



LAUREA
AMMATTIKORKEAKOULU
Yhdessä enemmän

Verkkosivuston toteutus – Case Loimaan seudun kennelkerho Ry

Junttila, Petri

2015 Laurea



Laurea-ammattikorkeakoulu
Leppävaara

Verkkosivuston toteutus – Case Loimaan seudun kennelkerho Ry

Junttila Petri
Tietojenkäsittelyn koulutusohjelma
Opinnäytetyö
Joulukuu, 2015

Junttila Petri

Verkkosivuston toteutus – Case Loimaan seudun kennelkerho Ry

Vuosi 2015 Sivumäärä 24

Tämän toiminnallisen opinnäytetyön tavoitteena oli luoda responsiivinen verkkosivusto Loimaan seudun kennelkerho Ry -nimiselle yhdistykselle sekä tutkia Robot Framework testausautomaatiokehityksen soveltuvuutta verkkosivuston testaukseen.

Responsiivinen verkkosivusto on toteutettu Drupal-sisällönhallintajärjestelmällä, jonka avulla sisällön hallitseminen on helppoa ja vaivatonta. Opinnäytetyössä käydään läpi Drupal-sisällönhallintajärjestelmän yleisimpiä toiminnallisuuksia sekä sivuston toteutuksessa hyödynnettyjä työvälineitä, kuten XAMPP, HTML, CSS ja PHP. Tärkeänä osana opinnäytetyötä oli Robot Framework testausautomaatiokehityksen soveltuminen verkkosivuston testaamiseen. Opinnäytetyön tietoperusta perustuu jo saatavilla olevaan tietoon. Työssä käydään myös läpi kirjoittajan omia kokemuksia Drupalin ja Robot Frameworkin käytöstä.

Opinnäytetyön lopputuloksena syntyi helposti ylläpidettävä verkkosivu, jota yhdistys voi hyödyntää itsenäisesti omiin tarpeisiinsa. Toimeksiantaja antoi melko vapaat kädet sivuston toteutuksen suhteen pois lukien heidän valmiiksi määritellyt toiminnalliset vaatimukset. Sivuston toteutuksessa onnistuttiin täyttämään kaikki toimeksiantajan sivustolle antamat vaatimukset ja molemmat osapuolet olivat lopputulokseen todella tyytyväisiä. Robot Framework osoittautui myös erittäin hyväksi työkaluksi verkkosivuston testauksen automatisointiin.

Petri Junttila

Implementation of a Website with Drupal Content Management System – A Case study of Loimaan seudun kennelkerho Ry

Year	2015	Pages	23
------	------	-------	----

The purpose of this thesis was to create a website for an organization named Loimaan seudun kennelkerho Ry and to study the suitability of Robot Framework for testing website.

The website was implemented with open source content management system called Drupal. With Drupal managing website content is easy and effortless. This thesis focuses on the most common functionalities that Drupal uses and other tools that were used during the implementation such as XAMPP, HTML, CSS and PHP. Another important part of the thesis was to test the suitability of the Robot Framework for testing websites. The knowledge of this thesis is based on already available information. This thesis investigates the author's own experiences on Drupal and Robot Framework.

The result was a website that satisfies the client and is easy to maintain. During the implementation of a website client gave a lot of freedom on how to implement the website. The only restriction was functionalities that client required during the design phase. All the required functionalities were achieved during the implementation. The end result of this thesis was a website that pleased both the creator and the client. Also Robot framework proved to be a very useful tool for automate testing during the implementation phase.

Keywords website, drupal, robot framework, test automation

Sisällys

1	Johdanto.....	6
2	Toimeksiantajan esittely	7
3	Opinnäytetyön tavoite ja menetelmät	7
4	Työvälineet ja menetelmät	8
4.1	Drupal 7 -sisällönhallintajärjestelmä.....	9
4.1.1	Drupalin asentaminen	9
4.1.2	Moduulit.....	10
4.1.3	Teemat	10
4.2	Cascading Style Sheets (CSS)	11
4.3	Hypertext Markup Language (HTML)	12
4.4	PHP: Hypertext Preprocessor	12
4.5	MySQL	13
4.6	XAMPP.....	14
4.7	Robot Framework	15
5	Verkkosivuston toteutus	16
5.1	Suunnittelu.....	16
5.2	Toteutus	18
5.3	Testaus	19
6	Yhteenveto	21
	Lähteet	23

1 Johdanto

Tämän toiminnallisen opinnäytetyön tavoitteena on Loimaan seudun kennelkerho Ry:n verkkosivuston uusiminen. Osana opinnäytetyötä selvitettiin myös testausautomaation käyttämistä osana verkkosivuston testausta. Loimaan seudun kennelkerho toimii aktiivisesti Loimaan seudulla ja verkkosivut ovat tärkeä väylä informaation jakamiseen jäsenille ja muille yhdistyksen toiminnassa mukana oleville. Yhdistyksen verkkosivut löytyvät osoitteesta: <http://www.lskennelkerho.net/>

Yhdistykselle on aiheuttanut suuria haasteita nykyisen verkkosivuston sisällön hallinta. Tämän vuoksi aloitimme miettiä, miten voisimme korjata tämän ongelman. Keskusteluiden perusteella päätimme toteuttaa sivuston käyttämällä sisällönhallintajärjestelmää. Päädyimme toteuttamaan verkkosivuston Drupal-sisällönhallintajärjestelmällä. Drupalin käyttäminen ei tarvitse välttämättä ohjelmoinnin osaamista. Tämä oli yksi tärkeimmistä syistä Drupaliin päättymiseen. Drupalin avulla yhdistys ei tarvitse ulkopuolista osaamista sivuston sisällön ylläpitämiseen.

Opinnäytetyön kirjallinen osa tarjoaa keskeisimmät tiedot Drupal -sisällönhallintajärjestelmästä ja sen toiminnallisuuksista. Kirjallisessa osassa on kuvattu myös opinnäytetyön produktin toteutuksen keskeisimmät vaiheet, sekä tutustuttu Robot Framework testausautomaatiokehityksen soveltavuuteen verkkosivuston testauksessa.

Aloitin projektin suunnittelemalla, miten toteutan sivuston hallinnan ja ylläpidon käyttäjille, joilla ei ollut juurikaan kokemusta verkkosivujen toteuttamisesta tai hallinnasta. Julkaisujärjestelmien avulla voisin toteuttaa sivuston, jonka ylläpito ei vaatisi suurempaa osaamista verkkosivustojen toteuttamisesta. Julkaisujärjestelmän avulla uusien sivujen lisääminen sivustolle olisi helppoa ja vaivatonta. Toimeksiantajalla oli selvä käsitys siitä, minkälaisia toimintoja sivustolle tarvittaisiin. Näin ollen pääosin määrittelytyö sivustoa varten oli jo valmiiksi tehtynä.

Jo valmiiksi tehtyjen määrittelyiden perusteella työn suunnittelun aloittaminen oli sujuvaa. Saatuaani toimeksiantajalta määrittelyt vaadituista toiminnallisuuksista, pystyin aloittamaan verkkosivuston suunnittelun. Julkaisujärjestelmistä päädyin Drupaliin, koska aiempien kokemusten perusteella se oli riittävän helppokäyttöinen. Drupal on myöskin teknisesti laadukas, joten en nähnyt tarvetta tehdä tarkempaa teknistä vertailua eri ohjelmistojen välillä tätä työtä aloittaessani.

2 Toimeksiantajan esittely

Loimaan seudun kennelkerho Ry on vuonna 1950 perustettu koirakerho. Yhdistys toimii aktiivisesti Loimaan seudulla. Kerho järjestää vuosittain match show -tapahtumia ja luentoja. Jäsenilleen yhdistys tarjoaa myös koulutusta ja epävirallisia kilpailuja.

Loimaan seudun kennelkerho Ry:llä on ollut tarkoituksena jo pidempään uudistaa internet -sivustoaan, koska vanha sivusto ei enää vastannut yhdistyksen tarpeita. Vanhan sivuston päivittävyys on ollut liian haastavaa useille sivuston tämän hetkisistä ylläpitäjistä.

Tämän toiminnallisen opinnäytetyön tuotoksena on kehitetty yhdistykselle uusi helpommin ylläpidettävä verkkosivusto, joka tukee heidän toimintaansa huomattavasti aiempaa sivustoa paremmin. Opinnäytetyön sivutuotoksena tuotetaan myös ylläpitodokumentti, jonka avulla yhdistys pystyy ylläpitämään sivustoa. Sivusto on toteutettu yhteistyössä yhdistyksen kanssa.

3 Opinnäytetyön tavoite ja menetelmät

Opinnäytetyön tavoitteena on toteuttaa Loimaan seudun kennelkerho Ry:lle helposti ylläpidettävä responsiivinen verkkosivusto, jonka avulla yhdistys voi jakaa tietoa jäsenilleen sekä muille yhdistyksen toiminnasta kiinnostuneille. Responsiivisuus on tärkeä ominaisuus sivustolle, koska nykypäivänä monet käyttäjät käyttävät sivustoa mobiililaitteilla. Juurikin ylläpidon helppouden vuoksi päädyimme Drupal-sisällönhallintajärjestelmään. Yhdistyksen jäsenillä ei ole juurikaan kokemusta verkkosivuston ylläpitämisestä, joten opinnäytetyön sivutuotoksena laaditaan ylläpitodokumentti, jonka avulla yhdistys voi itse päivittää ja luoda verkkosivuston sisältöä. Yhdistyksellä oli valmiina tehtyt vaatimusmäärittelyt uudelle sivustolle, joten näihin ei tarvinnut suurempaa panostusta tehdä suunnitellessa sivustoa. Kävimme nämä vaatimukset läpi toimeksiantajan kanssa heti projektin ensitoiksi. Tämä auttoi saamaan hieman kuvaa tulevan työn määrästä. Opinnäytetyössä tutustutaan myöskin testausautomaation hyödyntämiseen verkkosivuston testauksessa. Sivuston testauksessa hyödynnettiin Robot Framework testausautomaatiokehystä.

Opinnäytetyö toteutetaan toiminnallisena opinnäytetyönä. Toiminnallisen opinnäytetyön lopullisena tuotoksena syntyy aina jokin konkreettinen tuote. (Vilka & Airaksinen 2003, 51.) Tämän opinnäytetyön produktina syntyy toteutettava verkkosivusto. Tämän opinnäytetyön produktin lisäksi tuotetaan raportti, jossa kuvataan opinnäytetyön prosessia ja siinä käytettyjä työvälineitä.

Itsellä tavoitteena tämän opinnäytetyön aikana on kasvattaa kokemusta Drupal -sisällönhallintajärjestelmästä sekä Robot Frameworks testausautomaatiokehiksestä. Toivon myös tämän työn avulla saavuttavani paremman tuntemuksen yleisesti verkkosivujen toteutuksesta. Drupaliin olen tutustunut hieman jo ennen opinnäytetyön aloittamista, mutta koskaan en ole kokonaisia verkkosivuja sillä toteuttanut. (Vilkkä & Airaksinen 2003, 65.)

4 Työvälineet ja menetelmät

Sivuston toteutuksessa on käytetty Drupal-sisällönhallintajärjestelmää. Sisällönhallintajärjestelmillä (CMS) tarkoitetaan järjestelmiä, joiden avulla voidaan luoda Internet -sivustoja ilman ohjelmoinnin osaamista. Ohjelmoinnin osaamisen avulla voidaan kuitenkin hallita sivustoja paljon monipuolisemmin. Lyhenne CMS tulee englannin kielen sanoista Content Management System. (Shreves & Dunwoodie 2011, 4.) Sisällönhallintajärjestelmän avulla voidaan luoda yksinkertaisia verkkosivustoja, mutta yleisesti sisällönhallintajärjestelmät tarjoavat todella toimivan rakenteen helposti ylläpidettävän sivuston luomiseen.

Julkaisujärjestelmiä on useita erilaisia, joissa on todella paljon erilaisia ominaisuuksia. Tämän vuoksi, ennen sisällönhallintajärjestelmän valitsemista tulee tarkkaan harkita minkälaisia ominaisuuksia CMS-järjestelmältä vaaditaan. Sisällön tuottaminen on toisilla järjestelmillä monimutkaista, kun taas osassa rajallisuus saattaa asettaa rajoituksia haluttuihin toimintoihin. CMS-järjestelmää valittaessa tuleekin ottaa huomioon asiakkaan osaamistaso ja sivuston tarpeet.

Avoimeen lähdekoodiin perustuvista ilmaisista CMS -järjestelmistä yleisimmin käytettyjä ovat Drupal, Joomla, Wordpress. Parhaiten nämä järjestelmät soveltuvatkin pienille yrityksille ja yhdistyksille, joilla ei ole taloudellisesti mahdollisuuksia mainostoimiston tai muun toteuttajan käyttämiseen.

Drupalissa käytetään ulkoasun luomiseen CSS- ja HTML-merkintäkieliä. Näiden avulla voidaan luoda Drupal -sivustoille omia teemoja. Teemoilla tarkoitetaan Drupal -sivun ulkoasua. Valmiita teemoja on ladattavissa Drupal.org -sivustolta. Tässä opinnäytetyössä valitsimme valmiin teeman, jota muokkasimme yhdistyksen tarpeisiin sopivaksi. Tässä kappaleessa käydään läpi toteutuksessa käytettyjä keskeisimpiä työvälineitä ja merkintäkieliä, sekä niiden keskeisimpiä ominaisuuksia. Näiden avulla opinnäytetyö saatiin toteutettua toimeksiantajan vaatimukset täyttävä verkkosivu. (Shreves & Dunwoodie 2011, 6.)

4.1 Drupal 7 -sisällönhallintajärjestelmä

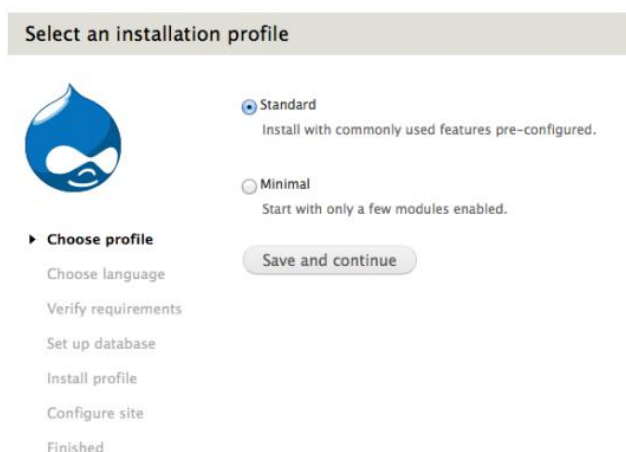
Drupal on yksi avoimeen lähdekoodiin perustuvista sisällönhallintajärjestelmistä. Drupalin on kehittänyt Dries Buytaert vuonna 2000. Se on yksi kolmesta yleisimmin käytetyistä sisällönhallintajärjestelmistä. (Shreves & Dunwoodie 2011, 10.) Drupalilla on todella laaja verkkoyhteisö, jossa kehitetään koko ajan uusia ominaisuuksia järjestelmään. Siihen on olemassa tuhansia ilmaisia ominaisuuksia, joita voidaan hyödyntää verkkosivuston toteutuksessa. Drupal on toteutettu PHP-ohjelmointikielellä ja sen vahvuuksia ovat laajennettavuus ja runsas aktiivisten käyttäjien määrä. Drupalista on myös saatavilla lisensoitu maksullinen versio, jonka avulla voidaan vaikuttaa enemmän tekniikoihin ja toimivuuteen. (Burge & McCourt 2013, 34.)

4.1.1 Drupalin asentaminen

Drupalin asentaminen asettaa tiettyjä vaatimuksia, jotta asentaminen onnistuu sujuvasti. Asennuksen vaatii vähintään 15 megatavua levytilaa, mutta käytännössä tämä nousee noin sataan megatavuun. Levytilaa syövät perusasennuksen lisäksi Drupalin moduulit ja teemat, joilla voidaan laajentaa Drupalin perustoimintoja. On hyvä pitää mielessä, että myös tietokanta ja palvelimen muut tiedostot, kuvat ja varmuuskopiot vaativat paljon levytilaa.

Web-palvelin vaatii myös Apache tuen, jotta Drupal voidaan asentaa. Drupal vaatii tietokannan toimiakseen, koska lähes kaikki tieto tallennetaan tietokantaan. Viimeisimmät Drupal versiot tukevat MySQL-, PostgreSQL- ja SQLite -tietokantoja. (Drupal.org 2015)

Ennen asennusta palvelimelle tulee luoda tietokanta, johon Drupal-sivusto tallentaa tiedot. Asennuspaketin voi ladata suoraan Drupal.org -sivustolta. Asennuspaketissa tulee joukko perusmoduuleita, joiden avulla yksinkertainen sivusto voidaan luoda. Näitä kutsutaan ydinmoduuleiksi. Asennuspaketti puretaan palvelimelle, jonka jälkeen se voidaan asentaa ajamalla `install.php` -komento. Komento ajetaan menemällä selaimella `http://omadomain/drupal/install.php` -osoitteeseen. Tämän jälkeen asennusohjelma alkaa asentaa Drupalia. Asennusohjelman ohjeita seuraamalla asennus on todella helppo tehdä ilman suurempaa osaamista.



Kuva 1: Drupal asennusohjelman aloitusruutu

Tässä työssä käytettiin toteutuksen aikana virtuaaliympäristöä, joka luotiin XAMPP -ohjelmiston avulla lokaalisti omalle tietokoneelle. Näin ei tarvinnut tehdä kehitystä suoraan palvelimella. Tämä helpotti työskentelyä, koska verkkoyhteys ei ollut pakollinen verkkosivustoa toteuttaessa.

4.1.2 Moduulit

Moduulit ovat yksi keskeisimmistä toiminnallisuuksista Drupalissa. Moduuleilla tarkoitetaan Drupaliin asennettavia työkaluja, joiden avulla voidaan hallita sivuston käytettävyyttä ja sisällön tuotantoa. Valmiina asennuksessa olevilla moduuleilla, joita kutsutaan ydinmoduuleiksi, voidaan toteuttaa useat verkkosivuston yleisimmistä toiminnoista. Drupal.org -sivustolta löytyy tuhansia Drupalin käyttäjien toteuttamia moduuleita, joiden avulla saadaan lisättyä sivustolle toimintoja. Tämän vuoksi sivustoa suunnitellessa tuleekin tarkkaan miettiä, mitä toimintoja sivustolla tullaan tarvitsemaan. PHP- tai Javascript-ohjelmointikielillä voidaan toteuttaa myös omia moduuleita tai muokata jo olemassa olevia moduuleita omaan käyttöön paremmin soveltuviksi. (Shreves & Dunwoodie 2011, 11.)

4.1.3 Teemat

Drupalissa teemalla tarkoitetaan sivuston visuaalista ilmettä eli ulkoasua. Teeman asetuksilla voi vaikuttaa sivuston fontteihin sarakkeiden määrään, sivuston ulkoiseen näköön, painikkeiden sijainteihin sekä sivustolla navigointiin. Teemat koostuvat yhdestä tai useammasta PHP-tiedostosta, jotka määrittelevät sivuston rakenteen HTML-kielenä, jonne haetaan sisältö tietokannoista PHP-kielillä. Drupalin perusasennuspaketissa tulee viisi teemaa, mutta jo valmiita teemoja löytyy todella paljon Drupalin kotisivuilta. (Shreves & Dunwoodie 2011, 485.)

Drupalissa teema on irroitettu sivuston toiminnallisuudesta, minkä vuoksi Drupalin teemajärjestelmä on todella joustava. Tämän vuoksi käytännössä sivuston toiminnallisuudet voidaan tehdä erilläänkin eri henkilöiden toimesta. (Shreves & Dunwoodie 2011, 486.)

4.2 Cascading Style Sheets (CSS)

Cascading Style Sheets on verkkosivustoille kehitetty tyyliohjeiden laji. CSS-kielen avulla voidaan määritellä kuinka HTML-elementit näytetään selaimessa. CSS-kielellä määritetään myös esimerkiksi miten kuvat, tekstit ja tekstien sijoittelut näytetään selaimessa (Shreves & Dunwoodie 2011, 499). Kun tyyli on määritelty valmiiksi CSS-tiedostoon, varmistetaan myös ettei käyttäjä vahingossa vaihda väärää fonttia, kuvien asetuksia tai linkkien asetuksia sivuston muista käytännöistä poiketen. CSS-tyylit voidaan sisällyttää myös HTML-tiedostoon, mutta yleisesti kannattaa CSS-tyylit toteuttaa omaan tiedostoon. Näin voidaan välttyä liian sekavilta ja suurilta HTML -tiedostoilta. CSS-tiedostoja voidaan myös luoda useampia samalla sivustolle. Yleensä näin tehdään, koska eri selaimet saattavat näyttää sivuston hieman eri tavoilla. CSS:n ja HTML:n erottaminen toisistaan mahdollistaa helpomman sivuston ylläpidon ja samojen tyylien käyttäminen eri puolilla sivustoa on helpompaa. Käytännössä tämä mahdollistaa sivuston rakenteen erottamisen sivuston muusta sisällöstä.

```

/* HTML5 display-role reset for older browsers */
article, aside, details, figcaption, figure,
footer, header, hgroup, menu, nav, section {
  display: block;
}

ol, ul {
  list-style: none;
}

table {
  border-collapse: collapse;
  border-spacing: 0;
}

/*
  Clear Floated Elements
*/

.clear {
  clear: both;
  display: block;
  overflow: hidden;
  visibility: hidden;
  width: 0;
  height: 0;
}

.clear:after {
  clear: both;
  content: ' ';
  display: block;
  font-size: 0;
  line-height: 0;
  visibility: hidden;
  width: 0;
  height: 0;
}

* html .clear {
  height: 1%;
}

```

Kuva 2: Drupal-sivuston teeman käyttämä Style.css -tyylitiedosto

Myös Drupalissa CSS-tyylit määräävät teemojen tyylit. Mikäli Drupal-teemoja halutaan muokata omiin tarpeisiin, olisi hyvä tuntea jonkin verran CSS -kieltä. Kuitenkin lähes olemattomilla tiedoilla pystyy tekemään todella suuria muutoksia sivuston ulkoasuun. (Burge & McCourt 2013, 500.)

4.3 Hypertext Markup Language (HTML)

Hypertext Markup Language on merkintäkieli, jonka avulla voidaan yhdistää informaatio ja tiedostot keskenään. HTML on yksi yleisimmin käytetyistä kielistä Internetin julkaisuissa. Verkkosivusto koostuu isosta määrästä erilaisia tiedostoja, kuten kuvista, tietokannoista, tekstistä, tyyli pohjista jne. HTML avulla nämä saadaan koottua toimivaksi kokonaisuudeksi selaimen ymmärtämässä muodossa. Eri selaimet tulkitsevat HTML-kieltä mahdollisimman samalla tavalla, jotta HTML-kieltä laatiessa ei tarvitsisi ottaa huomioon eri selaimia.

```
<html>
  <head>
    <title></title>
  </head>
  <body>
    ...
  </body>
</html>
```

Kuva 3: HTML -sivun perusrakenne

HTML -sisältää elementtejä, jotka erotellaan toisistaan tagien avulla. Ensimmäisenä tulee aina alkutagi <>, jonka jälkeen esimerkiksi verkkosivuilla näkyvä teksti. Elementti lopetetaan aina </> tagilla. Alku- ja lopputagin sisällä on aina sama elementin nimi, joka kertoo miten selaimen tulee tagien välissä oleva elementti esittää. HTML-sivulla on aina sama perusrakenne. Kuvassa 3 näytetään esimerkki HTML-sivun perusrakenteesta.

4.4 PHP: Hypertext Preprocessor

PHP on avoimen lähdekoodin ilmainen skripti- eli komentosarjakieli, jonka avulla voidaan automatisoida tehtäviä ilman tarvetta erilliselle ohjelmointikielille. PHP upotetaan yleensä HTML-koodin sekaan. PHP-koodi ei näy suoraan sivuston käyttäjälle, vaan se suoritetaan suoraan palvelimella. PHP:n syntaksiin on lainattu paljon C-, Java- ja Perl -ohjelmointikielistä. PHP omaa kuitenkin myös täysin omia ominaisuuksia. PHP:n kehityksen taustalla on tavoite pystyä luomaan dynaamisia verkkosivuja mahdollisimman nopeasti ja helposti.

```

<?php
/**
 * @file
 * Contains theme override functions and preprocess functions for the theme.
 */

/**
 * Implements hook_html_head_alter().
 */
function responsive_green_html_head_alter(&$head_elements) {
  $head_elements['system_meta_content_type']['#attributes'] = array(
    'charset' => 'utf-8'
  );
}

/**
 * Insert themed breadcrumb page navigation at top of the node content.
 */
function responsive_green_breadcrumb($variables) {
  $breadcrumb = $variables['breadcrumb'];
  if (!empty($breadcrumb)) {
    // Use CSS to hide title .element-invisible.
    $output = '<h2 class="element-invisible">' . t('You are here') . '</h2>';
    // Comment below line to hide current page to breadcrumb.
    $breadcrumb[] = drupal_get_title();
    $output .= '<nav class="breadcrumb">' . implode(' » ', $breadcrumb) . '</nav>';
    return $output;
  }
}

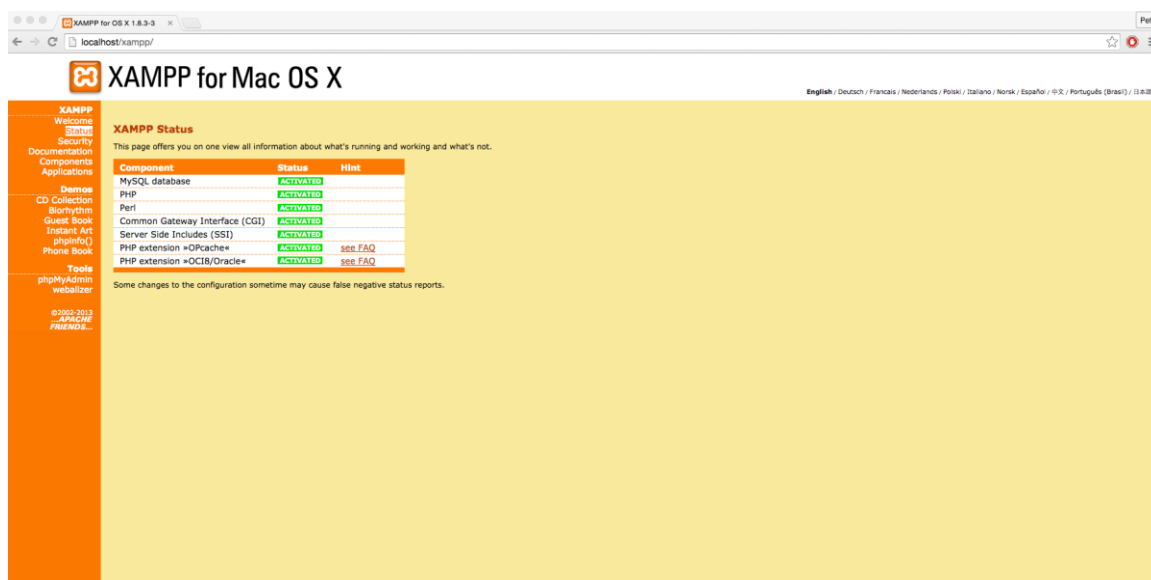
```

Kuva 4: Esimerkki Drupalin käyttämästä, teeman ulkoasuun liittyvästä, PHP -tiedostosta

PHP-lyhenne muodostuu sanoista PHP: Hypertext Preprocessor. Käyttäjämäärien ja suosion kasvettua alkuperäinen nimi on vanhentunut ja saanut nykyisen muotonsa (PHP) kielen ominaisuuksien kehittyessä. PHP on nykypäivänä käytössä noin 244 miljoonalla sivustolla. (The PHP Group 2015.)

4.5 MySQL

MySQL on avoimen lähdekoodin relaatiotietokantaohjelmisto, joka luotu vuonna 1995. Sen omistaa tällä hetkellä Oracle Corporation. Alun perin MySQL:n loi ruotsalainen MySQL AB. Yhtiö vastaa edelleen ohjelmiston kehityksestä. MySQL toimii useiden eri ohjelmointikielien kanssa, kuten PHP, C, C++, Perl, Python ja Ruby. MySQL on ollut suosittu helppoutensa ja hintansa vuoksi. Myös PHP:n suosio on lisännyt MySQL:n käyttöä, koska se on helposti käytettävissä PHP:n kanssa. (MySQL.com 2015)



Kuva 6: XAMPP -komponenttien tilat selainkäyttöliittymällä.

4.7 Robot Framework

Robot Framework on avoimen lähdekoodin testiautomaatiokehys, joka kehitetty etenkin hyväksymistestaukseen ja testivetoiseen kehitykseen. Robot Frameworkin on kehittänyt Pekka Klärck Nokia Siemens Networkille vuonna 2005. Ensimmäinen open source -versio julkaistiin vuonna 2008. Robot Framework voidaan asentaa kaikkiin yleisimpiin käyttöjärjestelmiin (Larman & Vodde 2010).

Robot Framework on kehitetty Pythonilla, mutta se tukee myös Jythonia, joka on Python -kielellä toteutettu Java-toteutus. Robot Framework hyödyntää avainsanaohjatun testauksen menetelmiä. Sen käyttämä taulukkorakenteinen syntaksi mahdollistaa testien kirjoittamisen helposti luettavissa taulukkomuodoissa, kuten tekstimuotoisissa tai HTML -kielellä luoduissa taulukkomuodoissa. Myös aineisto-ohjattujen testien kirjoittaminen on Robot Frameworkilla mahdollista. (Robotframework.org 2015.)

Robot Frameworkiin on kehitetty paljon valmiita testikirjastoja, joita voidaan hyödyntää testauksessa. Yleisimmät testikirjastot kuuluvat Robot Frameworkin asennuspakettiin. Testikirjastoja voidaan luoda myös itse käyttämällä Python- tai Java-ohjelmointikieliä. Myös erillisten projektien toteuttamia testikirjastoja on ladattavissa Robot Frameworkin internet -sivustolta. (Robotframework.org 2015.)

Testejä voidaan kirjoittaa useilla eri tavoilla. Ne kuitenkin määritellään aina taulukkomaisesti jäseneltyinä. Testit voidaan luoda seuraavissa tiedostomuodoissa: HTML, TSV (TAB Separated

Values), reST (reStructuredTEXT) tai tekstitiedosto (.txt). Alla olevassa kuvassa esimerkki .txt -muotoisesta testitapauksesta.

```

*** Settings ***
Resource          resource.txt

*** Testcases ***
Browser should open and close
  Open Browser   http://localhost/drupal-7.26/login   safari
  Set Selenium speed      1
  Maximize Browser Window
  Input text      name      admin
  Input text      pass      admin
  Click button    Ok
  Close Browser

```

Kuva 7: Esimerkki tekstitiedostoon tehdystä avainsana ohjatusta testitapauksesta.

Robot Framework -testit perustuvat avainsanoihin, joiden avulla voidaan ohjata testejä sekä vertailla muuttujia ja muokkauksia. Kun avainsana suoritetaan, palautuu siitä aina tieto siitä, onnistuiko avainsanan suoritus tai mikä oli vertailun tulos. Robot Frameworkissa on paljon sisäänrakennettuja avainsanoja, joiden avulla erilaisten muuttujien ja datan vertailu sekä testiajon ohjaus. Robot Frameworkin asennuksen mukana tulee myös oma integroitu testiympäristö Ride, joka on suunniteltu testien kirjoittamiseen ja suorittamiseen. Riden avulla voidaan helposti tarkastaa testien kokonaiskuva. Riden avulla testien ylläpito on myös monesti helpompaa. (Larman & Vodde 2010).

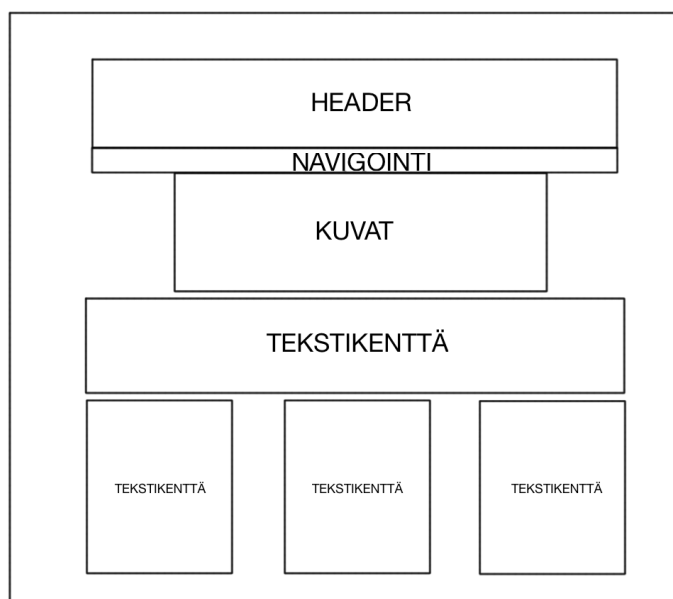
5 Verkkosivuston toteutus

Sivuston toteutus koostui seuraavista vaiheista suunnittelu, toteutus ja testaus. Tässä kappaleessa käydään läpi toteutusprosessia ja sen eri vaiheita. Testaus osiossa on erityisesti keskitytty toteutuksen yhteydessä toteutettuun testausautomaatioon, jonka avulla on pystytty ylläpitämään sivuston toimivuutta ja toteutuksen laatua koko projektin ajan.

5.1 Suunnittelu

Kun aloitetaan suunnittelemaan verkkosivustoa, on joitakin asioita jotka tulee ottaa huomioon. Näitä ovat verkkosivujen yhtenäisyys, verkkosivujen ulkoasu, typografia, visuaalinen ilme, verkkosivujen rakenne, navigointi, sekä sivuston käytettävyys. Verkkosivuston suunnittelu muistuttaa jonkin verran lehden suunnittelua. Molemmissa pyritään typografiseen yhtenäisyyteen. Verkkosivuilla on järkevää käyttää yhtenäistä tyyliä kaikilla sivuilla. Yhtenäistä tulisi olla ainakin taustakuva, värit, fontit, logo ja vakiolinkit. (Korpela & Linjama 2005, 48.)

Jesse James Garrettin mukaan verkkosivujen suunnitteluun suositellaan viisiportaista mallia. Ensin täytyy määritellä sivuston strateginen rooli, jolla käytännössä tarkoitetaan sivuston olemassaolon perimmäistä tarkoitusta. Toisessa vaiheessa suunnitellaan sivuston sisältö ja sivuston toimivuus, jonka jälkeen kolmannessa vaiheessa suunnitellaan sivuston rakenne. Neljännessä vaiheessa suunnitellaan sivuston navigaatio ja käyttöliittymä. Tällä tarkoitetaan sitä, miten käyttäjä voi sivustolla liikkua.



Kuva 8: Loimaan seudun kennelkerhon verkkosivuston etusivun rautalankamalli.

Useasti tässä vaiheessa tehdään ns. rautalankamalli sivustosta. Kuvassa kahdeksan esitetään Loimaan seudun kennelkerhon verkkosivuston etusivun rautalankamalli. Viimeisenä vaiheena on sivuston visuaalisuuden suunnittelu. Tässä vaiheessa määritellään kaikki tarvittavat elementit valokuvista fontteihin ja väreistä muihin graafisiin elementteihin. (Juslén 2011, 80.)

Kun aloimme suunnitella sivustoa yhdessä toimeksiantajan kanssa, mietimme ensin ketkä ovat sivuston ensisijaisia käyttäjiä. Sivuston pääasialliset käyttäjät ovat yhdistyksen jäsenet sekä muut yhdistyksen toiminnasta kiinnostuneet, kuten yhdistyksen järjestämiin kilpailuihin osallistuvat henkilöt. Tämän vuoksi olisi tärkeää, että esimerkiksi yhdistyksen järjestämät kilpailut löytyisivät sivustolta helposti.

Sivuston sisältö ja rakenne tulisi pysymään lähes aiemman kaltaisena, joten tähän ei tarvinnut laittaa suurta panosta suunnittelun aikana. Yksi isoimmista ongelmista vanhalla sivulla oli sivuston responsiivisuuden puute. Responsiivisella verkkosuunnittelulla tarkoitetaan sitä, että

toteutuksessa otetaan huomioon käyttäjän laite ja näyttökoko (Jehl 2014, 5). Tämä on tärkeää ottaa huomioon, koska nykyään verkkosivuja käytetään paljon mobiililaitteilla.

Nimenomaan responsiivisuuden vuoksi, päätimme ottaa sivuston ulkoasun pohjaksi Responsive Green -nimisen teeman (https://www.drupal.org/project/responsive_green). Tästä teemasta saimme muokattua yhdistyksen tarpeiden mukaisen ulkoasun sivustolle. Tässä teemassa on jo aiemmin otettu erityisesti huomioon sivuston responsiivisuus. Muutenkin sivuston helppokäyttöisyys ja selkeys nousi tärkeäksi kriteeriksi sivuston ulkoasun toteutukselle. Toimeksiantajalla oli toiveena myös pitää sivuston värimaailma entisen kaltaisena, eli vihreän sävyisenä. Tässä vaiheessa tein toimeksiantajalle rautalankamallin tulevan sivuston ulkoasusta.

Kun olimme päässeet yhteisymmärrykseen sivuston ulkoasusta, kävimme läpi muita toimeksiantajan asettamia vaatimuksia sivustolle. Pääasiassa nykyisellä sivulla on jo olemassa kaikki tarvittavat toiminnallisuudet, mutta niiden ylläpitäminen on liian vaativaa yhdistyksen jäsenille.

Toiveena oli myös, että sivustolle toteutettaisiin oma keskustelufoorumi. Tällä hetkellä sivustolla on erillinen keskustelufoorumi sovelluksensa, jota käyttäjät ovat käyttäneet. Tutkiskeltuani vaihtoehtoja keskustelufoorumin toteutukseen huomasin, että nykyisin käytössä oleva keskustelufoorumi on toiminnoiltaan huomattavasti kattavampi kuin Drupaliin kehitetyt keskustelufoorumi -moduulit. Tämän vuoksi päätimme jatkaa vanhan ulkoisen keskustelufoorumin käyttöä.

Kun kaikki suunnitteluvaiheen toimenpiteet oli käyty läpi, pääsin aloittamaan itse sivuston toteutuksen. Sivuston suunnittelun pohjalta toteutusvaiheeseen siirtyminen oli helppoa, koska kaikki tarvittavat toiminnallisuudet ja ulkoasuun liittyvät tiedot oltiin sovittu asiakkaan kanssa yksityiskohtaisesti.

5.2 Toteutus

Toteutusvaiheen aluksi tarvitsin kehitysympäristön, jossa voisin tehdä sivuston toteutuksen. Aiempien kokemusteni perusteella päädyin luomaan kehitysympäristön XAMPP -ohjelmistoa hyödyntäen. Tässä vaiheessa tarvitsin kehitysympäristön, koska yhdistyksen nykyinen verkkosivu haluttiin pitää verkossa koko toteutuksen ajan. XAMPP-ohjelmiston avulla pystyin luomaan paikallisen www-palvelimen, jonka avulla pystyin toteuttamaan sivustoa siten, ettei sitä tarvinnut siirtää oikealle palvelimelle. XAMPP-ohjelmiston avulla pystyin myös luomaan Drupalin vaatimat tietokannat.

Kun kehitysympäristö oli saatu valmiiksi, pystyin aloittamaan itse sivuston toteuttamisen. Ensimmäiseksi täytyi asentaa Drupal -sisällönhallintajärjestelmä. Kun Drupal oli saatu asennettua, pystyin aloittamaan sivuston ulkoasun toteuttamisen. Kuten edellisessä kappaleessa mainittiin, ulkoasun toteuttamisen pohjaksi valittiin Drupaliin valmiiksi toteutettu teema Responsive Green. Tämä helpottaa huomattavasti ulkoasun toteuttamista, koska tällöin ulkoasua ei tarvitse lähteä toteuttamaan alusta asti itse. Ulkoasun toteutuksessa muokattiin sivuston värejä, fontteja ja rakennetta. Sivuston rakenne haluttiin pitää selkeänä, joten osa teeman mukana tulleista rakenteista jätettiin pois, jotta kokonaisuus saatiin pidettyä yksinkertaisena.

Drupalin ydinmoduuleilla ei saanut kaikkia sivustolle vaadittuja toimintoja toteutettua. Tämän vuoksi, jotta sain kaikki vaaditut toiminnot sivustolle, joudun laajentamaan Drupalin perustoimintoja uusilla moduuleilla. Yksi uusista moduuleista joka asennettiin oli Colorbox. Tämän moduulin avulla sain sivustolle luotua toimeksiantajan vaatiman kuvagallerian. Colorboxin avulla käyttäjä pystyy selaamaan kuvia helposti suoraan sen hetkisen sivun päällä.

Toimeksiantajalla oli myös tarvetta tapahtumakalenterille, jonka sain helposti toteutettua ilman uusia moduuleita. Entinen tapahtumakalenteri oli luotu Google Calendarin avulla, jonka sai upotettua suoraan Drupalin sisältösivulle.

Yksi tärkeimmistä Drupalin lisämoduuleista on Views. Views -moduulin avulla voidaan esittää sivuston sisältöä dynaamisesti sivuna, rss -syötteenä tai monissa muissa formaateissa. (Drupal.org/views) Tällä sivustolla käytin Views -moduulia dynaamisesti päivittyvään listaan, jonka avulla loin listan uusimmista uutisista etusivulle. Views -moduulin konfigurointi aiheutti aluksi hieman haasteita, koska sen avulla voidaan suodattaa tai järjestellä esitettävää sisältöä lähes minkä muuttujan avulla tahansa.

Sivuston ylläpidon tuli olla käyttäjilleen helppoa, joten yksi tärkeimmistä ominaisuuksista, joka täytyi lisätä, oli monipuolisempi tekstieditori. Koska sivuston tulevat sisällönhallitsijat eivät osaa juurikaan ohjelmointia, tuli tähän tilanteeseen apuun Wysiwyg -moduuli. Wysiwyg on Drupalin käyttämä moduuli, joka ei itsessään toimi tekstieditorina vaan antaa tuen erilaisille tekstieditoreille. Tälle sivustolle valitsin editoriksi TinyMCE, joka on omien kokemuksieni mukaan yksi suosituimmista ja kattavimmista tekstieditoreista Drupalissa.

5.3 Testaus

Ohjelmistotestaus on yksi ohjelmistotuotannon tärkeimmistä osista. Ohjelmistotestauksella pyritään varmistamaan toteutettavan sovelluksen laatu, ja että kaikki sovellukselle toteutetut ominaisuudet vastaavat asiakkaan määrittämiä. Testauksen avulla voidaan siis tunnistaa ne kohdat, joissa tuotos eroaa suunnitelmasta (Kasurinen 2013, 10).

Sivuston toteutuksen alussa päätettiin hyödyntää testausautomaatiota sivuston testauksessa. Testausautomaatiolla tarkoitetaan sellaista testauksen muotoa, jossa ohjelmiston testaamista varten rakennetaan automaatiotyövälineitä testien tekemistä varten. Tämä onkin yksi testausautomaation suurimmista haasteista, koska testitapausten rakentaminen voi olla lähes yhtä vaativaa kuin itse järjestelmän kehittäminen. Automaatiotestaukseen on kuitenkin valmiina useita automaatiokehyksiä, joita voidaan hyödyntää testauksessa.

Monesti testiautomaation käyttöönoton syynä on kustannussäästöt. Testiautomaation avulla voidaan vähentää huomattavasti manuaalisen työn määrää. Näin henkilöresursseja voidaan kohdistaa tarkemmin tiettyjen osa-alueiden testaamiseen. Myös regressiotestauksessa hyödynnetään yleisesti testausautomaatiota. Regressiotestauksen tarkoituksena on tarkistaa, että sovellukseen tehdyt muutokset eivät ole aiheuttaneet ei-toivottuja ongelmia ja että sovellus on edelleen määritysten mukainen.

Loimaan seudun kennelkerho Ry:n verkkosivuston testauksessa hyödynnettiin vapaaseen lähdekoodiin perustuvaa Robot Framework testiautomaatiokehystä. Päädyin Robot Frameworkin käyttämiseen, koska halusin itse oppia lisää sen hyödyntämistä verkkosivustojen toteutuksessa.

Testauksen aluksi määrittelin yleisimmät testitapaukset, joiden avulla pystyttäisiin varmistamaan, että sivusto toimii edelleen määritysten mukaisesti. Eli mikään uusi toiminto tai muutos ei ole aiheuttanut ongelmia sivuston muilla osa-alueilla.

```

*** Settings ***
Suite Setup      Open browser to login page
Test Template    Login with invalid credentials should fail
Suite Teardown   Close browser
Resource         resource.txt

*** Test Cases ***
Invalid user name      USER NAME    PASSWORD
                       invalid       mode
Invalid password      demo         invalid
Both invalid          invalid      whatever
Empty user name       ${EMPTY}    mode
Empty password        demo        ${EMPTY}
Both empty            ${EMPTY}    ${EMPTY}

Invalid login
                       invalid       mode
                       demo         invalid
                       invalid      whatever
                       ${EMPTY}    mode
                       demo        ${EMPTY}
                       ${EMPTY}    ${EMPTY}

*** Keywords ***
Login with invalid credentials should fail
[Arguments]    ${username}    ${password}
Enter user name    ${username}
Enter password    ${password}
Press login button
Error page should be open
[Teardown]    Go to login page

Error page should be open
Title should be    Error Page

```

Kuva 9: Esimerkki sivustolle luodusta Robot Framework testitapauksista

Kun olin määritellyt yleisimmät testitapaukset aloitin itse Robot Frameworkin asentamisen. Robot Framework vaatii taustalle Pythonin, jotta pystyin asentamaan itse Robot Frameworkin. Robot Frameworkin asentaminen ja käyttöönotto on melko suoraviivaista ja helppoa. Testikirjastona testauksessa käytin valmiiksi luotua Selenium-kirjastoa. Selenium-kirjaston on tarkoitettu selaimen ohjaukseen. Robot Frameworkin ydin ei itsessään tarvitse mitään tietoa testattavasta sovelluksesta, vaan kaikki testit ajetaan testikirjastojen kautta.

Testien ajaminen tapahtuu joko Robot Frameworkin oman integroidun ympäristön Riden tai komentokehoteen kautta. Tämän opinnäytetyön produktin toteutuksessa testit ajettiin komentokehoteen kautta. Testien käynnistäminen komentokehoteelta tapahtuu antamalla käsky ”pybot” sekä testitiedoston nimen tai hakemiston, jossa testitiedostot ovat. Alla olevasta kuvasta löytyy esimerkki käskyistä.

```
$ pybot testi.txt
$ pybot hakemisto/
```

Kuva 10: Komennot Robot Framework testien käynnistämiseen komentokehoteella.

Kun kaikki testit ovat suoritettu, muodostaa Robot Framework testien tuloksista raportin. Raportti muodostetaan xml-muotoisen raportointityökalun toimesta. Muodostetun raportin voi avata suoraan selaimen, josta näkee testien tulokset sekä suoritusajat. Robot Frameworkin raportointi perustuu loki-, raportti- ja yhteenvetotiedostoihin joista raportti koostetaan.

6 Yhteenveto

Toiminnallisen opinnäytetyön produktin toteutus kesti melko pitkään ja siihen meni paljon työtunteja. Sivustosta tuli kokonaisuutena erittäin hyvin toimeksiantajan vaatimuksia vastaava. Sivuston ulkoasu ei loppujen lopuksi muuttunut kovin paljoa, koska värimaailma pysyi entisen kaltaisena toimeksiantajan toiveesta. Mielestäni sivuston ulkoasu vastaa kuitenkin nyt paremmin tämän hetkisiä trendejä verkkosivustojen suunnittelussa. Sivusto on selkeä ja sieltä löytyy vaaditut tiedot helposti. Responsiivisen sivuston toteutuksesta minulla ei ollut aiempaa kokemusta, mutta olen silti tyytyväinen saatuihin lopputuloksiin tämän suhteen.

Drupal julkaisujärjestelmän vuoksi sivusto on nyt paljon helpompi ylläpitää, koska ei vaadi merkintäkielten osaamista. Opinnäytetyön aikana huomasin, että Drupal on erittäin monipuolinen ja helppokäyttöinen julkaisualusta. Juuri helppokäyttöisyyden vuoksi uskonkin, että Drupalin suosio yhtenä suosituimmista WWW-julkaisujärjestelmistä säilyy jatkossakin. Minulla oli jonkin verran kokemusta jo aiemmin Drupalista, mutta työn aikana opin paljon uusia asioita.

Esimerkiksi merkintäkielien osalta osaamiseni kasvoi loppujen lopuksi melko paljon. Tämä tulee varmasti hyödyttämään tulevaisuudessa työelämässä. Suurempia haasteita ei loppujen lopuksi ilmaantunut työn tekemisen aikana.

Koska yhtenä osa-alueena opinnäytetyössä oli Robot Framework -testiautomaatiokehikseen tutustuminen, osaamiseni kasvoi myös sen osalta todella paljon. Kuten Drupalkin, Robot Framework oli ennestään jo jonkin verran tuttu. Tämän työn aikana osaamiseni kuitenkin syvenyi todella paljon ja uskoisinkin, että pääsen hyödyntämään uutta osaamistani tulevissa työtehtävissä testauksen parissa. Esimerkiksi uudelleenkäytettävien testien rakentaminen kehittyi huomattavasti.

Robot Framework saattaa kuitenkin olla melko haasteellinen ottaa käyttöön verkkosivuston testaamisessa, jos ei omaa jo ennestään jonkin verran ohjelmointitaitoja. Edes auttava ohjelmoinnin osaaminen helpottaa esimerkiksi testikirjastojen kirjoittamisessa. Yhtenä haasteena huomasin myös automaatiotestien ylläpidettävyyden. Siksi onkin tärkeää, että testit ovat helppolukuisia ja itsensä selittäviä. Tällöin testien ylläpito onnistuu helpommin myös muilta kuin sivuston ylläpitäjältä. Testit tulisikin kirjoittaa siten, että käytännössä kuka tahansa pystyy selvittämään niiden tarkoituksen ja toiminnan ilman, että osaa tehdä automaatiotestejä.

Robot Framework soveltuu kuitenkin todella hyvin verkkosivuston automaatiotestaukseen. Sivuston kehityksen tukena automaatiotestauksen käyttö on todella suositeltavaa. Automaatiotestauksen avulla sain pidettyä sivuston toteutuksen laadun koko ajan vaadittavalla tasolla, eikä suuria yllätyksiä sivuston toiminnallisuuden rikkoontumisen kanssa päässyt tapahtumaan. Myös testauksen työmäärän kannalta testiautomaation käyttö oli todella hyvä ratkaisu. Testiautomaation avulla pystyin suorittamaan testit huomattavasti nopeammin, kuin jos olisin tehnyt sen manuaalisen testauksen menetelmin.

Toimeksiantaja on ollut tyytyväinen uuteen sivustoon ja sivuston luominen oli todella opettavainen kokemus myös minulle. Sivuston ylläpitäminen on nyt toimeksiantajalle paljon helpompaa ja he pystyvät ylläpitämään sivustoa itsenäisesti ilman ulkopuolista osaamista.

Lähteet

- Burge, S. & McCourt, C. 2013. Drupal 7 explained : your step-by-step guide.
- Jehl, S. 2014. Responsive Responsible Design. New York: A Book Apart.
- Juslén, J. 2011. Nettimarkkinoinnin karttakirja. Helsinki: Tietosykli.
- Kasurinen, J. 2013. Ohjelmistotestauksen käsikirja. Jyväskylä: Docendo.
- Korpela, J. & Linjama, T. 2005. Web -suunnittelu. Jyväskylä: Docendo.
- Kraynak, J. & Bluttman, K. 2011. Brilliant Web Design. Harlow: Pearson Educations.
- Shreves, R. & Dunwoodie, B. 2011. Drupal 7 Bible. Indianapolis: Wiley.
- Vilkkä, H. & Airaksinen, T. 2003. Toiminnallinen opinnäytetyö. Helsinki: Tammi.
- Drupal.org. 2015. Views. Viitattu 1.12.2015.
<https://www.drupal.org/project/views>
- Ekonoja, A., Lahtonen, T. & Mäntylä, J. 2004. Relaatiotietokannat. Viitattu 27.11.2015.
<http://appro.mit.jyu.fi/doc/tiedonhallinta/tietokannat/index1.html>
- Larman, C & Vodde, B. 2010. Acceptance Test-Driven Development With Robot Framework. Viitattu 23.11.2015
http://wiki.robotframework.googlecode.com/hg/publications/ATDD_with_RobotFramework.pdf
- MySQL AB:n WWW-sivu. Viitattu 15.11.2015.
<http://www.mysql.com/>
- The PHP Group. 2015. Usage stats for January 2013. Viitattu 30.11.2015.
<http://www.php.net/usage.php>
- Robotframework.org. 2015. Introduction. Viitattu 15.11.2015.
<http://robotframework.org/>

Kuvat

Kuva 1: Drupal asennusohjelman aloitusruutu.....	10
Kuva 2: Drupal-sivuston teeman käyttämä Style.css -tyylitiedosto	11
Kuva 3: HTML -sivun perusrakenne	12
Kuva 4: Esimerkki Drupalin käyttämästä, teeman ulkoasuun liittyvästä, PHP -tiedostosta	13
Kuva 5: XAMPP -ohjelmiston phpMyAdmin MySQL -tietokannan hallintatyökalu.....	14
Kuva 6: XAMPP -komponenttien tilat selainkäyttöliittymällä.	15
Kuva 7: Esimerkki tekstitiedostoon tehdystä avainsana ohjatusta testitapauksesta.....	16
Kuva 8: Loimaan seudun kennelkerhon verkkosivuston etusivun rautalankamalli.....	17
Kuva 9: Esimerkki sivustolle luodusta Robot Framework testitapauksista	20
Kuva 10: Komennot Robot Framework testien käynnistämiseen komentokehotteella.	21