöin tiedosto todennäköisesti sisältää kaikki Drupal -asennukseen tarvittavat tiedostot. Tiedoston voi ladata esimerkiksi vetämällä sen suoraan WinSCP -ohjelmasta omalle työpöydälle tai kansioon. Palvelimen polusta **/domains/seniori365.fi/public\_html/sites/default** haetaan myös kansio **files**, sekä **settings.php** ja ne kopioidaan paikallisen tietokoneen **\*\main-site\sites\default** -kansioon.

### 3.3.4 PHPMyAdmin konfiguraatiot

Edellisessä osioissa ladattu **.mysql.gz** tiedosto tulee ladata paikalliselle palvelimelle. Ensin tehdään kuitenkin säätöjä, sillä yleensä kyseisenlaiset tiedostot ovat suurempia, kuin mitä XAMPPin oletusasetukset sallivat normaalisti ladata palvelimelle.

Tämä aloitetaan ottamalla esille XAMPP ohjauspaneeli ja klikkaamalla moduulin **Apache** vierestä **Config** ja kohtaa **PHP.ini**.



Kuva 10: PhpMyAdmin -konfiguraatitiedoston avaaminen

Avatusta tekstitiedostosta haetaan seuraavat arvot esimerkiksi CTRL + F -komennolla, ja muokkataan niiden arvoja vastaamaan oman ympäristön tarpeita. Tässä ohjeessa annetut arvot ovat esimerkkejä, joiden pitäisi toimia useimmissa tilanteissa.

`upload_max_filesize = 10240M`

`memory_limit = 2048M`

`post_max_size = 0M`

`user_ini.cache_ttl = 9001`

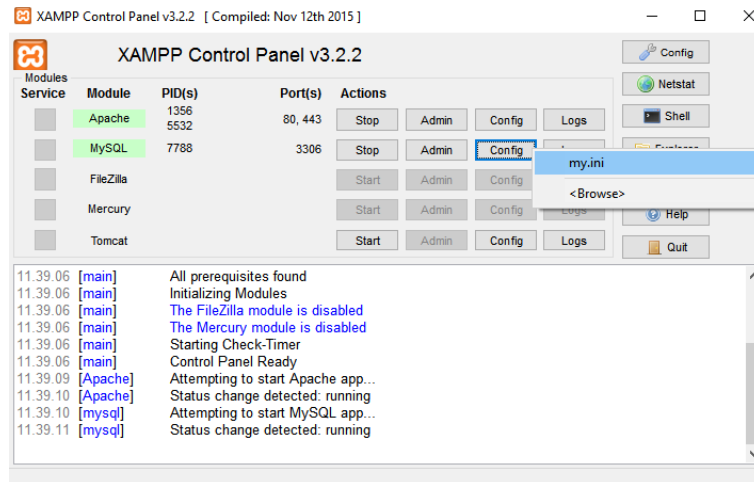
`default_socket_timeout = 9001`

`mysql.connect_timeout = -1`

Tiedostoon **phpMyAdmin (config.inc.php)** lisätään seuraava rivi kohdan **Server configuration** alle:

`$cfg['ExecTimeLimit'] = 0;`

Tämän jälkeen avataan XAMPP -ohjauspaneelista moduulin **MySQL** vierestä **Config** ja valitaan **my.ini**.



Kuva 11: MySQL -konfiguraatiotiedoston avaaminen

Samoin kuin aiemmin, etsitään seuraavat parametrit ja tehdään seuraavat muutokset:

`key_buffer = 9999M`

`max_allowed_packet = 9999M`

`sort_buffer_size = 9999M`

`net_buffer_length = 9999M`

`read_buffer_size = 9999M`

`read_rnd_buffer_size = 9999M`

`mysam_sort_buffer_size = 9999M`

Kun muutokset on tehty ja tallennettu, käynnistetään moduulit Apache ja MySQL uudelleen painamalla niiden vierestä ensin Stop, ja sitten Start.

Tämän jälkeen navigoidaan internet-selaimella osoitteeseen **localhost/phpmyadmin/**. Uusi SQL-taulu luodaan painamalla vasemmalla olevasta listasta "New." Taulun nimeksi tulee tässä vaiheessa **seniori365\_melaz**. Kun taulu on luotu, se avataan ja klikataan ylhäältä "**Tuonti / Import**".

Tuotava tiedosto valitaan painamalla **Choose file**. Tähän syötetään **.mysql.gz** -päätteinen tiedosto, joka aiemmin ladattiin WinSCP:n avulla. Lataus hyväksytään lopuksi sivun alareunasta.

Tiedoston lataus ei näytä itsestään edistymistietoa ja lataaminen voi ilman SSD-levyä kestää useita tunteja.

### 3.3.5 Paikalliseen ympäristöön kirjautuminen

Aiemmin WinSCP:lla palvelimelta ladattuun, `*\main-site\sites\default` -polussa sijaitsevaan `settings.php` -tiedostoon tehdään seuraavat muutokset:

```
$databases = array('default' => array( 'default' => array(
'driver' => 'mysql',
'database' => 'seniori365_melaz',
'username' => 'root',
'password' => "",
'host' => 'localhost',
'port' => '3306',
'prefix' => 'drup_',
'collation' => 'utf8_general_ci',
)));
```

Tämä määrittää phpMyAdminille käytettävän tietokannan nimen ja siihen liittyvät kirjautumistiedot. Mikäli kirjautumistiedot puuttuvat, localhost ohjaa ”Tilapäinen käyttökatko” -sivulle.

Kun sivusto näkyy normaalisti toimivana paikallisesti, voidaan kirjautua myös Drupaliin osoitteessa localhost/user käyttämällä voimassa olevia Drupal-tunnuksia. Tämän jälkeen työympäristö on käyttövalmis.

### 3.3.6 Yleisten ongelmien vianmääritys

Jos Apache tai MySQL ei käynnisty:

Tarkistetaan käytössä olevat portit. Oletuksena Apache ottaa käyttöön portit 80 ja 443, sekä MySQL portin 3306. Jos portit ovat käytössä, moduulit eivät suostu käynnistymään ohjauspaneelistä. Käynnissä olevat verkkoa käyttävät prosessit voidaan tarkistaa esimerkiksi komentorivin komennolla `netstat -ano | find "80"`. Tämä näyttää kaikki porttia 80 käyttävät prosessit, jolloin niitä voi verrata resurssienhallinnassa näkyviin prosesseihin. On myös hyvä tarkistaa, ettei käytettävät portit ole estetty esimerkiksi palomuurin toimesta.

SQL-taulu ei lataudu phpMyAdminiin, jolloin tulee Aikakatkaistu tai muu virheilmoitus:

Tarkistetaan `php.ini`, `config.inc.php`, sekä `my.ini` -tiedostojen asetukset. Tiedostoa ei myöskään voida tuoda tauluun, joka ei ole ennestään tyhjä. Virhetilanteen jälkeen on suositeltavaa tuhota taulu kokonaan ja luoda uusi. Vika saattaa myös olla ladatussa `.mysql.gz` -tiedostossa, jolloin kannattaa yrittää toista vastaavanlaista.

Jos tulee ilmoitus tilapäisestä käyttökatkoksesta tai sivusto pyytää asentamaan Drupalin: Tarkistetaan paikallisen työaseman `settings.php` -tiedoston asetukset. Tarkistetaan, että `settings.php` löytyy paikallisesta polusta `*\main-site\sites\default`, jossa on myös tiedosto `default.settings.php`.

Jos Drupaliin kirjautuminen osoitteessa `localhost/user` ei onnistu:

Palvelimelta ladattu varmuuskopio ei sisällä verkossa olevaan versioon luotuja tunnuksia, mikäli varmuuskopion päivämäärä on vanhempi, kuin tunnuksen luontipäivä. Tässä tilanteessa voidaan käyttää Seniori365.fi-palvelun projektiryhmän pilvitalennustilasta löytyvää admin-tunnusta.

## 4 Sisällön lokalisointi

Osana opinnäytetyötämme toteutamme Seniori365.fi-palvelun verkkosivun sisällön kategorisoinnin ja mahdollistamme kuntakohtaisen sisällön tuottamisen. Toimeksiantomme on luoda mahdollisuus selata sivustolla Espoon ja Vantaan kunnille suunniteltua sisältöä. Ennen tätä opinnäytetyönä toteutettua projektia käyttäjän ei ole mahdollista selata sisältöä kuntakohtaisesti sivustolla. Seniori365.fi-palvelun jatkokehityksestä on tehty aikaisempia suunnitelmia ja opinnäytetöitä, joiden pohjalta lokalisoidun sisällön luomis ominaisuutta lähdetään rakentamaan.

### 4.1 Tavoitteet ja vaatimukset

Vaatimukset Seniori365.fi-palvelun verkkosivuille ja sen lokalisoinnille saimme toimeksiantajaltamme, sekä sisällöntuotannosta ja yritys yhteistyöstä vastaavalta suunnittelutiimiltä. Tämä suunnittelutiimi koostu Laurean liiketalouden opiskelijoista.

Seniori365.fi-palvelun sivusto on suunnattu ensisijaisesti seniori-ikäisille ihmisille, joten sivuston helppo käytettävyys on erityisen tärkeää. Sivuston käytettävyyden säilyttäminen onkin tärkein vaatimus sivustolle. Päätimme, että pyrimme pitämään sivuston rakenteen lähes nykyisen kaltaisena, jotta nykyiset käyttäjät sopeutuvat muutoksiin mahdollisimman hel-

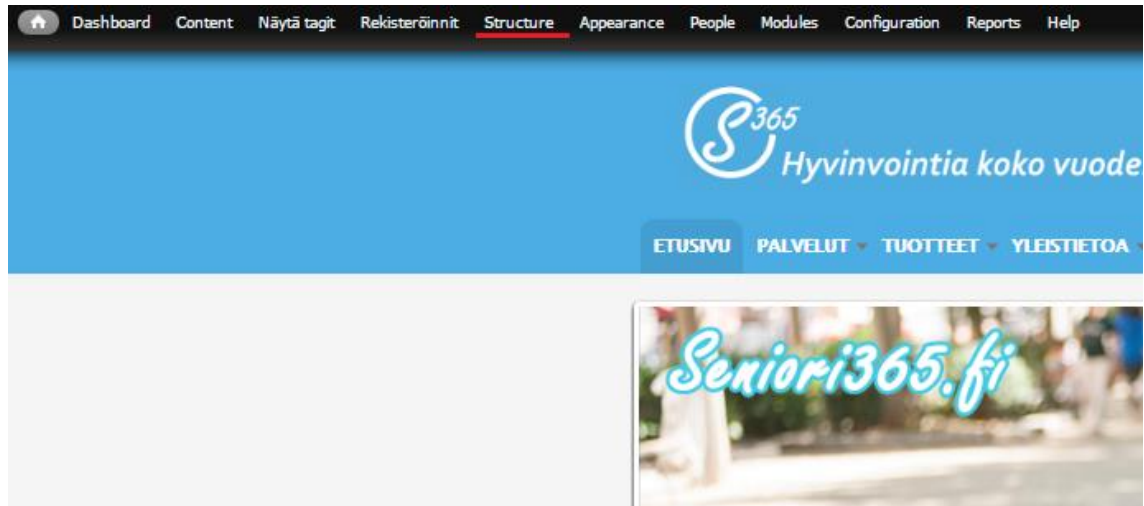
posti. Muutokset eivät myöskään saa rikkoa vanhoja toiminnallisuuksia sivustolla, kuten ole-massa olevaa hakuvaihtoehtoa. Laajennus pitää toteuttaa myös jatkokehitystä ajatellen eli uuden kunnan lisäämisen tulisi olla mahdollista kohtalaisen helposti.

Kaikkeen sivuston sisältöön ei luoda kategorisointia vaan kävimme yhdessä suunnittelutiimin kanssa kategorisoinnin vaativat sivuston osat. Näitä osioita ovat ensisijaisesti palvelut, tuotteet, linkit ja aktiviteettien alisivuna oleva kalenteri.

#### 4.2 Kunta kategorian luonti

Kuntakohtaisen sisällön luomisen mahdollistamiseksi luotiin ensimmäisenä kategorisointi. Kategorisointi on käytännöllisin tapa luoda pohja lokalisoidulle sisällölle. Kategorisointi mahdollistaa sisällön jakamisen erilaisiin ryhmiin valitun tagin eli sanan perusteella. Meidän tapauksessamme sisältöä kategorisoidaan kunnan perusteella.

Kategorian lisääminen tapahtuu drupalin hallintapaneelin kautta Structure → Taxonomy → Add vocabulary.



Kuva 12: Structure valikon sijainti admin näkymässä

Seuraavaksi siirrytään näkymään, jossa valitaan taksonomiale nimi. Nimesimme taksonomian ”County” nimellä. Taksonomiaa luodessa on hyvä antaa myös sille kuvaus ”Description kohtaan”. Tämän kuvauksen on tarkoitus auttaa nykyisiä- ja jatkokehittäjiä ymmärtämään, min-kälaisia termejä taksonomiaan tulee ja voi lisätä myöhemmin. Tämä kuvaus auttaa jatkokehittäjiä ymmärtämään sivuston rakennetta.

**Name** ▼  
 Machine name: county [\[Edit\]](#)

**Description**

---

▶ **XML SITEMAP**

---

▼ **MULTILINGUAL OPTIONS**

**Translation mode**

- No multilingual options for terms. Only the vocabulary will be translatable.
- Localize. Terms are common for all languages, but their name and description may be localized.
- Translate. Different terms will be allowed for each language and they can be translated.
- Fixed Language. Terms will have a global language and they will only show up for pages in that language.

For localizable elements, to have all items available for translation visit the [translation refresh](#) page.

---

▼ **TAXONOMY MENU**

**Menu location**  
 ▼  
 The menu and parent under which to insert taxonomy menu items.

**Menu path type**  
 ▼  
 The path will be taxonomy/term/tid if *Default* has been selected.  
 The menu path will be passed through `drupal_get_path_alias()` function so all aliases will be applied.

Kuva 13: Lisäämämme County taksonomia-sanasto

Tallennettuamme ”County” taksonomian luotiin sille sanastot. Meidän tapauksessamme termeiksi tulevat halutut kunnat, joiden mukaan pystytään myöhemmin filttöimään sisältöä. Sanastoa luodaan Add term painikkeen kautta, joka löytyy kun valitaan luomamme taksonomia ”County” → List. Emme lisänneet mitään luomamme taksonomian Manage fields tai manage display valikkoon.

[+ Add term](#) [Show row weights](#)

NAME	OPERATIONS
<a href="#">+</a> Espoo	<a href="#">edit</a>
<a href="#">+</a> Vantaa	<a href="#">edit</a>

Kuva 14: Lisätyt termit eli kunnat, joiden perusteella sisältöä tullaan kategorisoimaan

### 4.3 County taksonomian lisääminen sisältötyyppeihin

Kun taksonomia on luotu pitää määrittellä mihin sisältötyyppeihin se vaikuttaa. Meidän tapauksessamme tämä käytännössä lisää mahdollisuuden määrittää tietyn sisältötyypin kunnan. Sisältötyypit löytyvät hallintapaneelin kautta Structure → Content types, josta etsitään se sisältötyyppi mitä halutaan muokata ja valitaan operations kohdasta manage fields. Käytämme esimerkkinä dokumentaatioissa sivuston Linkit sisältötyyppeihin.

Seuraavaksi lisäsimme Linkit sisältötyyppeihin jo luomamme taxonomian County kenttänä (Add existing field). Valitsimme valikosta kentän tyyppiä viittauksen taksonomia Countyn termeihin ja "widget" kohdasta "check boxes and radio buttons", mikä luo mahdollisuuden määrittää minkä County taksonomia termin alle luotu sisältö kuuluu. Lisäsimme County -kentän muihin valittuihin sisältötyyppeihin samalla toimintaperiaatteella.

LABEL	MACHINE NAME	FIELD TYPE	WIDGET	OPERATIONS
+ Linkki	title	Node module element		
+ XML sitemap	xmlsitemap	XML sitemap module element		
+ URL path settings	path	Path module form elements		
+ Body	body	Long text and summary	Text area with a summary	edit delete
+ Meta tags	metatags	Meta tag module form elements.		
+ Add new field				
<input type="text"/>		- Select a field type -	- Select a widget -	
Label		Type of data to store.	Form element to edit the data.	
+ Add existing field				
County	Term reference: field_county (County)		Check boxes/radio buttons	
Label	Field to share		Form element to edit the data.	

Save

Kuva 15: County kentän lisääminen linkit sisältötyyppeihin

Nyt kun luodaan uutta sisältöä on sisällön luonti lomakkeeseen ilmestynyt uusi kenttä. Tässä kentässä valitaan minkä County taksonomiaan määritellyn termin mukaan sisältö kategorisoidaan. Meidän työssä tämä käytännössä tarkoittaa minkä kunnan palvelu tai tapahtuma on esimerkiksi kyseessä. Tämän luodun kategorisoinnin pohjalle pystymme rakentamaan mahdollisuuden tuottaa kuntakohtaista sisältöä.



**County**

- Espoo
- Vantaa

**Menu settings**

Not in menu

- Provide a menu link

[Provide information](#)

Kuva 16: Kunnan valinta sisältöä luodessa

#### 4.4 Filtterin lisääminen

Päätimme toteuttaa paikkakuntaakohtaisen sisällön mahdollistamisen filtteröimällä näkymiä (views). Päädyimme yhdessä suunnittelutiimin kanssa ratkaisuun, jossa sivuston käyttäjä voi itse valita dropdown valikosta haluamansa kunnan. Käytämme Drupalin filter criteria ominaisuutta tämän toiminnallisuuden toteuttamiseen. Päädyimme tähän ratkaisuun, koska se saadaan toteuttua yhtenäisesti sivuston eri kategorisoitaville osille, on käyttäjäystävällinen, mahdollista implementoida osaksi sivustoa ilman liiallista kuormitusta, sekä helposti laajennettavissa jatkokehitystä ajatellen.

## 4.5 Kalenteritapahtumien filtteriöinti

Käytämme työssämme esimerkkinä kalenterin filtteriöintiä. Filtterin luonti aloitetaan menemällä kalenterille luotuun näkymään Structure → Views → Calendar. Seuraavaksi lisäsimme filtterin painamalla ”filter criteria” kohdasta ”add” painiketta.

Home » Administration » Structure » Views » View  
Modify the display(s) of your view below or add new displays.

**Displays**

Month Week Day Year Block Upcoming + Add edit view name / description

**Month details**

Display name: Month view Month

**TITLE**  
Title: Tapahtumat

**FORMAT**  
Format: Calendar | Settings  
Show: Calendar Entities | Settings

**FIELDS** Add

Content: Title  
Content: Tapahtuma-aika

**FILTER CRITERIA** Add

Content: Published (Yes)  
Content: Type (= Calendar event)  
Content: Tapahtuman tyyppi (= Kalenteritapahtuma)  
Field: County (exposed) | Settings

**SORT CRITERIA** Add

Content: Tapahtuma-aika - start date (asc)

**PAGE SETTINGS**

Path: /tapahtumat  
Menu: Tab: Month | Parent menu item  
Access: Permission | View published content

**HEADER** Add

**FOOTER** Add

**PAGER**  
Use pager: Page by date | Position: top, format: clean.  
More link: No

**META TAGS**  
Meta tags: Overridden

**Advanced**

**CONTEXTUAL FILTERS** Add

Date: Date (node) (Content: Tapahtuma-aika - start date (field\_tapahtuma\_aika))

**RELATIONSHIPS** Add

**NO RESULTS BEHAVIOR** Add

**EXPOSED FORM**  
Exposed form in block: No  
Exposed form style: Basic | Settings

**OTHER**  
Machine Name: page\_1  
Comment: No comment  
Use AJAX: No  
Hide attachments in summary: No  
Hide contextual links: No  
Use aggregation: No

Kuva 17: Filtterin lisääminen kalenteriin

Seuraaviksi etsimme luomamme County kentän käyttäen hakua. Valitaan Field: County (field\_county) kenttä ja siirrytään eteenpäin Apply painikkeella. Tämän jälkeen valintatyypiksi määritellään ”Dropdown”. Tästä siirrytään filtterin konfigurointiin.

Search  Filter

- Content: Comment count  
The number of comments a node has.
- Content: County (field\_county)  
This is an alias of Field: County (field\_county).
- Field: County (field\_county)  
Appears in: node:ruokaohjeet, node:product, node:service, node:calendar\_event, node:e\_palvelut, taxonomy\_term:tuotteet, taxonomy\_term:palvelut, taxonomy\_term:tst. Also known as: Taxonomy term: County (field\_county), Content: County (field\_county).
- Field: County (field\_county:delta)  
Delta - Appears in: node:ruokaohjeet, node:product, node:service, node:calendar\_event, node:e\_palvelut, taxonomy\_term:tuotteet, taxonomy\_term:palvelut, taxonomy\_term:tst.
- Taxonomy term: County (field\_county)  
This is an alias of Field: County (field\_county).
- File Usage: Use count  
The number of times the file is used by this entity.

Kuva 18: Taksonomian haku

Filtteri tulee konfiguroida, jotta se toimii halutulla tavalla. Oleellisin toiminnallisuus filteri ratkaisussamme on mahdollistaa kunnan valinta käyttäjälle dropdown valikon avulla. Käytämme jo olemassa olevaa toiminnallisuutta ”Expose filteri:ä”. Expose filter luo käyttäjälle valikon, josta hän valitsee haluamansa kunnan. Kunnat ovat määriteltynä County taksonomiaan. Label kohtaan otsikoidaan filteri. Otsikoimme filterin ”valitse kunta”, koska tämä otsikointi näkyy käyttäjälle Dropdown valikon yläpuolella ja näin ohjeistaa käyttäjää. Tarkastetaan, että operator kohtaan on valittu ”Is one of”. Muut konfiguraatiot voidaan jättää vielä oletukselle.

**Configure filter criterion: Field: County (field\_county)**

For

Appears in: node:ruokaohjeet, node:product, node:service, node:calendar\_event, node:taxonomy\_term:tst. Also known as: Taxonomy term: County (field\_county), Content: C

Expose this filter to visitors, to allow them to change it

**Filter type to expose**

Single filter

Grouped filters

Grouped filters allow a choice between predefined operator|value pairs.

Kuva 19: Filterin konfigurointi

Nyt käyttäjä näkee kalenterisivulla tapahtumat valitseman kunnan perusteella. Tämä filteri vaikuttaa vain kyseiseen sivuun, eikä valinta jää muistiin. Kunta valinta tehdään valitsemalla pudotusvalikosta kunta ja painamalla Apply painiketta. Valittaessa Any vaihtoehto pudotusvalikosta näyttää kalenteri kaikki tapahtumat. Kunnan valinta kalenterisivulla on vielä ulkoasultaan oletusmuodossa. Tulemme muuttamaan painikkeet suomenkielelle. Vastaava toteutus tehdään muille määritellyille sivuille, kuten palveluille.

**November 2016** « Prev Next »

Valitse kunta  
 Apply

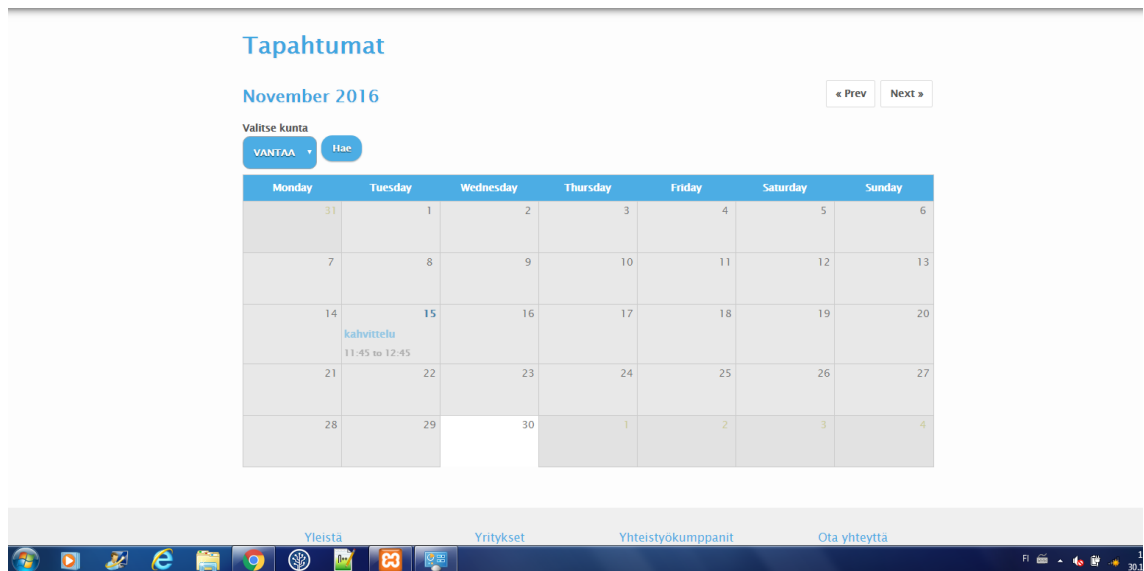
Leave blank for all. Otherwise, the first selected term will be the default instead of "Any".

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15 kahvittelu 11:45 to 12:45	16	17	18	19	20
21	22	23	24	25	26	27

Kuva 20: Käyttäjän näkymä, kun kunnaksi on valittu Vantaa

#### 4.6 Kunnan valinnan mallintaminen

Sivuston yhtenäisen ulkoasun ylläpitämisen kannalta on kunnan valinta mahdollisuuden mallintaminen sivuston muun ulkoasun mukaiseksi suotavaa. Tämän toteuttaminen onnistuu CSS-koodin avulla. Muutokset tehdään sivuston tyylitiedostoon, div-luokan .form-select alle.



Kuva 21:Tapahtumakalenteri CSS-muutoksien jälkeen

Jotta kuntavalinnan käyttöliittymä pysyy muun sivuston kanssa yhdenmukaisena, on painikkeen taustana käytetty vaalean sinistä väriä, jonka HTML -värikoodi on #4cade4. Väri toimii myös hyvin sekä tumman, että vaalean tekstin taustalla. Lisäksi painikkeissa teksti on lihavoituna ja isoilla kirjaimilla, sekä varjostettu tummemmalla sinisellä, jonka värikoodi on #003547.

Sekä valikko, että hakupainike ovat myös pyöristettyjä. Painike myös muuttaa käyttäjän kursorin sormen malliseksi, samalla tavalla kuin esimerkiksi linkin päälle viettäessä. Tällä pyritään välittämään loppukäyttäjälle viesti siitä, että molemmat elementit ovat käytettäviä painikkeitä. Nämä visuaaliset tehosteet ovat toteutettu CSS-attribuuteilla:

```
border-top-left-radius: 10px;
border-top-right-radius: 10px;
border-bottom-left-radius: 10px;
border-bottom-right-radius: 10px;
cursor: pointer;
```

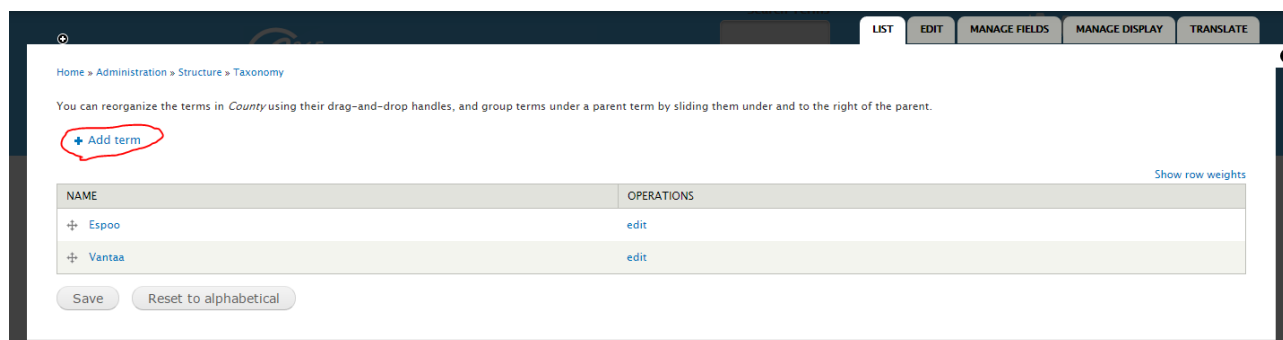
Painikkeen ja sen kuvauksen ympärille on myös asetettu “padding” -attribuutilla kymmenen pikselin kokoinen rako muusta sisällöstä erottumisen vuoksi.

## 5 Sivuston laajentaminen uusille paikkakunnille

Tärkeä arvo Seniori-365 sivustoa laajennettaessa oli sen skaalautuvuus eli mahdollisuus jatkokehittää ja laajentaa sivustoa lisäämällä esimerkiksi uusia paikkakuntia. Kuntakohtaisen sisällön mahdollistaminen filterien avulla on kevyt ja selkeä ratkaisu sivustolle ja jatkokehittäjille. Sivuston rakenne ei myöskään muutu kyseisellä toteutuksella. Uusien kuntien lisääminen on helppoa ja niitä pystytään lisäämään useita. Päädyimme tähän toteutukseen, koska se on selkeä käyttäjälle ja jatkokehittäjälle.

Uuden kunnan lisääminen tapahtuu yksinkertaisesti lisäämällä County taksonomiaan uusi termi eli tässä tapauksessa kunta. Valitaan Drupalin hallintapaneelista Structure → Taxonomy → County ja kyseisestä taksonomiasta Add term. Termin nimeksi laitetaan lisättävän kunnan nimi. Nyt sisältöä luodessa tai muokatessa on mahdollista valita aikaisemmin luotu kunta.

Sivusto on kehitetty siten, että kun uusi termi lisätään County taksonomiaan, niin tämä termi tulee myös kunnan valinta pudotusvalikkoon ja pystytään sen perusteella etsimään sisältöä. Näin sivuston laajentaminen muille kunnille on jatkokehittäjän kannalta helppoa. Seuraavaksi tulee määritellä sisältöä, jotka kuuluvat kyseiseen kuntakategoriaan.



Kuva 22: Uuden termin lisääminen County taksonomiaan

## 6 Yhteenveto

Kokonaisuudessaan projekti onnistui hyvin, mutta alussa ilmenneiden teknisten ongelmien ja niiden selvityksen vuoksi projektin valmistuminen viivästyi odotetusta aikataulusta. Kommunikaatio toimeksiantajan ja sidosryhmien välillä toimi erinomaisesti

Sivustolla on muutosten valmistuttua helppokäyttöinen ja kevyt pohja uusien kuntien lisäämiselle. Tehdyt muutokset tukevat tehdyllä mallilla hyvin useidenkin paikkakuntien lisäämistä haittamatta silti sivuston käytettävyyttä tai rikkomatta siistiä ulkoasua. Projektin aikana ilmeni kuitenkin muutamia kehitysideoita, jolla sisällöntuottajien ja sivuston kehittäjien työtä voidaan jatkossa nopeuttaa.

Uusien kuntien, tuotteiden, palveluiden tai niiden tarjoajien tullessa mukaan palveluun, on heidän sisältönsä koskevat tiedot tällä hetkellä määritettävä manuaalisesti täyttämällä lomake. Tämä tarkoittaa sisällön lisääjän kannalta käytännössä sitä, että kymmenen yritystä, joista kukin tuo esille kymmenen tuotetta johtaa sadan lomakkeen täyttämiseen manuaalisesti.

Ongelman voisi yrittää ratkaista esimerkiksi moduulilla, jolla uusia sisältöjä voi luoda massana. Vaihtoehtoisesti yrityksille voisi mahdollistaa kuvausten ja muiden tietojen päivittämisen omiin sisältöihinsä.

Myös taksonomioiden muokkaaminen massana olisi hyvä saada mahdolliseksi esimerkiksi tarkoitukseen sopivan moduulin avulla. Lokalisoinnissa ongelmaksi muodostui se, että jokaiseen olemassa olevaan Espoon sisällön joutu määrittämään manuaalisesti Espoolle kuuluvaksi. Sama ongelma tulee uusia paikkakuntia lisättäessä toistumaan niiden sisältöjen kohdalla, jotka koskevat myös sivustolle lisättävää paikkakuntaa jo ennestään.

## Lähteet

Antti Laaksonen, 2011

[http://www.ohjelmointiputka.net/oppaat/opus.php?tunnus=php\\_01](http://www.ohjelmointiputka.net/oppaat/opus.php?tunnus=php_01)

Apache Friends

<http://www.apachefriends.org/index.html>

Atlassian

<https://bitbucket.org>

Drupal, 22.8.2016

<https://www.drupal.org/docs/7/organizing-content-with-taxonomies/organizing-content-with-taxonomies>

Drupal, 22.9.2016

<https://www.drupal.org/docs/develop/local-server-setup/windows/skype-interferes-with-xampp-and-wamp-on-port-80>

Drupal, 3.11.2016

<https://www.drupal.org/docs/7/understanding-drupal/content-types>

Oracle, 2016

<https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>

phpMyAdmin contributors, 2003-2016

<https://www.phpmyadmin.net/>

phpMyAdmin contributors, 2003-2016

<https://www.phpmyadmin.net/about/>

Suomen Drupal -yhdistys, 2014

<http://www.drupal.fi/>

Vangie Beal,

<http://www.webopedia.com/TERM/R/repository.html>

Wikipedia®, 3. helmikuuta 2015

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

WPBeginner LLC, 2009-2016

<http://www.wpbeginner.com/glossary/apache/>



Kuviot	
Kuva 1: Asennettavat XAMPP-komponentit .....	12
Kuva 2: XAMPP-ohjauspaneelistä käynnistettävät moduulit .....	13
Kuva 3: Git-lataus SourceTreessä .....	14
Kuva 4: Repositorion osoite .....	14
Kuva 5: Repositorion tuominen SourceTreehen .....	14
Kuva 6: Repositorion tuonnin määrittäminen .....	15
Kuva 7: Apache-juuren määrittäminen .....	15
Kuva 8: Esimerkki Apache-juuren määrittämisestä .....	16
Kuva 9: Polku, jossa Seniori365.fi-palvelun varmuuskopiot sijaitsevat .....	17
Kuva 10: PhpMyAdmin -konfiguraatiodoston avaaminen .....	18
Kuva 11: MySQL -konfiguraatiodoston avaaminen .....	19
Kuva 12: Structure valikon sijainti admin näkyvässä .....	22
Kuva 13: Lisäämämme County taksonomia-sanasto .....	23
Kuva 14: Lisätyt termit eli kunnat, joiden perusteella sisältöä tullaan kategorisoimaan ...	23
Kuva 15: County kentän lisääminen linkit sisältötyyppiin .....	24
Kuva 16: Kunnan valinta sisältöä luodessa .....	25
Kuva 17: Filtterin lisääminen kalenteriin .....	26
Kuva 18: Taksonomian haku .....	27
Kuva 19: Filtterin konfigurointi .....	27
Kuva 20: Käyttäjän näkymä, kun kunnaksi on valittu Vantaa .....	28
Kuva 21: Tapahtumakalenteri CSS-muutoksien jälkeen .....	29
Kuva 22: Uuden termin lisääminen County taksonomiaan .....	30