



**LAUREA**  
AMMATTIKORKEAKOULU

*Uuden edellä*

# Verkkosivuston tuottaminen Drupalilla Case City Rabbits

---

Mansikka, Satu & Ventä, Sini

2013 Leppävaara

Laurea-ammattikorkeakoulu  
Laurea Leppävaara

Verkkosivuston tuottaminen Drupalilla  
Case City Rabbits

Mansikka, Satu & Ventä, Sini  
Tietojenkäsittelyn koulutusohjelma  
Opinnäytetyö  
Toukokuu, 2013

Mansikka, Satu & Ventä, Sini

### Verkkosivuston toteuttaminen Drupalilla Case City Rabbits

Vuosi	2013	Sivumäärä	40
-------	------	-----------	----

---

Tämän opinnäytetyön tavoitteena oli internetsivuston rakentaminen City Rabbits-verkkopalvelulle. Palvelu on tarkoitettu toimimaan kontaktina matkailijan ja paikallisen tiedon välillä. Palvelussa matkailijan on mahdollista selvittää häntä kiinnostavia kohteita tulevalta matkaltaan.

Sivuston kehittäminen tehtiin konstruktiivisella tutkimusotteella. Kehittämisprosessi vaiheistettiin vesiputousmallin mukaan. Suunnittelussa ja toteutuksessa käytettiin iterointia ja käyttöliittymän tuottamisessa protoilua.

Sivuston vaatimusmäärittely perustui toimeksiantajan liikeideaan ja sen pohjalta luotuun suunnitelmaan sivuston ulkoasusta ja toiminnallisuuksista. Itse sivusto toteutettiin sisällönhallinta- ja julkaisujärjestelmä Drupal 7:lla, joka mahdollistaa sivuston helpon ylläpidon ja palvelun jatkokehittämisen. Prosessin aikana syntyneet jatkokehittämisideat raportoitettiin osana opinnäytetyötä.

Lopputuloksena syntyi City Rabbits- internetsivusto toimeksiantajan käyttöön. Palvelun keskustelu toimii Facebookin kautta ja palvelu sisältää toimintamahdollisuudet muun muassa neuvontaan, bloggaukseen ja yhteydenottoon. Sivusto on sijoitettu toimeksiantajan operaattorin palvelimelle, josta se voidaan ottaa käyttöön tarvittaessa. Sivuston ylläpitoa ja hallinnointia varten toimeksiantajalle tuotettiin englanninkielinen ylläpitodokumentti.

Mansikka, Satu & Ventä, Sini

**Implementation of a website with Drupal: a case study of City Rabbits**

Year	2013	Pages	40
------	------	-------	----

---

The objective of this thesis was the implementation of a website for the web service City Rabbits. This service is meant to be a contact between travelers and the local people. With it, travelers can find information about their personal interests concerning the destination.

The constructive research method was used in the project. The developing process was performed with the waterfall model. Iteration was used then planning and implementing. The user interface was developed from prototypes.

The table of requirements was based on the plan received from the customer. The visuals and functions were derived from this plan. The site itself was developed with Drupal 7, which enables easy maintenance and further developing. Proposals for further developing are gathered as a part of the thesis.

As an outcome, a webpage was created for the use of the customer. The chat functions through Facebook and the service include the possibility of advising, blogging and contacting the site administrators. The webpage is located at a server under the customer's management. Also, it can be taken in use when required. For maintenance and content update there was a separate guide produced in English for the customer.

Drupal, website, content management system, web service, City Rabbits

## Sisällys

1	Johdanto.....	6
1.1	City Rabbits .....	6
1.2	Palvelun ja sivuston kehittyminen .....	7
2	Tavoitteet ja menetelmät.....	8
2.1	Opinnäytetyön tavoitteet.....	8
2.2	Menetelmät .....	9
3	Sivuston kehittäminen.....	10
3.1	Vaatimusmäärittely .....	10
3.2	Käyttöliittymän suunnittelu .....	11
3.3	Sivuston tietohierarkia .....	14
4	Sivuston toteuttaminen .....	15
4.1	Toteutuksen työvälineet .....	16
4.2	Drupal 7 .....	17
4.3	Käytetyt Drupal -moduulit.....	19
4.4	XAMPP .....	20
4.5	cPanel .....	21
4.6	Sivujen testaus .....	21
4.7	Ylläpitodokumentin suunnittelu ja toteutus .....	21
5	Opinnäytetyön haasteet .....	23
5.1	Työskentely .....	23
5.2	Riskit ja niiden toteutuminen.....	24
6	Johtopäätökset ja jatkokehitys.....	25
6.1	City Rabbits -sivusto .....	26
6.2	Kehitysehdotukset .....	29
6.3	Yhteenveto .....	30
	Lähteet .....	32
	Sanasto.....	34
	Kuvat .....	37
	Kuviot .....	38
	Taulukot .....	39
	Liitteet.....	40

## 1 Johdanto

Tämä opinnäytetyö käsittelee City Rabbits -verkkosivuston tuottamista. City Rabbits on omatoimimatkailijan sivusto, jonka on tarkoitus mahdollistaa kontakti matkailijan ja matkakohteen välillä. Palvelun kautta matkailija voi kysyä matkakohteestaan kysymyksiä, jotka auttavat matkailijaa yksilöllisen matkailukokemuksen luomisessa. Sivustolta saa siis informaatiota, jotka jäävät usein matkailutoimistojen pakettimatkakokonaisuuksien ulkopuolelle. Kysymyksiin vastaavat City Rabbits -palvelun muut käyttäjät, paikalliset ihmiset tai kohteessa vierailleet matkailijat.

Verkkopalvelusta haluttiin sivusto, joka on helppokäyttöinen ja interaktiivinen ja joka tarjoaa omatoimimatkailijalle oleelliset toiminnot yhteydenpitoon. Asiakkaan kanssa sovittiin myös sivuston ylläpitämiseen liittyvän oppaan laatimisesta. Lopullisina tuotteina toimeksiantaja saa palvelimellaan toimivan verkkopalvelusivuston ja sen ylläpitodokumentin.

Opinnäytetyössä on raportoitu sivuston rakentamisen vaiheet ja tekniset ratkaisut, sekä projektin aikana esille tulleet jatkokehitysehdotukset. Opinnäytetyön liitteistä löytyy sivuston ylläpitodokumentti, josta on poistettu muun muassa käyttäjätunnukset.

### 1.1 City Rabbits

City Rabbits on palvelu, joka toimii verkossa. Se on omatoimimatkailijan sivusto, josta matkailijat voivat hakea tietoa muilta matkailijoilta ja määränpään paikallisilta asukkailta. Liike-idea sai alkunsa Laurean opiskelijalta, Elisa Jouttunpäältä, joka kehitti liike-ideaansa opinnäytetyössään.

”City Rabbits on niille matkustelijoille, jotka löytävät omat tiensä uniikkeihin matkakokemuksiin. Tämä interaktiivinen palvelu mahdollistaa oikean paikallisen vastaamaan yksilöllisiin kysymyksiin koskien omaa matkakohdettaan.” (Jouttunpää 2012a, 1.)

Elisa Jouttunpään mukaan City Rabbitsin visio on: ”CityRabbits.com tulee olemaan ensimmäinen osoite, kun matkaa suunnitteleva lähtee hakemaan uniikkeja nähtävyyksiä ja mielenkiintoista tietoa uudesta matkakohteesta. Meiltä haetaan parhaimmat vaihtoehtoiset ja paikallisten suosimat vinkit, jota voi lähteä täydentämään paikallisten avulla kohdistuen juuri omiin kiinnostuksen kohteisiin.” (2012b, 1)

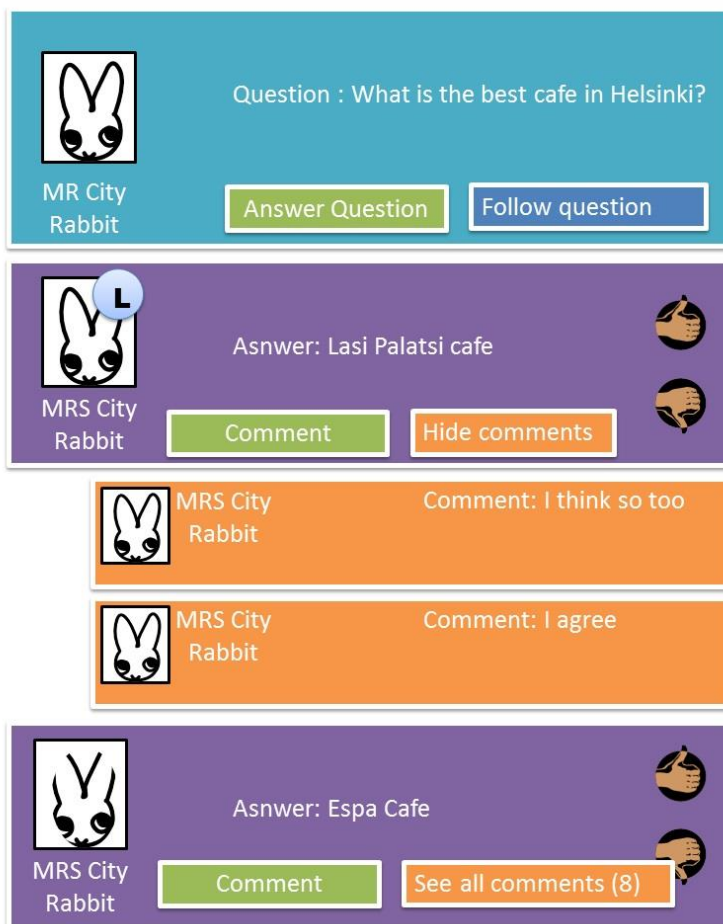
## 1.2 Palvelun ja sivuston kehittyminen

Liikeidea ja ajatus yrityksen perustamisesta syntyivät syksyllä 2010. Ensimmäisenä toimeksiantaja selvitti, onko ideasta kannattavaksi liiketoiminnaksi. Liikeidea testattiin kouluympäristössä. Liikeidean kehitystiimi osallistui erilaisiin tapahtumiin, jossa nostettiin esiin keskustelua ja kyseltiin kiinnostusta palveluun.

Toimeksiantajan tiimi rakensi testisivut, joiden tekoon tiimin ICT-osaaminen riitti. Sivustolla ei ollut graafista ilmettä eikä juurikaan sisältöä, ja tärkein osa sivustoa oli kysymyslomake. Jatkokehityksen kannalta oli tärkeä saada palautetta palvelusta ja selvittää, kuinka monesta matkakohteesta nämä koekäyttäjät kykenisivät esittämään omia kokemuksia sivuston oppaiden tapaan.

Testisivuille saatiin noin sata koekäyttäjää, joiden ilmoittamien matkakohteiden määräksi saatiin 33 eri maata ja näistä vielä useampia eri kaupunkeja. Valtaosa kaupungeista oli luonnollisesti suomalaisia johtuen testikäyttäjien taustasta. Palvelun kehittämisen kannalta tärkein kysymys saatiin kuitenkin ratkaistua: olisiko tälle liikeidealle kannattavaa rakentaa palveluliiketoimintaa. Liikeidean testauksen jälkeen tarvittiin ICT-osaajia sivuston rakentamiseen. Varsinainen selvitystyö palvelua varten tehtiin keväällä 2011, sen tuotoksena oli kuvion 1 mukainen sivuproto.

# Helsinki



Kuvio 1: Ensimmäisiä luonnostelmia sivustosta

## 2 Tavoitteet ja menetelmät

Alkuperäinen kevyt suunnitelma sivuston ulkoasusta ja toiminnoista oli olemassa, tämän opinnäytetyön toteutussuunnitelma laadittiin sen pohjalta. Opinnäytetyössä käytettiin käytännönläheisiä menetelmiä.

### 2.1 Opinnäytetyön tavoitteet

Opinnäytetyön tavoitteena on rakentaa toimiva verkkopalvelusivusto, joka on valmis asiakkaan käyttöön. Sivuston on tarkoitus olla valmiina asiakkaan rahoittamalla palvelimella, jotta hän pystyy helposti käynnistämään yrityksen toiminnan.



Verkkopalveluun on tarkoitus integroida omatoimimatkailijaa palvelevia perustoiminnallisuuksia kuten "usein kysytyt kysymykset" (FAQ), blogi, opas -profiilit, käyttöehdot ja yhteystiedot. Sivustolla on myös varattu tilaa mainostajille. Tavoitteena on myös tuottaa sivustolle erillinen englanninkielinen käyttöohje, jonka avulla City Rabbits - sivuston päivittäminen ja sisällön lisääminen onnistuu.

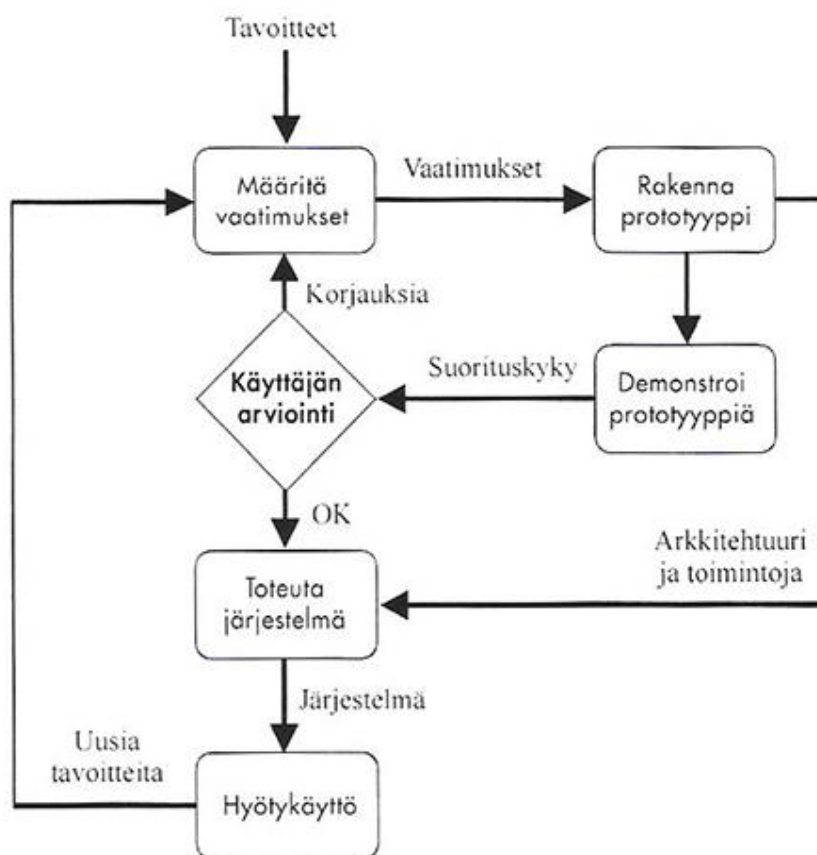
## 2.2 Menetelmät

Opinnäytetyössä käytettiin konstruktivistista tutkimusotetta, joka ilmenee muun muassa suunnittelussa ja toteutuksessa. Toimeksiantaja osallistui alkuvaiheessa sivuston kehittämiseen henkilökohtaisesti. Myöhemmin tapaamiset ja kommunikointi tapahtui Facebookin kautta.

Prosessissa oli havaittavissa selkeää iterointia, jossa edellisen osan toimivuudesta täytyi varmistua ennen seuraavaan osaan siirtymistä. Kehittämisprosessi vaiheistettiin vesiputousmallin mukaan (kuvio 2). Suunnittelussa ja toteutuksessa tarvittavaa tietoutta kerättiin yhdessä toimeksiantajan kanssa. Yhteisissä palavereissa myös hyväksyttiin toimeksiantajalla sivuston viimeisimmät ratkaisut.

Konstrukttiivinen tutkimusote soveltuu case -tutkimukselle. Tämän tutkimusotteen avulla pyritään ratkaisemaan aitoja reaali maailman ongelmia ja tätä kautta tuottamaan kontribuutioita eli tuotoksia. Tässä opinnäytetyössä tuotoksena syntyi verkkosivusto, ylläpito-dokumentti ja opinnäytetyön kirjallinen raportti. (Lukka.)

Prototyypilähestymistavassa tilaajan on aluksi vaikeaa tarkkaan määritellä haluamaansa järjestelmää. Tässä lähestymistavassa asiakkaalle tuotetaan prototyyppejä, joista selviää kevyesti luonnostellut perustoiminnot. Kuviosta 2 näkee prototyypilähestymistavan periaatteen. Prosessi alkaa vaatimusten ja tavoitteiden analysoinnilla, joiden perusteella rakennetaan prototyyppi. Asiakkaan palautteen mukaan prototyyppiä parannellaan, kunnes asiakas on tyytyväinen. (Pohjonen, 41.)



Kuvio 2: Prototyypilähestymistapa (Pohjonen 2002, 41)

Sivuston suunnittelu ja toteutus tapahtui iteratiivisesti. Iteratiivisessa toteutustavassa suunnittelua ja toteutusta tehdään pienissä osissa ja tätä prosessia toistetaan. Tällä tavoin sivusto kehittyi inkrementaalisesti, eli koko ajan kasvaen, loppujen lopuksi päätyen lopputulokseen. Toteutusvaiheessa sisällönhallintajärjestelmän ratkaisuja benchmarkattiin olemassa olevilta sivustoilta. Benchmark -tutkimuksessa valmiiden tuotteiden ominaisuuksia verrataan toisiinsa ja tätä kautta hyviä ratkaisuja voidaan hyödyntää uudestaan (Sinkkonen, Nuutila & Törmä 2009, 257).

### 3 Sivuston kehittäminen

Ajatus City Rabbits -palvelusta on alun perin Elisa Joutunpään luoma, ja hänen opinnäytetyönsä ja alustavan ajatustyönsä pohjalta tuotettiin tässä opinnäytetyössä konkreettiset verkkosivut palvelulle City Rabbits.

#### 3.1 Vaatimusmäärittely

Hyvän käyttöliittymän suunnittelun ja toteutuksen pohjalla on hyvä käyttökokemus ja käytettävyys. Käyttöliittymän suunnittelu on tärkeä osa sivuston toteuttamista. Sivuston

tuottamisessa ensisijaisena lähtökohtana voidaan pitää alkuperäisen liikeidean esitystä siitä, mitä palvelulla halutaan saavuttaa. Joutunpään (2012a) mukaan City Rabbits on suunnattu itsenäisille matkailijoille, jotka haluavat löytää uusia kokemuksia matkoiltaan. Palvelun tarkoitus on vastata tämän päivän matkailijoiden yksilöllisiin tarpeisiin. Interaktiivisesta yhteisöstä löytyy juuri oikean henkilö vastaamaan uniikkeihin kysymyksiin matkaa koskien. Cityrabbits.com tuo paikallisten tiedon matkustajien saataville helposti ja nopeasti.

Keskeisenä ajatuksena on kokemusten jakaminen matkakohteista lomamatkaa tai muuttoa varten. Joutunpään mukaan kuitenkin viimeisimpiin yksityiskohtiin vievää keskustelua ei ole tarpeellistakaan käydä sivuston kautta, vaan osapuolet voivat palvelun avulla löytää toisensa ja jatkaa keskustelua valitsemallaan sosiaalisen median kanavalla.

Palavereissa keskusteltiin pitkään siitä, miten kysymysten esittäminen ja keskustelu palvelussa kyettäisiin järjestämään. Toistaiseksi päädyttiin siihen, että palvelussa on Facebook -seinä, jolle Facebookiin rekisteröityneet käyttäjät voivat julkaista viestejä. Facebook -liitännäisten tuominen palveluun mahdollistaa sen, ettei tavallisten käyttäjien tarvitse erikseen rekisteröityä palveluun hyötyäkseen siitä. Usein kirjautumista vaativat palvelut nostavat kynnyksen käyttäjillä palvelun käyttämisessä. Facebook -seinän lisäksi palveluun tehtäisiin lyhyet esittelyt henkilöistä, jotka voivat jakaa kokemuksiaan omista matkakohteistaan. Esittelyt sisältäisivät yhteenottomahdollisuuden näihin henkilöihin.

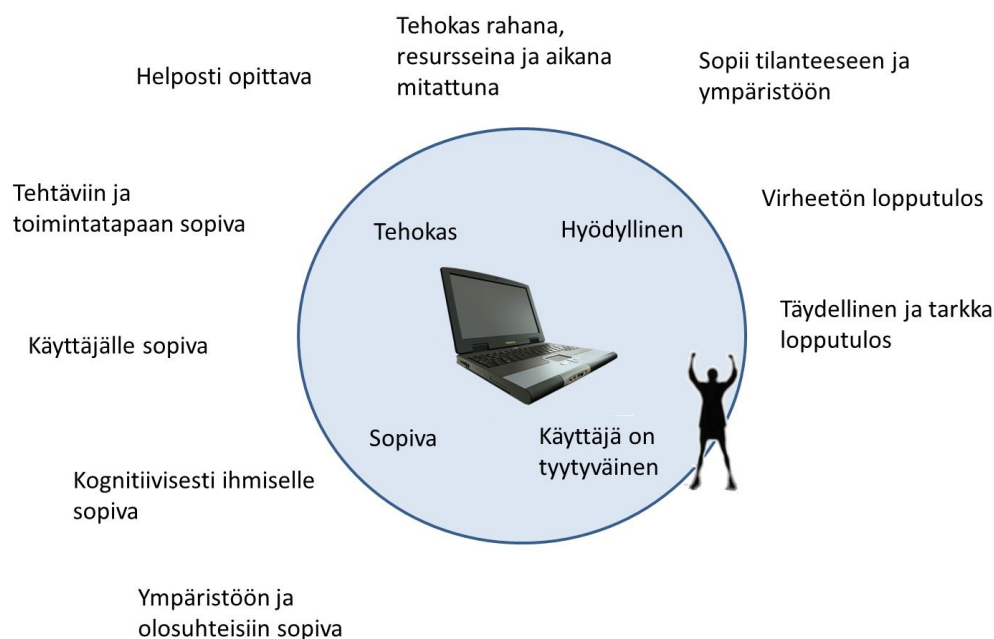
Asiakkaan toivomuksesta sivustolle luodaan myös blogi -osio, jonka tarkoitus olisi osoittaa se, että palveluun tulee uutta sisältöä. Tämä tasapainottaisi erityisesti palvelua alkuvaiheessa, mikäli keskustelualueen toiminta jää vähäisemmäksi. Blogi on myös sivuston ylläpitäjän keino profiloitua itsestään tai luoda kontaktia palvelussa vieraileviin käyttäjiin.

### 3.2 Käyttöliittymän suunnittelu

Sinkkosen ym. (2009, 23) mukaan käyttökokemuksella tarkoitetaan käyttäjän tuntemuksia hänen käyttäessään palvelua. Käyttökokemuksen tulee olla sellainen, että käyttäjä jää sivustolle ja käyttää sitä sekä palaa käyttämään palvelua. Tuntemukseen vaikuttavat itse palvelu ja käyttötilanne, sekä käyttäjän vanhat kokemukset ja mielipiteet palvelun ominaisuuksista, hyödyllisyydestä, sisällöstä ja tuotteen esillepanijasta.

Verkkosivuston käyttökokemus koostuu sisällöstä, omaksuttavuudesta, merkittävydestä, asioiden löydettävyydestä, terminologiasta, visuaalisesta ilmeestä ja monesta muusta asiasta. Käyttäjäkokemukseen taas vaikuttaa enemmän se, kuinka hyvin palvelun logiikka istuu käyttäjien tapaan tehdä tehtäviään. Kun sovellusta ei huomaa lainkaan, verkkopalvelun käyttökokemus on silloin parhaimmillaan. (Sinkkonen ym. 2009, 23.)

Sinkkosen ym. (2009, 20, 27) mukaan ISO 9241-11 -standardi määrittelee käytettävyyden mittariksi, jolla mitataan, kuinka käyttökelpoinen, tehokas ja miellyttävä tuote on käyttäjä oikeassa käyttöympäristössään, kun käyttäjinä ovat sen omat käyttäjät. Käytettävyys palveluun ja hyvä käyttäjäkokemus rakennetaan käyttäjäkeskeisillä suunnittelumenetelmillä.



Kuvio 3: Käytettävyys (Sinkkonen 2009, 21)

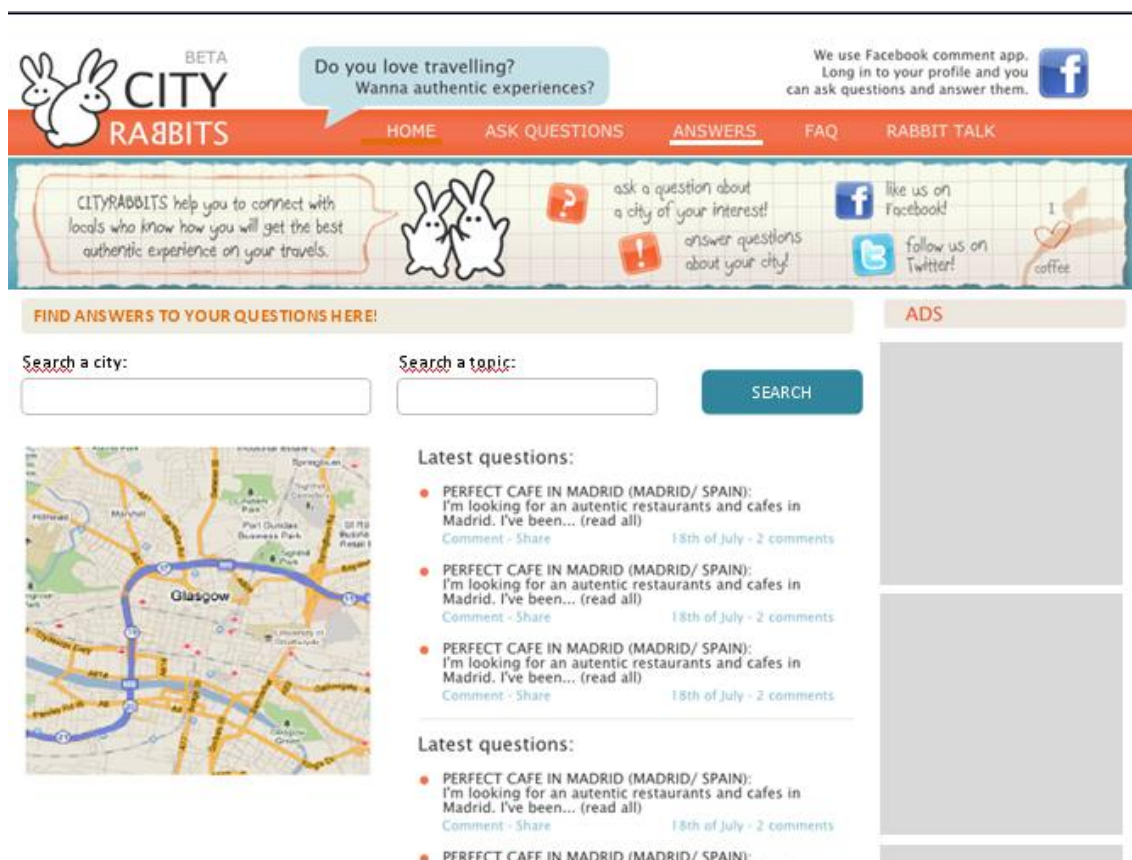
Ennen varsinaisen työn aloittamista asiakkaalta saatiin alustavat suunnitelmat siitä, miltä sivuston ilme tulisi näyttää. Palavereissa kävimme läpi tärkeimmät elementit ja sen miltä näiden tulisi näyttää (muun muassa logot ja taustavärit). Todellinen sivun ulkoasu muokkautui iteratiivisesti tekniikan sanelemissa rajoissa. Muutoksista keskusteltiin asiakkaan kanssa palavereissa ennen kuin ne jäivät osaksi lopullista työtä.

Asiakkaan kanssa keskusteltiin palavereissa myös sivuston pitämisestä yksinkertaisena. Monet yksinkertaisuuden tähtäävistä ulkoasu ratkaisuista on tehty iteratiivisesti. Esimerkkinä kuitenkin mainittakoon Jukka Korpelan ja Tero Linjaman (2005, 256) maininnan kirjassaan verkkosivujen ulkoasusta: ilmeen tulisi olla yhtenäinen läpi sivuston, mutta siitä huolimatta helposti luettavissa. Tekstiä sisältävissä palstoissa taustan tulisi olla tasainen, mielellään koostua vain yhdestä väristä ja teksti tästä taustasta selkeästi erottuvaa. Samoin myös tekstipalstoja ei kannata olla yhtä peruspalstaa enempää.

Korpelan ja Linjaman (2005) periaatteet toimivat myös yleisinä periaatteina ulkoasun tuottamisessa ja City Rabbits -sivustolle päädyttiin valkoiseen taustaan tummalla tekstillä. Tekstin perusväri ei ole täysin musta, mikä esimerkiksi Juseliuksen (2004) mukaan helpottaa hivenen luettavuutta. Linkit on luonnollisesti korostettu sivustoon sopivalla värillä. Palstoittamisesta yleisilmeessä päädyttiin kahden palstan tekniikkaan, josta vain toiselle sijoitetaan teksti. Toinen palsta on tarkoitettu mahdollisia kuvia varten.

Ylä- ja alatunniste -osiot osiot on toteutettu sisältämään navigoinnin kannalta oleelliset asiat. Ylätunniste eli ylin navigointipalkki sisältää sivuston toimintojen kannalta keskeiset elementit. Alatunniste eli sivuston alempi navigointipalkki taas sisältää sivuston ylläpitäjään ja yhteystietoihin liittyvät linkit. Tämä lajittelu on ensisijaisesti asiakkaalta lähtöisin oleva ehdotus, mutta noudattaa myös Korpelan ja Linjaman kirjan (2005) mukaista käsitystä helposta navigoinnista.

Alkuperäisen asiakkaalta tulleen sivuston suunnitelman graafisesta ilmeestä vastasi Riina Ruppenen, toimeksiantajan tuttava, joka on opiskellut grafiikkaa. Hän oli myös mukana kokouksissa, joissa sivuston ulkoasua kehitettiin eteenpäin opiskelijoiden toimesta. Kuviosta 4 näkyy Ruppenen luonnostelmaa nettisivujen ulkoasusta, joka toteutettiin ennen opinnäytetyön tekijöiden tuloa projektiin. Opinnäytetyön aikana tälle suunnitelmalle pyrittiin olemaan mahdollisimman uskollisia. Tästä poikettiin, kun suunnitelmassa esitettyä toimintoa ei ollut mahdollista rakentaa teknisistä syistä täysin samanlaiseksi.



Kuvio 4: Riina Ruppen luonnostelma sivuston graafisesta ilmeestä

Sivustolla ei ole käyttäjälle rekisteröinti-mahdollisuutta. Ylläpito-tason käyttäjät pystyvät kuitenkin luomaan uusia käyttäjätunnuksia. Tällä hetkellä käyttäjätunnuksille ei käyttäjien puolesta ole tarvetta, sillä kysymyspalsta toimii Facebook kirjautumisen kautta. Oppaiden profiilien lisäämisestä sivustolle vastaa ylläpitäjä.

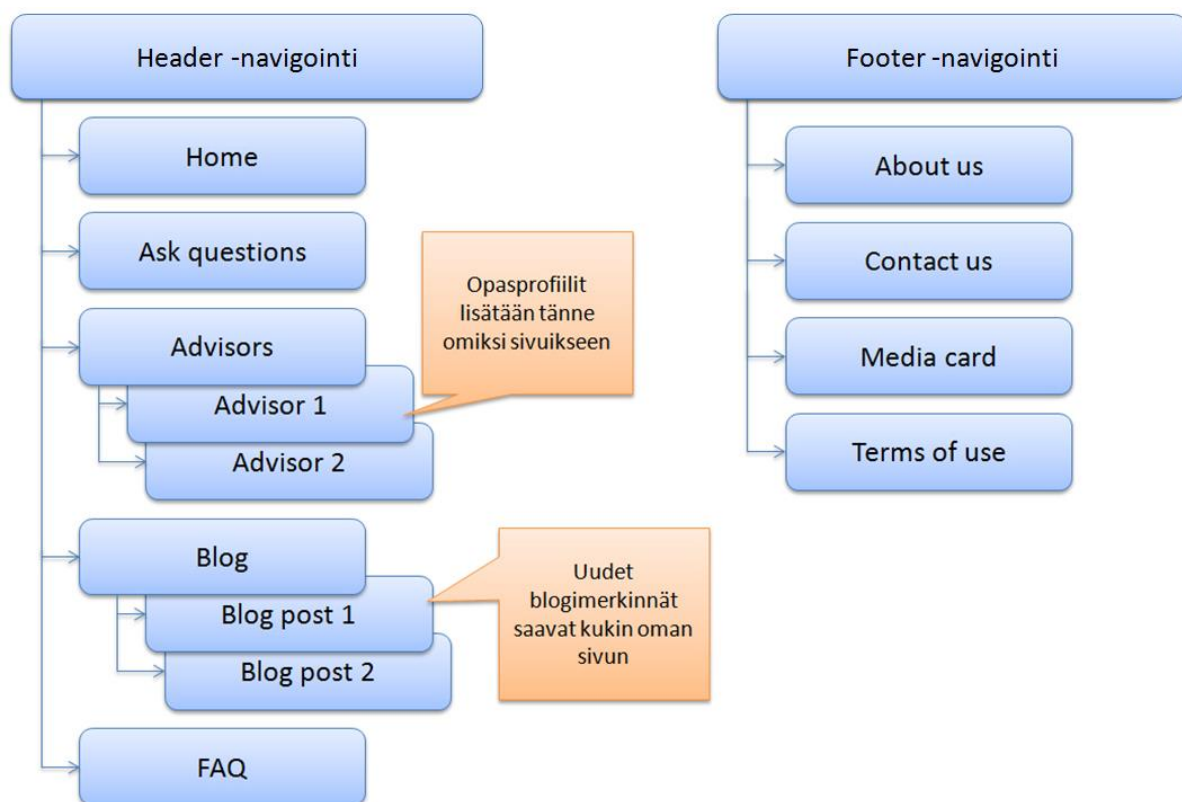
### 3.3 Sivuston tietohierarkia

Sivustolla navigointi oli alunperin tarkoitus pitää selkeänä ja yksinkertaisena. Perinteisestä keskusnavigointipalkista löytyy sivuston keskeiset linkit (muun muassa kysymyspalsta, oppaiden esittely ja blogi) ja sivuston kannalta tekniset tiedot, kuten yhteystiedot löytyvät alatunnisteesta. Nämä molemmat navigointipalkit ovat näkyvissä kaikilla sivuilla, mikä helpottaa oikean tiedon löytymistä.

Ylätunnisteen navigointi edustaa keskusnavigointipalkkia ja on sivuston palvelun kannalta tärkeämpi. Alatunnisteen navigointi sijaitsee sivun alalaidassa ja sisältää linkit, joita ei voida jättää pois sivustolta, mutta sisällöltään asetettiin alatunniste -palkkiin. Muun muassa tämä

jaottelu oli peräisin asiakkaan alkuperäisestä suunnitelmasta. Hierarkian ollessa kevyt, sivustolla ei eksy.

Kun vertaamme kuvion 5 sivukarttaa edellisen alaluvun kuvioon 4 , voidaan tietohierarkiassa havaita, että Answers -osio on jäänyt kokonaan pois. Kuvio 4 esittää alkuperäisen suunnitelman luonnosta sivustosta ja sen linkeistä. Facebookin käyttöehdoista johtuen, vastausten "tietovarasto" -ideasta jouduttiin luopumaan. Keskusteluiden informaation tallennusta on mahdollista opinnäytetyön valmistumisen jälkeen vielä jatkokehittää. Jatkokehitysehdotukset löytyvät kokonaisuudessaan luvusta 6.2 Kehitysehdotukset.



Kuvio 5: Sivuston tietohierarkia

#### 4 Sivuston toteuttaminen

Toteutuksen vaiheet keskittyivät pitkälti teknisten ratkaisujen tuottamiseen.

Toteutussuunnitelman mukaisesti sivustolle oli saatava aikaiseksi määritellyt toiminnot ja säilyttää alkuperäinen ilme.

Sivuston näkyvyyttä ja hallinnointia verkossa hallinnoidaan cPanelin kautta, joka on tämän palvelun käyttämä palvelin. Itse osoite Cityrabbits.com oli varattu valmiiksi sivustoa varten

ennen projektin alkua. Tämä osoite pysyi huoltotilassa niin kauan kun sivustoa rakennettiin, jolloin osoitteesta avautui ainoastaan ilmoitussivu, jossa kerrottiin ettei sivusto ole vielä käytössä. Sivusto avattiin käyttöön cPanelin ja sivuston ylläpito -tilin kautta, kun lopulliset tiedostot oli ladattu cPanelin palvelimelle.

Sivusto itsessään on rakennettu Drupalilla ja työskentelyyn on käytetty Laurea - ammattikorkeakoulun tietokoneita. Erityisesti projektin loppuvaiheilla työskentelyä on jatkettu myös kotona. Sivuston tuottamisen aikana väliaikaiset tiedostot ja varmuuskopiot ovat olleet tallennettuna muun muassa liikuteltaville kovalevyille ja Dropbox -palvelimelle. Dropboxin kautta saatiin tiedostot avattua myös kotoa käsin.

Palvelun kehittämisessä ja rakentamisessa on käytetty mukana asiantuntijoiden mielipiteitä ja kokemuksia tukemassa työtä. Tätä kautta saatiin hyvät käytänteet osaksi lopullista sivustoa. Palaverit asiantuntijoiden kanssa keskittyivät tekniseen toteutukseen ja työkalujen valintaan. Drupal -asiantuntijoina opiskelijat tapasivat Niko Suomalaisen ja Janne Pesun. Niko Suomalainen työskenteli Laureassa. Janne Pesu, web-suunnittelija, on Muuvgo Ltd:n perustajajäsen. Riina Rupponen, joka auttoi yleisesti teknisissä asioissa ja grafiikoiden kanssa, on myös Muuvgo Ltd:n perustajajäsen.

#### 4.1 Toteutuksen työvälineet

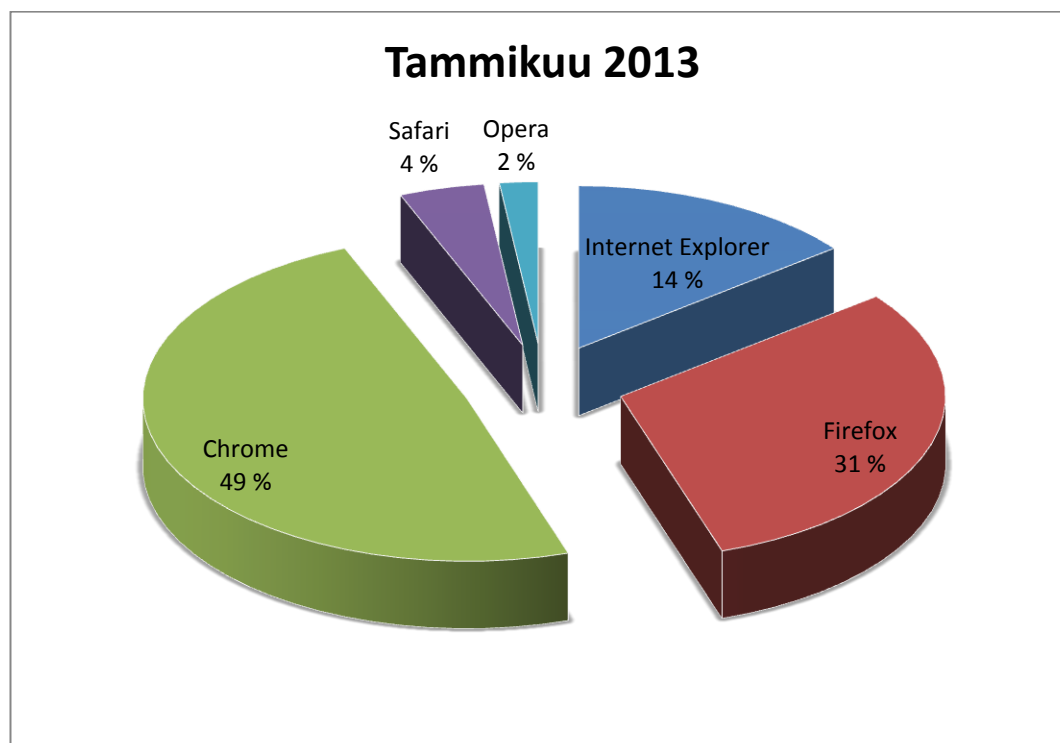
Verkkosivu City Rabbits on rakennettu kokonaan Windows -ympäristössä. Olosuhteet vaikuttivat tähän valintaan, sillä edelleen valtaosa tietokonekäyttäjistä on Windowsin piirissä, sekä työskentely koulun laitteilta onnistui ensisijaisesti kyseiseltä käyttöjärjestelmältä. Opiskelijoiden omilla tietokoneilla on myös Windows -käyttöjärjestelmät. Opiskelijoilla ei ole juurikaan kokemusta muista käyttöjärjestelmistä.

Drupalin rinnalle tarvittiin joukko ohjelmia, jotka ovat omalla osallaan vaikuttaneet lopputulokseen. Useat sivuston toiminnallisuudet ja visuaaliset ratkaisut on toteutettu siten, että yksinkertaisimman Drupal-teeman lähdekoodiin on tehty muutoksia ja lisäyksiä. Tähän on käytetty internetistä ilmaisena ladattavaa Notepad++ -ohjelmaa, joka on perinteisestä Notepad -ohjelmasta koodaajille kehitetty versio.

Visuaalisesta ilmeestä ovat pitkälti vastuussa myös kuvat, joista monet tulivat asiakkaalta. Kuitenkin jonkin verran kuvia on tuotettu opinnäytetyön aikana aina tarpeen vaatiessa muun muassa Adobe Photoshop- ja Inkscape -ohjelmilla. Alkuperäiset vektorigrafiikka - kuvatiedostot oli tuotettu Inkskapella, jolloin tarvittavat muokkaukset on tehty luonnollisesti kyseisellä ohjelmalla.



Internet -selaimet ovat olleet tärkeässä roolissa sivuston toteuttamisessa, joista Mozilla Firefox valikoitui sivustolle käytössä olevista työkaluista koulun koneilla. Laurean koneilla on valittavissa Internet Explorer (IE) ja Firefox, joista Firefox on kevyempi, helppokäyttöisempi ja toimii luotettavammin. Samoin kuvioon 6 verraten, Firefox on edelleen paljon yleisemmin käytetty selain kuin IE. Tammikuussa 2013 Google Chrome on yleisin käytössä olevista selaimista, mutta valintaan vaikutti aiempi opetus jossa oli suositeltu Firefoxia, jolle lisäosa Firebug on tarkoitettu.



Kuvio 6: Internet selainten käyttö maailmalla tammikuussa 2013 (W3Schools.com)

Firebug-lisäosa Firefoxille on yksinkertainen lisätoiminto Drupal-kehittäjille, jonka avulla on mahdollista jäljittää Drupal -sivulta jokainen objekti ja sen vastaava lähdekoodi sen omista CSS-tiedostoista. Käytännössä lisäosa listasi näytölle mitä CSS-tiedoston osaa muokkaamalla on mahdollista muokata kyseistä objektia. Näitä kohteita ovat esimerkiksi ylätunniste, navigointipalkki ja tekstipalstat. Firebug lite on vastaavan lisäosan versio Google Chrome-selaimelle, joka on alkuperäistä kevyempi ja toiminnoiltaan hivenen suppeampi. Hetken työskentely tässä ympäristössä osoitti kuitenkin Firefoxin ja Firebugin luotettavammaksi työkaluksi.

#### 4.2 Drupal 7

Drupal -sisällönhallintajärjestelmä on yksi käytetyimmistä ohjelmista internet -sivujen ja sivustojen luomiseen. Ohjelma on nimensä mukaisesti sisällönhallintajärjestelmä, eli sillä on

mahdollista luoda ja muokata sisältöä ymmärtämättä varsinaisesta koodaamisesta mitään. Opinnäytetyön työvaiheet osoittivat että koodaustaito on eduksi, erityisesti mainittakoon CSS-koodauskielen hallinta. Drupalissa käytetty koodi on pääsääntöisesti CSS -koodia, jolla yksinkertaisillakin muutoksilla saadaan aikaiseksi visuaalisia muutoksia.

Drupal versioista valittiin versionumero 7 työhön, joka oli opinnäytetyön alkamisen aikoihin usuin täysversio. Vanhempi versio 6 oli vielä yleisessä käytössä, mutta version 7 todettiin olevan monipuolisempi.

Riippuen pitkälti siitä, millä aikavälillä sivustolle on tarkoitus tehdä muutoksia, on hyvä ottaa huomioon sen hetkinen usuin Drupal versio. Erityisesti ohjelmistokehityksen alalla uusia versioita tulee nopeaan tahtiin ja uuden version hyödyt kannattaa punnita suhteessa siihen kuinka helposti vanhan työn tuominen uudelle Drupal alustalle onnistuu. Myös uusien moduulien saatavuus käytössä olevalle Drupal versiolle sanelee pitkälle sen, millä Drupal versiolla työtä on mahdollista jatkaa.

Drupal ydintoiminnallisuuksineen ilman moduuleita on melko rajallinen ja työstetyille sivustoille on tarkoitus ladata lisäominaisuuksia Drupal -moduulien muodossa. Moduulit käytännössä säästävät koodin kirjoittamisessa ja monimutkaisemmat toiminnot on helppo lisätä uudella moduulilla. Moduulien kääntöpuolena on se, että monet ladatut moduulit monimutkaistavat sisällönhallintaa ja samalla rasittavat varsinaista sivustoa. Myös monet moduulit sisältävät paljonkin ominaisuuksia, joita ei usein tarvita sivustolla, mikä edelleen tekee sisällönhallinnasta haasteellisempää. Moduuleita löytyy paljon erilaisia samaan käyttötarkoitukseen. Käyttäjän tulee asentaa ja testata moduuleita jonka jälkeen valita käyttöön parhaiten toimiva ratkaisu. Moduuleita ei tule kuitenkaan tulla ottaa käyttöön suuria määriä, sillä tämä saattaa hidastaa sivuston toimintaa ja ylläpidettävyyttä.

Kuviosta 7 voi nähdä Builtwith.com-palvelun kaavion, jonka mukaan julkaisujärjestelmä WordPress hallitsee tällä hetkellä tuotettujen sivujen järjestelmissä. Drupalin etu kuitenkin WordPressiin verrattuna on sen muokkausmahdollisuudet. WordPress tarjoaa lisäosien mahdollisuuksia, mutta näitä ei ole opinnäytetyön aikana kehitetty riittävästi. (wordpress.org.) WordPressin suosiota todennäköisesti kuitenkin puoltaa sen helppokäyttöisyys, sillä WordPressillä ei ole tarvetta muokata koodia. Lopputulos on yleensä kuitenkin tiettyjen valmiiden mallien mukainen. Sivustot erottaa helposti WordPressillä tehdyksi. WordPress toimii hyvin kun halutaan tehdä helppo sivu tai blogi. Voidaan sanoa että WordPress on lähempänä julkaisujärjestelmää kuin sisällönhallintaa.



Kuvio 7: Sisällönhallintajärjestelmien käyttö 2013 (Builtwith.com)

Julkaisujärjestelmät ovat yksinkertaisia web-editoreita, joissa ei ole erityisiä työkaluja sisällön hallintaan, ja ne keskittyvät siihen, miltä lopputulos näyttää. Mitä enemmän näihin järjestelmiin lisätään ominaisuuksia, joilla voi hallita sisältöä ja toiminnallisuuksia, ollaan lähempänä sisällönhallintajärjestelmää. Julkaisujärjestelmät ovat niin ikään kevytversioita sisällönhallintajärjestelmistä, joissa erilaisten toimintojen ja sisältöjen teko on paljon rajatumpaa. (Tolvanen, 2008.)

#### 4.3 Käytetyt Drupal -moduulit

Ladattaessa Drupaliin sisältyy joitakin toimintoja, jotka ovat Drupalin perusversion mukana tulleita valmiita moduleita. Näiden lisäksi usein drupal.org -sivustolta on ladattavissa lukuisia lisäominaisuuksia. Valtava modulien kirjo mahdollistaa toiminnallisuksiltaan hyvinkin erilaisten sivujen toteuttamisen. City Rabbits -sivustolla käytetyt moduulit ovat esiteltyinä seuraavassa.

**Panels** -moduuli palstoittaa sivustot tehokkaammin kuin Drupalin omat asetukset sallivat. Moduulissa on mahdollista valita palstojen määrän, koon ja sisällön uudelle luodulle sivulle. City Rabbitsin sivut ovat kaksipalstaisia, joiden suhde on 70/30.

**Captcha** -moduuli on pääsääntöisesti toiminnassa yhteydenottosivulla. Captcha -moduuli pyytää viestin lähettäjää syöttämään vääntyneen kirjainmerkkijonon annettuun kenttään. Tämän on tarkoitus rajoittaa spam -viestien määrää.

**Facebook comments** -moduuli on vastuussa sivuston yleisestä keskustelupalstasta. Keskustelusivulla on Facebook -seinän kaltainen tila, johon kaikki Facebook -tilin omistavat käyttäjät voivat kirjautuessaan jättää viestejä, jotka ovat kaikkien nähtävissä.

**Facebook like button** -moduuli on yksinkertainen ja pieni moduuli, joka lisää City Rabbits-sivuston etusivulle Facebookin "like" -painikkeen. Facebookiin kirjautuneet käyttäjät voivat "tykätä" City Rabbitsin omasta Facebook-ryhmästä ja seurata ryhmässä julkaistuja viestejä.

**Chaos tools suite (CTools)** -moduuli on Drupalin kehittäjille suunnattu, joka tekee moduulien hallinnasta yksinkertaisempaa. Tämän moduulin olemassaolo ei vaikuta tavallisiin käyttäjiin City Rabbits -sivuston puolella.

**Backup and Migrate** -moduuli on myös Drupal -kehittäjille suunnattu. Tämän moduulin avulla vielä keskeneräisestä työstä on yksinkertaista saada varmuuskopiot tai pakata tiedostot ja jatkaa työskentelyä myöhemmin tai toiselta tietokoneelta.

**Views** -moduulin avulla voidaan määritellä, mitkä sivuston sisällöt näkyvät käyttäjille. Esimerkiksi jos sivustolla olisi useampia eri käyttäjätilejä, tämän moduulin avulla olisi mahdollista rajata näkyvää sisältöä käyttäjien mukaan. Toistaiseksi erillaisia käyttäjärooleja sivustolla on käyttäjät ja ylläpitäjät.

#### 4.4 XAMPP

XAMPP on ohjelma, joka mahdollistaa työskentelyn Drupalilla tavallisella internet selaimella, ilman että tarvitsee olla yhteydessä internetiin. XAMPP sisältää Apache HTTP Server ja MySQL -tietokannat ja on rakennettu PHP ja Perl-ohjelmointikielillä. Ohjelma kääntää tiedostot tietokoneen kovalevyllä näkymään selaimelta, kuin ne olisivat olemassa valmiina palvelimella. XAMPP on Apache Friends -projektin tuote. ([apachefriends.org](http://apachefriends.org).)

XAMPP on luonnollinen ohjelmavalinta sivuston tuottamisessa, sillä kyseinen ohjelma oli käytössä opiskelijoiden aikaisemmillä opintojaksoilla koskien Drupal työskentelyä. XAMPPin

etuihin kuuluu myös se, että ohjelma on ilmainen ja helppo käyttää. Ainoa yhteensopivuusongelma joka opinnäytetyön aikana kohdattiin, oli XAMPPin ja Skype puhe ja chat -ohjelman käyttö samanaikaisesti. Kumpikin ohjelma käyttää samoja tietoliikenne-portteja, joten ohjelmat eivät voi olla samaan aikaan päällä.

#### 4.5 cPanel

cPanel on Unix -pohjainen sovellus, jolla voidaan ylläpitää niin sanottuja hosting -palveluita. Se tarjoaa välineet www -sivujen ylläpitäjille, jälleenmyyjille ja loppukäyttäjille. cPanelia voidaan käyttää internetselaimen kautta. cPaneliin siirretään Drupalin tiedostot sekä tietokanta. Tällä palvelimella myös säilytetään internetsivustolla käytettävät kuvat. Sivuston hallittavuuden kannalta on yksinkertaisempaa säilöä kuvat yhteen paikkaan. cpanel otettiin käyttöön, koska toimeksiantaja oli jo ostanut palvelintilaa cPanelilta tulevaa sivustoa varten.

#### 4.6 Sivujen testaus

Vesiputousmallin mukaisesti testaus kuuluu osana prosessia, jota seuraavat käyttöönotto ja ylläpito. Käytännössä sivuston rakentamisessa testausta tapahtui useammassa vaiheessa ja toiminnot testattiin samalla sen asennuksen yhteydessä. Sivustoa on siten rakennettu hyvin iteratiivisesti, jossa edellisen toiminnon on valmistuttava, ennen kuin seuraavan työstäminen voidaan aloittaa. Myös uusien toimintojen lisääminen saattoi toisinaan aiheuttaa konflikteja vanhojen kanssa, jolloin toimintojen testaaminen korostui.

Ulkoasuun liittyvät ongelmat tulivat esille heti muutosten tultua voimaan. Koska ulkoasu on toteutettu valtaosin CSS -koodauksella, virheellinen koodi johti suoraan toimimattomaan osakokonaisuuteen. Ulkoasua pystyttiin tarkastelemaan myös muilla internetselaimilla aina tarpeen vaatiessa. Toimintojen testaaminen toteutettiin samassa yhteydessä yleisen ulkoasun testauksen yhteydessä.

Sivuston toimintoja testattiin useammalla eri tietokoneella ja eri sijainneissa. Myös näyttöjen eri resoluutiot on otettu huomioon ulkoasun suhteen. Muun muassa näytön koko oli huomioitu jo suunnitteluvaiheessa, jolloin sisältö esittäytyi testauksessa suunnitelman mukaisesti. Pieniä eroavaisuuksia havaittiin eri internetselainten välillä, mutta nämä eivät oleellisesti vaikuttaneet sivuston ilmeeseen tai toimivuuteen.

#### 4.7 Ylläpitodokumentin suunnittelu ja toteutus

Käyttöohjetta laadittaessa on ensisijaisesti määriteltävä sen käyttäjät. Sivustoa varten laadittu ylläpidodokumentti on siten osoitettu toimeksiantajalle tai hänen palkkaamalleen jatkokehittäjälle ja ylläpitäjälle. Tämän vuoksi oli vältettävä ammattislangia, joka tietotekniikan parissa työskenteleville on tuttua, mutta muille saattaa olla vaikea ymmärtää.

Asiakkaan pyynnöstä käyttöohje laadittiin englanniksi ja se annettiin tarkastettavaksi useamman kerran, jotta kieli olisi mahdollisimman selkeää. Kulttuurierot toivat oman haasteensa. Nykäsen (2002, 156) mukaan vieraalla kielellä kirjoitettaessa on ajateltava muutakin kuin kieltä. Täten pyrittiin kirjoittamaan mahdollisimman yleismaallista kieltä, jotta kaikki ymmärtäisivät ylläpidodokumentin tekstiä.

Käyttöohjeen tyyliä haettaessa, kirjoitusprosessissa päädyttiin käyttämään yksinkertaisia vaiheita. Myös kirjoitustyylin tuli olla mahdollisimman lyhyttä ja selkeää. Prosessien vaiheet numeroitiin, esimerkiksi ohjelmaa asennettaessa vaihe 1, vaihe 2 ja vaihe 3. Jaottelu vaiheisiin pyrittiin myös tekemään selkeäksi, eli yksi toiminto tai yhden asennus kohdan vaihe kerrallaan. Tavallisesta kappaleisesta tekstistä luovuttiin, sillä siitä käyttäjän olisi vaikeampi löytää yksittäisiä tietoja.

Käyttöohjeen suunnitteluvaiheessa tarkentui kuva siitä, mitä asiakokonaisuuksia sen tuli pitää sisällään. Drupal -sivujen päivittämisestä olisi mahdollista kirjoittaa hyvinkin laaja opas, mutta opinnäytetyön puitteissa päädyttiin kirjaamaan vain ne osat sivuston rakentamisesta ja päivittämisestä, jotka olivat sivuston valmistumishetkellä relevantteja. Koska verkkosivuja tuotettaessa Drupalilla mahdollisuudet ovat lukuisat ja tavallisimpiin ratkaisuihin löytyy runsaasti ohjemateriaalia internetistä, uusien ominaisuuksien lisääminen on käyty läpi vain lyhyesti.

Ylläpidodokumenttiin listattiin myös sivustolla käytetyt moduulit ja niiden toiminnot. Uusien moduulien lataamisen luonteeseen kuuluu se, etteivät kaikki internetistä löytyvät moduulit ole yhteensopivia keskenään. Tämän lisäksi monet niistä ovat harrastelijoiden tuottamia, jolloin yksittäisten moduulien toimivuutta ei voida taata. Mikäli moduulien kanssa ilmenee ongelmia, aktiiviset Drupal -yhteisöt ja keskustelupalstat ovat parhaimpia tiedonlähteitä. Itse ongelmaa on kuitenkin suositeltavaa lähestyä hakukoneiden kautta ja esittää moduulin ongelma suoraan hakukenttään. Yksi suurimmista internetin drupal -yhteisöistä on drupal.org-palvelun foorumit.

Dokumentti luovutettiin sivuston valmistuessa asiakkaalle ja sisältää käytännön ohjeet sivuston luovutushetkellä oleviin toimintoihin ja päivitykseen. Oppaassa on ohjeet uusien moduulien lataamiselle, minkä kautta sivustolle käytännössä on mahdollista tuoda uusia ominaisuuksia. Asiakasta myös neuvotaan käyttöohjeen turvallisesta säilyttämisestä, sillä se sisältää sivuston

hallinnan kannalta salassa pidettävää materiaalia. Tämän opinnäytetyön liitteisiin on tästä johtuen liitetty kyseinen käyttöohje, niin että siitä on poistettu muun muassa käyttäjätunnukset (liite 1).

## 5 Opinnäytetyön haasteet

Opiskelijoilla oli Drupalin käytöstä aikaisempaa kokemusta yhden opintojakson verran. Tähän sisältyivät peruskäsitteet ja ajatus ohjelmapaketin toiminnasta. Opinnäytetyöprosessin aikana tämä osaaminen kehittyi moneltakin osin, mihin opintojakson resurssit eivät olisi riittäneet. Sivustoa tuottaessa teknisten ratkaisujen löytäminen ja ongelmien ratkaisu oli haaste, joka lopulta mahdollisti oppimisen.

Yksi keskeisimmistä haasteista sivuston rakentamisessa oli yleisilmeen säilyttäminen alkuperäisen suunnitelman mukaisena. Drupalilla ja sen moduuleilla oli usein tietynlaiset perusasetukset siitä, millaisena uudet toiminnot näkyvät. Joissakin tapauksissa ulkonäköön oli mahdollista vaikuttaa itse moduulien omien asetusten kautta, mutta valtaosassa tapauksista ulkonäölliset ratkaisut tehtiin CSS -koodin kautta.

Opinnäytetyön toimintasuunnitelmassa oli alunperin pohdittu opinnäytetyön tekemiseen liittyviä riskejä, joista monet liittyivät käytännön tekemiseen. Näistä riskeistä osa myös toteutui, mikä heijastui nopeasti suunnitellun aikataulun venymiseen. Alkuperäisen toimintasuunnitelman heikkouksena oli se, että lopullinen projekti oli suunniteltua laajempi.

Opinnäytetyön toissijaisena lopputuloksena syntyi ylläpitodokumentti. Tämän dokumentin haasteena oli lähteiden löytäminen. Internetistä löytyy pieniä ohjeistuksia manuaalien kirjoittamiseen, mutta kattavaa yleisteoksen kaltaista lähdettä oli vaikeampi löytää. Lisäksi oman haasteensa toi se, että dokumentti päätettiin kirjoittaa englanniksi.

### 5.1 Työskentely

Opinnäytetyö on toteutettu työskentelemällä Laurean tiloissa, mutta myös kotikoneilta. Suurin osa kirjoitustyöstä tapahtui kotikoneilla. Tehdyistä töistä otettiin jatkuvasti backup-tiedostoja jotka sijaitsivat paitsi kannettavilla kovalevyasemilla, myös Dropbox-palvelimella. Tätä kautta aina palvelun ollessa toiminnassa, työt saatiin helposti tarkasteltavaksi paikasta riippumatta.

Varsinaisen sivuston työstämistä hidasti osaamisen kehittäminen työn edetessä. Prosessin edetessä työhön saatiin asiantuntijoilta apua, joista tärkeimmäksi osoittautui palaveri Drupal-

asiantuntija Niko Suomalaisen kanssa. Palaverin jälkeen sivusto rakennettiin sen lopullisen logiikan perusteella, eli valittiin mahdollisimman tyhjä Drupal teema. Tämän teeman päälle rakennettiin sivusto osin CSS-koodauksella, osin kevyillä moduuleilla. CSS-koodaaminen mahdollisti sen, että lopullinen sivusto oli datamäärältään kevyempi, eli nopeammin avautuva tavallisella selaimella. Saman palaverin aikana Suomalainen myös suositteli Fire Bug-lisäosan käyttämistä Firefox selaimen kanssa, josta oli erityistä hyötyä CSS-koodin paikallistamisessa. (Suomalainen 2012)

Kommunikointia varten Facebookiin luotiin omat suljetut ryhmät, jossa valtaosa käytännön kommunikoinnista tapahtui. Ensimmäinen ryhmä, nimeltään Citikanit, oli suljettu ryhmä jonka kautta oltiin yhteydessä asiakkaaseen. Ryhmän jäseniksi oli kutsuttu Elisa Jottunpää ja Riina Jupponen, joka oli ollut mukana alkuperäisen City Rabbits liikeidean kehittämisessä. Opiskelijoiden keskenäiseen kommunikointiin luotiin oma erillinen Facebook-ryhmä, jonka sisällä käytiin kaikki etätyöskentelyyn liittyvä kommunikointi.

Sivuston rakentamisen ohessa luonnollisesti työ dokumentoitiin opinnäytetyöksi ja ylläpidodokumentiksi. Näiden tiedostojen ollessa jaossa Dropboxissa, oli helppo seurata koska opinnäytetyön toinen osapuoli oli päivittänyt tekstiä. Tämän jälkeen toinen luki toisen kirjoittamaa osuutta ja antoi kehitysehdotuksia, korjasi tekstiä ja osallistui tekstin tuottamiseen. Tämän tarkoituksena oli varmistaa molempien opiskelijoiden oppiminen. Paritöissä on yleensä riskinä se, että osatekijät oppivat vain oman osuuteensa.

## 5.2 Riskit ja niiden toteutuminen

Tutkimussuunnitelmaa laadittaessa oli pohdittu joitakin selkeämpiä riskejä, joita opinnäytetyön tekemiseen mahdollisesti liittyisi. Taulukko näistä riskeistä esitettynä taulukossa 1. Samalla arvioitiin näiden riskien toteutumismahdollisuutta asteikolla 1-5, jossa 1 edustaa ei erityisen todennäköistä ja 5 erittäin todennäköistä. Vastaavasti laadittiin asteikko sille kuinka vakava kyseinen riski olisi lopputulokselle, jossa 1 vastasi ei erityisen vakavaa ja 5 erittäin vakavaa. "Miten riskiin reagoitiin" sarakkeessa "-" merkitsee ettei riski toteutunut ollenkaan.

Riski	Riskin todennäköisyys	Riskin vakavuus	Miten riskin toteutuminen estetään	Miten riskiin reagoitiin
-------	-----------------------	-----------------	------------------------------------	--------------------------



Lopputulos ei ole asiakkaiden mieleen	1	5	Säännölliset palaverit asiakkaiden kanssa	-
Aikataulutuspettä	2	4	”Checkpointit”	Aikataulu suunniteltiin uudestaan
Osaamisen kehittäminen	1	3	Haetaan tietoa kirjallisuudesta ja Internetistä, ollaan yhteyksissä asiantuntijoihin, ei jäädä ihmettelemään liian pitkäksi aikaa	Riskin toteutuessa noudatettiin suunnitelman toimintakaavaa
Opinnäytetyön kirjoittaminen ei edisty	2	5	Ei aloiteta kirjoittamista vasta kun itse työ on tehty	Aikatauluun tehtiin muutoksia tarpeen mukaan
Sairastuminen	3	3		Huomioitiin töiden viivästyminen aikataulussa
Tiedostojen katoaminen	1	5	Säännölliset varmuuskopioinnit useampaan paikkaan	Osittain toteutunut, vain vähäisiä määriä tiedostoja jouduttiin tekemään uudestaan
Tekniset ongelmat	2	4	Haetaan ajoissa apua	Riskin toteutuessa noudatettiin suunnitelman toimintakaavaa

Taulukko 1: Tutkimussuunnitelmassa kirjatut mahdolliset riskit ja niiden toteutuminen

Opinnäytetyön aikana toteutuneita riskejä olivat osaamisen kehittäminen, aikataulun pettäminen ja osittain toteutuneet riskit: tiedostojen katoamiset, tekniset ongelmat ja sairastumiset. Riskien toteutuminen ensisijaisesti vaikutti aikataulun viivästyymiseen. Alkuperäisen aikataulutuksen heikkouksina oli loppujen lopuksi se, että projekti oli todellisuudessa laajempi kuin alkuperäisessä suunnitelmassa arvioitiin.

## 6 Johtopäätökset ja jatkokehitys

Sivusto itsessään on toiminnallisuuksiltaan valmis kokonaisuus opinnäytetyön valmistuessa. Sisältö monelta osin tarvitsee vielä tulevan ylläpitäjän lisäämät osuudet, kuten tietosivujen tekstit ja opasprofiilien oikeat tiedot. Opinnäytetyön valmistuessa näiden osuuksien kohdalla on täytetekstiä ja esimerkki -osuuksia. Ohjeet tietojen päivittämiseen ja muokkaamiseen on kirjattu liitteessä 1 olevaan ylläpitodokumenttiin. Koska sivu on toiminnallisuuksiltaan valmis ja liiketoiminta sen osalta vasta alkamassa, sivusto on konkreettinen keino hankkia liikekumppaneita palvelun aloittamiseen. Kuvassa 1 kuvattuna sivuston etusivu.

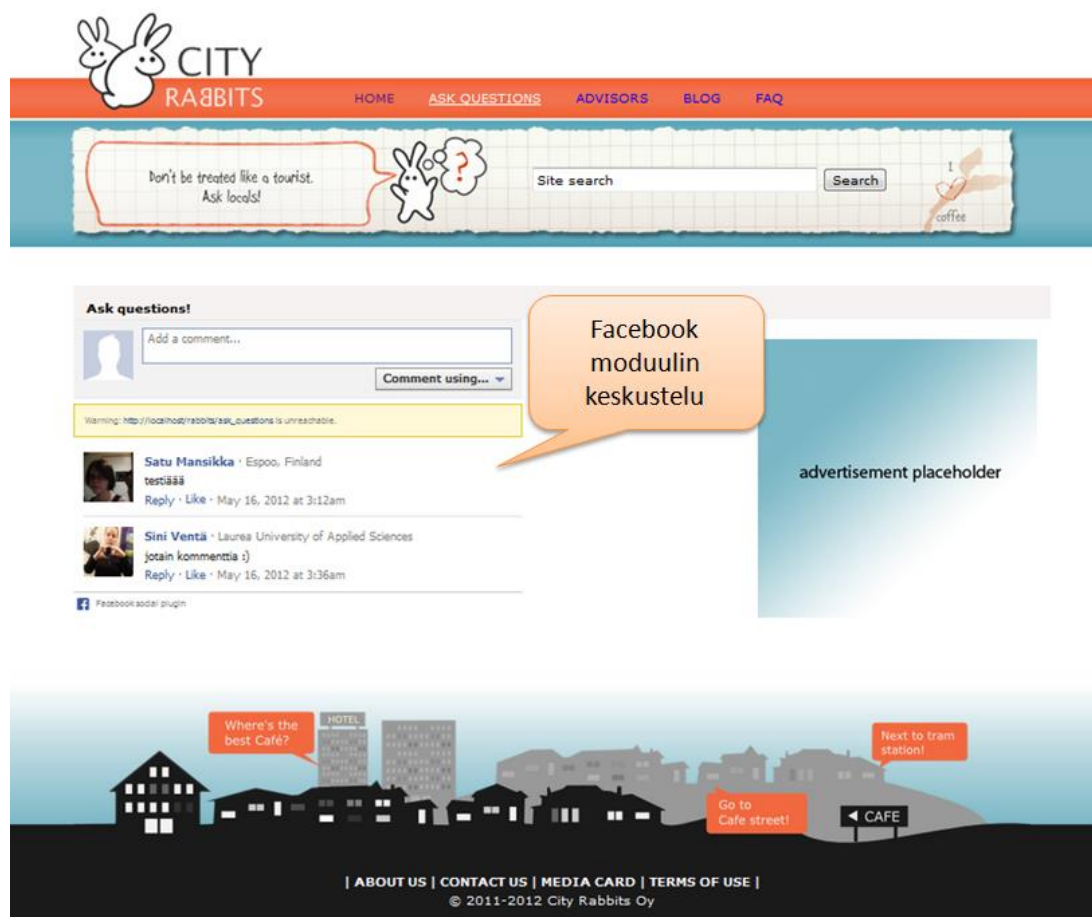


Kuva 1: City Rabbits sivuston etusivu

## 6.1 City Rabbits -sivusto

Jatkokehityksen kannalta siitä vastaavan henkilön on huomioitava, että sivuston kehittäminen vaatii koodaamista Drupalin yhteydessä. Useat ulkoasuun liittyvät ratkaisut on tuotettu CSS ja HTML -koodilla. Toiminnallisuudet on tuotu sivustolle erilaisien moduulien avulla. Näitä palveluja ovat kysymysten esittäminen, opasprofiilit, blogi ja usein kysytyt kysymykset. Muut sivuston sivut ovat yleistä tietoa tai niiden kautta on mahdollista olla yhteydessä sivuston ylläpitäjään.

**Kysy kysymyksiä** (ask questions) sivulla käyttäjä voi kysyä kysymyksiä muilta sivuston käyttäjiltä. Tällä hetkellä sivun kysymyspalsta toimii Facebook -moduulin kautta. Sivua olisi mahdollista kehittää siten, että kysymyksiä voisi luokitella esimerkiksi matkakohteen mukaan ennen sen lähettämistä. Tämä myös vaatisi sen, että sivustolle kerääntyisi vähitellen oma tietokanta kysytyistä kysymyksistä ja vastauksista. Sivuston Facebook-ratkaisu on esiteltyinä kuvassa 2.



Kuva 2: City Rabbits sivuston kysymyspalsta

**Oppaat** (advisors) sivulta löytyvät verkkopalvelun opasprofiilit. Sivuston ylläpitäjän kautta on mahdollista tuoda uusia oppaita sivustolle. Oppaiden esittelyt koostuvat käytännössä vapaamuotoisesta tekstistä, yhteystiedoista ja mahdollisesta kuvasta. Omassa profiilissaan opas voi kertoa omista matkakohteistaan, joista hänellä on palvelun kannalta omakohtaista tietoa. Palvelun käyttäjä voi tämän sivun kautta löytää kohdemaassa asuvan henkilön tai siellä vierailleen matkailijan ja olla häneen yhteydessä. Toistaiseksi uusien oppasivujen lisäämisestä vastaa ylläpito. Jatkokehityksen kannalta oppaille voisi sallia omat tunnukset, joiden kautta profiilin tietoja olisi mahdollista muokata. Kuvassa 3 lista sivustolle luoduista esimerkkiopasprofileista.



Kuva 3: City Rabbits sivuston opasprofiilit

**Blogi** (blog) on ylläpitäjän ja toimeksiantajan kanava, jonka kautta erityisesti sivuston alkuvaiheessa on mahdollista tuottaa uusia blogi -viestejä. Muut sivuston käyttäjät, esimerkiksi oppaat, voivat tuottaa sisältöä blogiin. Näiden julkaiseminen onnistuu kuitenkin vielä ylläpidon kautta, sillä sivustolle eivät muut käyttäjät kykene tekemään uusia käyttäjätilejä. Myöhemmin sivuston muut käyttäjät voisivat osallistua blogin kirjoittamiseen esimerkiksi kommentoimalla.

**Usein kysytyt kysymykset (FAQ)** sivu on tekniseltä toteutukselta ainoastaan tekstiä sisältävä sivu, mutta se sisältää palvelun kannalta usein kysytyt kysymykset. Aluksi sivulle on tarkoitus tuoda palvelun kannalta keskeisiä kysymyksiä, mutta ylläpidolla on mahdollisuus päivittää sen sisältöä kun kanssakäymistä käyttäjien kanssa ilmenee.

Sivustosta luotiin opinnäytetyön aikana ainoastaan englanninkielinen versio. Toimeksiantaja ei nähnyt tarpeelliseksi luoda useamman kielen internetsivustoa. Kielivaihtoehdot tulevat ajankohtaiseksi, mikäli toiminnasta tulee kannattavaa ja useammille kielille tulee kysyntää. Rajauksia sivuston ulkoasuun tapahtui, kun jokin toiminto ei ollut mahdollista saada toimimaan luotettavasti. Esimerkkinä mainittakoon Google Maps, jonka oli määrä sisältyä piti palveluun tuomassa lisäominaisuuksia palvelulle.

Sivusto on mahdollista muuttaa niin sanottuun huoltotilaan, joka on päivitystilanteita varten. Sivusto on myös tarkoitus pitää huoltotilassa, kunnes sisällöt ovat enimmäkseen valmiita. Huoltotilassa verkkosivuston etusivulle ilmestyy viesti, joka kertoo käyttäjälle, ettei sivusto

ole käytössä. Tässä tilassa tavallisilta käyttäjiltä on estetty pääsy muille sivuille. Ylläpitodokumentin ohjeiden mukaan toimeksiantaja kykenee julkaisemaan sivuston, jolloin sen palvelut aukeavat tavallisille käyttäjille. Sivuston luovutushetkellä sivusto on siinä tilassa, että toimeksiantaja voi helposti esitellä sitä mahdollisille rahoittajille.

## 6.2 Kehitysehdotukset

Sivuston data luovutetaan opinnäytetyön päätösvaiheessa asiakkaalle ja tätä tietoa muokkaamalla on mahdollisuus hallinnoida sivuston ominaisuuksia, esimerkiksi tuomalla uusia toiminnallisuuksia sivustolle tai muuttamalla ulkoasua. Internetissä tämä sisältö näkyy kun sivuston data päivitetään palvelimelle.

Selkeimmäksi kehityskohteeksi voidaan nimittää sivuston kysymyspalstan toteutus. Asiakkaan kanssa käytiin keskustelua pidempäänkin siitä miten keskustelu tulisi järjestää. Alunperin haluttiin pitää järjestelmä mahdollisimman helppona muun muassa niin, ettei palveluun tarvitse kirjautua. Siitä huolimatta keskustelulle täytyi olla jokin kontrolli, sillä vapaat kirjoituskentät voivat nopeasti täytyä spam -viesteistä. Alunperin oli tarkoitus luoda sivustolle oma tietopankki aikaisemmin kysytyistä kysymyksistä ja vastauksista. Facebook -ratkaisun kannalta Facebookin omat käyttöehdot kieltävät Facebookin kautta käytyjen keskustelun tietojen tallentamisen kolmannelle osapuolelle. Tähän yksi ratkaisu olisi omien foorumien tekeminen sivustolle, mutta ajatus hylättiin sen monimutkaisuuden vuoksi. Keskustelun käyminen ja tärkeän informaation taltioiminen jää siten jatkokehityksen puolelle jossa kontrolli ja vapaa keskustelu on saatu tasapainoon.

Google maps toiminnosta puhuttiin opinnäytetyön aikana asiakkaan kanssa palvereissa. Toimintoa yritettiin tuoda sivustolle useaan otteeseen erilaisten moduulien avulla, mutta projektin aikana näistä mikään ei rakentunut riittävän vakaalle pohjalle, jotta kartat olisivat näkyneet moitteettomasti. Google maps palvelun tuominen sivustolle lopulta hylättiin, koska tavoitteena oli saada toimiva sivusto, jossa olisi keskeisimmät toiminnot olemassa. Google maps palvelu päätettiin näin jättää jatkokehityksen puolelle.

Mobiilisivujen tekeminen myöhemmässä vaiheessa saattaa nousta ajankohtaiseksi kun palvelu myöhemmin kenties menestyy. Mobiililaitteiden näytöt ovat resoluutioltaan paljon pienempiä, jolloin tavallisten tietokoneruudulle suunniteltujen internetsivujen selaaminen ruudulla on paljon hankalampaa. Mobiiliratkaisuina voisi toimia perinteisesti vain uusien rinnakkaissivujen luominen mobiilimuotoon tai oman mobiilisovelluksen tekeminen. Modiilisovellusten tapauksessa olisi tarpeellista tarkastella uudelleen mitä palveluita sovelluksen kautta olisi mahdollista tarjota.

Puhuttaessa käyttäjien paluusta palveluun uudelleen, kysymykseen nousevat koukuttavuutta luovat tekijät. Pelillinen tyyli voitaisiin lisätä sivustolle eräänlaisena palkitsemismenetelmänä. Esimerkkinä tästä käyttäjät voisivat palvelussa kerätä pisteitä omalla aktiivisuudellaan. Tai sivuston oppaat kykenisivät samalla tavalla keräämään kokempisteitä omasta aktiivisuudestaan sivustolla. Kumpikin vaihtoehto vaatisi jonkinlaista käyttäjän tunnistamista, jolloin kyseeseen nousisi uudestaan tarve erilaisten käyttäjäroolien luomisesta.

Sivuston on alunperin toteutettu Drupal 7 -versiolla ja jatkokehityksen ajankohta on vielä määrittelemättömässä tulevaisuudessa. Tästä johtuen jatkokehitys voidaan tehdä myös uudemmilla Drupal versioilla. Opinnäytetyön aikana Drupal 8 versiosta oli tieto, että se on tulossa vuoden 2013 jälkimmäisellä puoliskolla. Tämän uuden version pitäisi mahdollistaa internetsivujen rakentamisen niin, että ne toimivat sujuvasti myöskin mobiililaitteissa. Moduulien yhteensopivuutta on tähän versioon paranneltu ja Drupal 8 ottaa käyttöön myös HTML5 ohjelmointikielen. (Drupal.org 2013.)

Drupal versiosta päivittäminen seuraavaan versioon on käytännössä mahdollista. Tässä siirtymisprosessissa on kuitenkin useita mahdollisuuksia aiheuttaa vahinkoa alkuperäiselle tiedolle, joten ennen siirtoprosessia on hyvä ottaa alkuperäisistä tiedostoista varmuuskopiot. Tämä Drupal version päivittäminen muodostuu pakolliseksi vain siinä tapauksessa jos sivustolle vaadittavien uusien toiminnallisuuksien uudet moduulit tukevat ainoastaan uusinta Drupal versiota. Drupal version päivittäminen määriteltiin opinnäytetyön rajauksissa kuuluvaksi työn ulkopuolelle johtuen siitä, ettei tietoa Drupal 8 versiosta ollut saatavilla riittävästi. Uusien Drupal versioiden tullessa markkinoille, internetin Drupal -yhteisöt yleensä päivittävät ohjeistuksensa koskemaan uusien versioiden päivitystä. Esimerkiksi Drupal 6:sta päivittäminen 7 versioon on neuvottu Drupalin virallisilla sivuilla (Drupal.org). Tällä ohjeistuksella pyritään siihen, ettei valmista sivustoa olisi tarpeellista uudelleen rakentaa.

### 6.3 Yhteenveto

Opinnäytetyö on toteutettu asiakkaan kanssa sovitun tavoitemäärittelyn mukaisesti ja tekniset tavoitteet on saavutettu. Tässä mielessä projekti on onnistunut ja sen tuotokset ovat olleet hyödyllisiä kaikille osapuolille. Toimeksiantaja kykenee sivuston avulla suunnittelemaan liiketoiminnan aloittamista ja yhteistyökumppaneita. Palaute sivuston rakentamisen edetessä on ollu pääosin hyvää ja rakentavasta palautteesta on kehitetty palvelusta toimeksiantajan toiveiden mukainen.

Opiskelijat puolestaan ovat oppineet verkkosivuston rakentamista Drupalilla käytännössä. Tämä on ennen kaikkea syventänyt Drupal ja koodausosaamista. Kirjoitusprosessi on

vahvistanut vanhaa osaamista, jossa tältä osin taitoja täytyi soveltaa pitkän asiatekstin tuottamiseen. Asiakkaalle luovutetun englanninkielisen ylläpitodokumentin saralta opittiin käyttöohjeen laatimista sekä pidemmän tekstin laatimista englanniksi.

Koko opinnäytetyöprosessin osalta työhön sovellettiin prosessien malleja, joista monet osat olivat lopputuloksen kannalta itsestäänselvyyksiä. Raportoidessa kuitenkin näistä saatu hyöty nousi esille eri näkökulmien huomioimisessa sekä loppuvaiheessa. Lopputulosta tarkistettiin useampia kertoja varmistellen, että se vastasi alkuperäisen suunnitelman tavoitteita. Huolimatta siitä, että aikataulutusta suunniteltiin useaan otteeseen opinnäytetyön aikana uudestaan, lopputuloksen kannalta kiirehtiminen ei olisi tuottanut vastaavaa tulosta. Uusien taitojen ja oppimisen kannalta yhteistyö toimeksiantajan kanssa tarjosi arvokkaan kokonaisuuden. Nämä taidot tulisivat käyttöön myöhemmin seuraavan koulutuksen yhteydessä tai työelämässä, jossa oppiminen tulee jatkumaan edelleen.

## Lähteet

Korpela, J. & Linjama, T. 2005. Web-suunnittelu. 2. uudistettu painos. Porvoo: WS Bookwell.

Nykänen, O. 2002. Toimivaa tekstiä: Opas tekniikasta kirjoittaville. Helsinki: Tekniikan Akateemisten Liitto TEK.

Pohjonen, R. 2002. Tietojärjestelmien kehittäminen. Vantaa: Docendo.

Sinkkonen, I. 2009. Helppokäyttöisen verkkopalvelun suunnittelu. Helsinki: Tietosanoma.

Sinkkonen, I., Nuutila, E. & Törmä, S. 2009. Helppokäyttöisen verkkopalvelun suunnittelu. Helsinki: Tietosanoma.

Verkkotekniikka + Training Kit. 2000. Microsoft. Suomennos Mäenpää, Y. Jyväskylä: IT Press.

## Sähköiset lähteet

Afterdawn. Synkroinointi. Viitattu 2.4.2013.  
<http://fin.afterdawn.com/sanasto/selitys.cfm/synkroinointi>

ApacheFriends. XAMPP. Viitattu 24.4.2013. <http://www.apachefriends.org/en/index.html>

Builtwith.com. 2013. CMS Usage Statistics, Overview of statistics for CMS technologies. Viitattu 23.3.2013. <http://trends.builtwith.com/cms>

Drupal 8 Updates and How to Help. 2013. Drupal.org. Viitattu 8.3.2013.  
<http://drupal.org/community-initiatives/drupal-core>

Grammarly.org. Viitattu 1.4.2013. <http://www.grammarly.com/>

Joutunpää, E. 2012a. Verkkopalveluliiketoiminnan toteutussuunnitelma. Viitattu 16.4.2013.  
<https://publications.theseus.fi/bitstream/handle/10024/40650/Elisa.Joutunpaa.2012.pdf?sequence=1>

Juselius, U. 2004. Typografia. Viitattu 18.5.2013.  
<http://www.phpoint.fi/ulrikaj/www/typo.htm>

Lukka, K. Konstruktiivinen tutkimusote. Viitattu 5.4.2013.  
[http://www.metodix.com/fi/sisallys/04\\_virtuaalikirjasto/dokumentit/aineistot/konstruktiiivientutkimusote](http://www.metodix.com/fi/sisallys/04_virtuaalikirjasto/dokumentit/aineistot/konstruktiiivientutkimusote)

Tolvanen, P. 2008. Julkaisujärjestelmät Suomessa, markkinakatsaus 2008. Viitattu 7.3.2013.  
<http://vierityspalkki.wordpress.com/2008/03/31/julkaisujarjestelmat-suomessa-markkinakatsaus-2008/>

Upgrading from Drupal 6 to 7. 2012. Drupal.org. Viitattu 7.3.2013.  
<http://drupal.org/documentation/upgrade/6/7>

W3Schools- verkkosivusto. Browser Statistics. Viitattu 3.3.2013.  
[http://www.w3schools.com/browsers/browsers\\_stats.asp](http://www.w3schools.com/browsers/browsers_stats.asp)

WordPress.org. 2013. Viitattu 23.3.2013. <http://wordpress.org/>



#### Julkaisemattomat lähteet

Joutunpää, E. 2012b. Liikeidean kirkastaminen ja palvelun suunnittelu. Laurea-ammattikorkeakoulu. Laurea Leppävaara. Espoo. Raportti.

Pesu, J. 2012. Drupal -asiantuntijan haastattelu 17.4.2012. Laurea-ammattikorkeakoulu. Laurea Leppävaara. Espoo. Muistio.

Suomalainen, N. 2012. Drupal -asiantuntijan haastattelu 20.4.2012. Laurea-ammattikorkeakoulu. Laurea Leppävaara. Espoo. Muistio.

## Sanasto

**Alfa -sivut, beta -sivut** ovat termejä joita käytetään verkkosivuista ennen kuin niitä lasketaan yleisesti käyttöön tai markkinoitavaksi. Beta-sivut (myös käytetään nimitystä beta -versio) on yleisemmin käytössä, joka ei ole vielä täysin valmis sivu, mutta se on julkaistu ja sille toivotaan käyttäjiä, jotka antavat palautetta jo olemassa olevista toiminnoista. Beta -sivujen ulkoasu ei välttämättä vastaa vielä valmista, mutta suuri osa sen toiminnallisuuksista on jo käytössä. Alfa -sivut ovat beta -sivuja varhaisempi versio, joita harvemmin annetaan käyttäjien kokeiltavaksi. Alfa -sivut loogisesti ovat paljon varhaisemmassa vaiheessa ja sisältävät kenties vain osittain toimivia osia. Palautteen saaminen ja sitä kautta mahdollisten bugien ja virheiden löytäminen on alfa- ja beta -sivujen julkaisun perusajatus.

**CSS, HTML, Java** ovat ohjelmointikieliä. Ohjelmointikielillä kirjoitetaan kaikki tietokoneohjelmat ja verkkosivut toiminnallisuuksista ulkoasuun, jotta loppukäyttäjän ei tarvitse keskittyä koodin osaamiseen vaan siihen, mihin kyseistä ohjelmaa tai palvelua käytetään. Eri ohjelmointikielien on luotu aikanaan hivenen eri tarkoitusta varten ja niiden logiikka ja kirjoitustapa vaihtelee. Kielet on kuitenkin sen verran yhteensopivia, että yhdellä sivustolla voi olla osia jotka on rakennettu eri kielillä kirjoitettuun lähdekoodiin, mutta kokonaisuus on silti toimiva.

**Dropbox** on tiedostojen synkrointisovellus, joka mahdollistaa tiedostojen synkronoinnin (katso synkrointi) eri laitteiden välillä. Tiedostojen jakaminen on helppoa muiden kanssa tämän sovelluksen avulla. Opiskelijat käyttivät ilmaista versiota, jossa on vähemmän tallennustilaa kuin maksullisessa.

**Drupal** on sisällönhallintajärjestelmä, jolla on mahdollista luoda verkkosivuja. Ohjelma on maksuton ja ladattavissa sen omilta kotisivuilta tietokoneelle. Drupalin etuihin kuuluu mahdollisuus muokata sivuista erityistoiveiden mukaisia CSS -koodauksen ja useiden modulien kautta.

**Fire Bug** on Mozilla Firefoxille ladattavissa oleva lisäosa. Lisäosan avulla Drupal sivuston ollessa auki selaimella on mahdollista rajata elementtejä sivustolta (esimerkiksi otsikoita ja kuvia) ja Fire Bug antaa listan ehdotuksia, missä Drupal tiedostojen osassa sijaitsee CSS -tiedostoja, jotka vaikuttavat kyseisen elementin ominaisuuksiin. Lisäosa on tarkoitettu apuvälineeksi Drupalin kanssa työskenteleville. Fire Bug toimii sellaisenaan ainoastaan Firefox-selaimella. Kuitenkin esimerkiksi Google Chromelle on olemassa **Fire Bug -lite**, joka on lähes samanlainen lisäosa Chromen käyttäjille. Kyseessä on kuitenkin vain lite -versio, eli toiminnallisuuksiltaan rajoitetumpi versio.

**Footer** on verkkosivuston osana headerin vastakohta, eli sivuston alalaita tai alatunniste. Aivan kaikilla sivuilla ei ole footeria tai se on erittäin kapea. Footerin sisältöön ei ole yhtä paljon sääntöjä olemassa, kuin headeriin. Esimerkiksi footerit voivat sisältää navigointipalkin linkit toiseen kertaan, tai joitakin "vähemmän tärkeitä" linkkejä, jotka eivät keskeisesti liity sivuston tarjoamaan palveluun (esimerkiksi sivuston ylläpitäjän yhteystiedot). Footer usein myös sisältää copyright merkinnät.

**Header** on verkkosivuista puhuttaessa sivuston yläreuna tai ylätunniste, joka yleensä sisältää palvelun tai yrityksen logon ja sivuston otsikon. Headeri on usein sijoitettu perinteisen ylärivin navigointipalkin yläpuolelle.

**Joomla!** on sisällönhallintajärjestelmä ja Drupalin kilpailija samoilla markkinapaikoilla.

**Julkaisujärjestelmä** on yleinen termi ohjelmalle tai verkkopalvelulle, jolla on mahdollista luoda internet-sivu, sivusto tai blogi. Julkaisujärjestelmä on sisällönhallintajärjestelmästä yksinkertaistettu versio, jolla on mahdollista tuottaa yksinkertaisia sivuja, joilla ei ole erityisiä vaatimuksia ulkoasun suhteen. Näitä ovat esimerkiksi harrastelijoiden kotisivut ja blogit.

**Lähdekoodi** on sivuston data koodikielen muodossa. City Rabbits-verkkosivuston tapauksessa valtaosa sivuston koodista on kirjoitettu CSS-koodilla. Lähdekoodi on sivuston perusta, joka määrittelee sen mitä sivustolla on, miltä se näyttää ja miten se toimii.

**Moduulit, Drupal-moduulit** ovat Drupaliin liitettäviä lisäominaisuuksia, joita usein harrastelijat tai ammattikoodaajat julkaisevat internetissä. Moduulien kautta omalle Drupal sivustolle on mahdollista tuoda erilaisia toiminnallisuksia, jotka eivät kuulu Drupal-ohjelman peruspakettiin. Moduulien lataamista voi myös puolustaa siltä kannalta, että uuden moduulin toiminnallisuus on rakennettu paremman logiikan varaan tai toiminto sopii paremmin sivuston yleisilmeeseen.

**MySQL** on relaatiotietokanta, jossa tietokannan tiedot ovat yhteydessä toisiinsa. Internetsivuston tiedot ja data tallennetaan tietokantaan, ns. omaan tietovarastoon.

**Navigointipalkki** verkkosivuilla on usein joko lista linkkejä sijoitettuna joko sivuston vasempaan laitaan muun sisällön ulkopuolelle tai rivinä sivuston sisällön ja headerin välissä. Navigointipalkkiin usein kootaan sivuston sisällön kannalta keskeisimmät linkit.

**Perl** on skriptimäinen ohjelmointikieli, jolla on kirjoitettu osa XAMPP-ohjelman skripteistä yhdessä PHP-kielen kanssa.

**PHP** on ohjelmointikieli, jota käytetään erityisesti internetsivujen rakentamiseen. PHP:tä käytetään myös tietokantojen rakentamiseen, kuten tämänkin opinnäytetyön yhteydessä. PHP on osana XAMPP-ohjelmaa jonka avulla luodaan tietokanta rakennettavalle sivustolle.

**Plone** sisällönhallintajärjestelmä ja Drupalin kilpailija samoilla markkinapaikoilla.

**Sisällönhallintajärjestelmä** on yleinen termi ohjelmille tai verkkopalveluille, joilla on mahdollista luoda internetsivu, sivusto tai blogi. Ominaisuudet ja muokkausmahdollisuudet vaihtelevat eri ohjelmien välillä suuresti. Yhteistä näille ohjelmille on se, että ne ovat maksuttomia. Usein kuitenkin sivuston julkaiseminen vaatii serverin, jonka avulla sivu näkyy verkossa ja se on yleensä maksullinen.

**Synkronointi** -sanalla tietotekniikan parissa tarkoitetaan usein kahden eri järjestelmän saattamista samaan tilaan (Afterdawn). Dropbox -palvelussa useilla käyttäjillä voi olla pääsy samaan kansioon ja käyttäjät pystyvät muokkaamaan, lisäämään tai poistamaan tietoja. Palvelu sitten synkronoi tiedot, että yhden käyttäjän tekemät muutokset näkyvät muillekin.

**Tekstipalsta** internetsivustolla tarkoittaa tekstiä sisältävää aluetta. Sivuston tekstipalstat on usein jäsenelty samaan tyyliin kuin esimerkiksi sanomalehtien tekstipalstat, mutta internetsivulla näitä palstoja on vierekkäin yleensä vain yksi tai kaksi. Tekstipalstat edustavat usein sivuston sisältöä.

**Tietoliikenne portti**, **portti** on numerolla varustettu tietoliikenteen osa jonka kautta tietokonesovellukset ovat yhteydessä verkon välityksellä toisiin tietokoneisiin. Porttinumero on sovelluksen oma osoite lähetettäessä ja vastaanottaessa dataa verkon kautta toiselta samanlaiselta sovellukselta. Data itsessään vaatii verkon kautta välittymiseen kohde tietokoneen IP-osoitteen, mutta porttinumero kertoo mihin sovellukseen kyseinen data on tarkoitettu.

**Windows Taskbar** (suomeksi **Windows Tehtäväpalkki**) on normaali Windows-asetuksilla ruudun alalaidasta löytyvä kapea palkki, joka sisältää väliaikaiset pikakuvakkeet käynnissä oleviin ohjelmiin. Tehtäväpalkista löytyvät myös normaalisti muun muassa kello ja ääniasetukset.

**XAMPP** on ohjelma, joka mahdollistaa Drupalilla työskentelyn internetselaimessa.

## Kuvat

Kuva 1: City Rabbits sivuston etusivu .....	26
Kuva 2: City Rabbits sivuston kysymyspalsta.....	27
Kuva 3: City Rabbits sivuston opasprofiilit .....	28

## Kuviot

Kuvio 1: Ensimmäisiä luonnostelmia sivustosta .....	8
Kuvio 2: Prototyypilähestymistapa (Pohjonen 2002, 41) .....	10
Kuvio 3: Käytettävyys (Sinkkonen 2009, 21) .....	12
Kuvio 4: Riina Ruppösen luonnostelma sivuston graafisesta ilmeestä .....	14
Kuvio 5: Sivuston tietohierarkia .....	15
Kuvio 6: Internet selainten käyttö maailmalla tammikuussa 2013 (W3Schools.com) .....	17
Kuvio 7: Sisällönhallintajärjestelmien käyttö 2013 (Builtwith.com) .....	19

## Taulukot

Taulukko 1: Tutkimussuunnitelmassa kirjatut mahdolliset riskit ja niiden toteutuminen .. 25

Liitteet

# **Administrators' guide to City Rabbits online service**



## Table of Contents

1	Introduction.....	42
2	Preparing to install .....	42
3	XAMPP Installation .....	43
4	After the installation.....	44
	4.1 XAMPP start-up and potential problems .....	44
	4.2 Drupal 7 installation .....	45
	4.3 Creating the database with phpMyAdmin.....	46
	4.4 Drupal site configuration and confirming the database .....	48
	4.5 Addresses and other functions.....	51
5	Backuppping.....	51
	5.1 Taking a backup from the Database (mySql).....	51
	5.2 Restoring the database .....	52
	5.3 Drupal.....	53
6	Modules .....	53
	6.1 Modules used in Drupal .....	54
	6.2 Installing modules.....	54
7	Content creation and editing .....	54
	7.1 Create a new page.....	55
	7.2 Editing an existing page .....	57
	7.3 Linking the new sub -page to the existing pages, editing HTML code.....	59
	7.4 Finding content.....	61
	7.5 Creating new blog entry .....	62
8	Web Hosting Management: Cpanel .....	63
	8.1 Adding pictures to cPanel .....	64
	8.2 Uploading the Drupal files to the server.....	<b>Error! Bookmark not defined.</b>
9	Updating.....	679
	9.1 Updating Drupal versions .....	67
	9.2 Updating modules.....	67
	9.3 The maintenance mode .....	67
	References.....	69
	Screenshots .....	70

## 1 Introduction

This guide is meant for City Rabbits website administrators who are responsible for monitoring and updating the sites. As the sites were originally built with Drupal, the guide includes step-by-step advice for updating site content and working with the online server. This document also includes instructions for XAMPP and Drupal installation. Drupal is the content management system which was used to create the web service City Rabbits. XAMPP is a program that allows Drupal to display the web page as it would look when published. In the guide there is also information about how to make backup files and update the content. At the end of the document, there are instructions on how to add new content to the actual sites of City Rabbits and how they are updated.

The City Rabbits web service is intended for travellers who are interested in the information locals can offer about their travel destination. Through the embedded Facebook module users can ask questions about the city or country they are interested in travelling to. In addition to online communication, there are advisor profiles in the sites to contact them for questions about several locations. City Rabbits service can be found at [cityrabbits.com](http://cityrabbits.com).

City Rabbits is a service that was originally planned and designed in a thesis by Elisa Jouttunpää and developed as a functioning service in a thesis by Satu Mansikka and Sini Ventä. The project consisted of building the sites and planning its functioning using Drupal. Also included was the writing of this administrator document for later use and the final thesis report about the work. This is the Administrator's document for those who will manage the City Rabbits web sites after the thesis project is over.

## 2 Preparing to install

Before the installation of a few core applications, the administrator should ensure that the correct version of XAMPP and Drupal is chosen. The City Rabbits web sites were built with Drupal 7 version and done within the Windows environment.

- Downloading XAMPP and Drupal is needed only when building sites from the start. The site content can be updated without these programs and this is described at chapter 7.
- Drupal can be downloaded at [Drupal.org](http://Drupal.org)
- XAMPP can be downloaded at [Apachefriends.org](http://Apachefriends.org)

XAMPP can be downloaded for Windows, Linux, Mac OS X and Solaris from the Apache Friends home site under XAMPP download section. This program allows the user to create website content that is saved on the hardware of a computer. XAMPP makes sites look like the actual sites, when opened in any web browser. This way it is possible to create wholly complete web sites before publishing them. On the installing site, there are several means to download XAMPP. It is recommended to use the installation program instead of .zip-files.

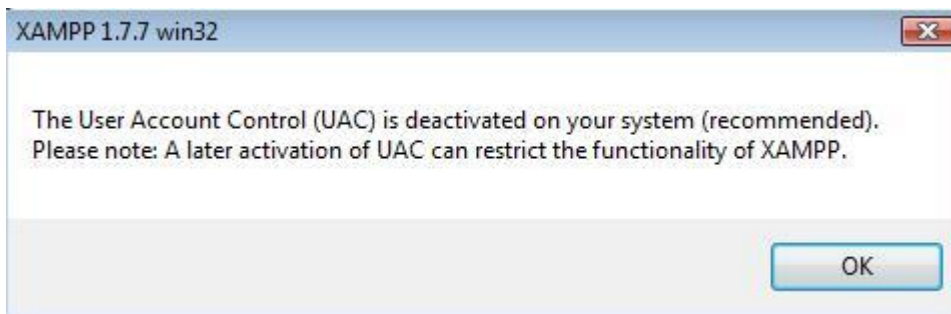
Drupal is a program for creating websites. The program will not function alone, but the remaining programs needed are covered with the XAMPP installation. XAMPP is combination of programs that enables creation of Drupal databases. Databases contain all the information needed to build sites with Drupal. Also the same databases are uploaded on a web server when the sites are intended to be published. A common version of Drupal still in use is Drupal 6 but the City Rabbits web site is done with the latest version, Drupal 7. There are no special requirements for Windows or other operating systems, and the same installation is meant to work on all platforms.

### 3 XAMPP Installation

**Step 1:** After XAMPP is downloaded, installation can be started by clicking the installation file.

**Step 2:** To continue, the language must be chosen. In this thesis, XAMPP was configured with an English interface.

**Step 3:** The XAMPP Installer notifies about the User Account Control (UAC). The message depends on the operating system used. The message as seen in screenshot 1 is from Windows XP.



Screenshot 1: XAMPP notification

**Step 4:** Windows Firewall may notify that the file does not come from a trusted source. The program should be allowed to run this program.

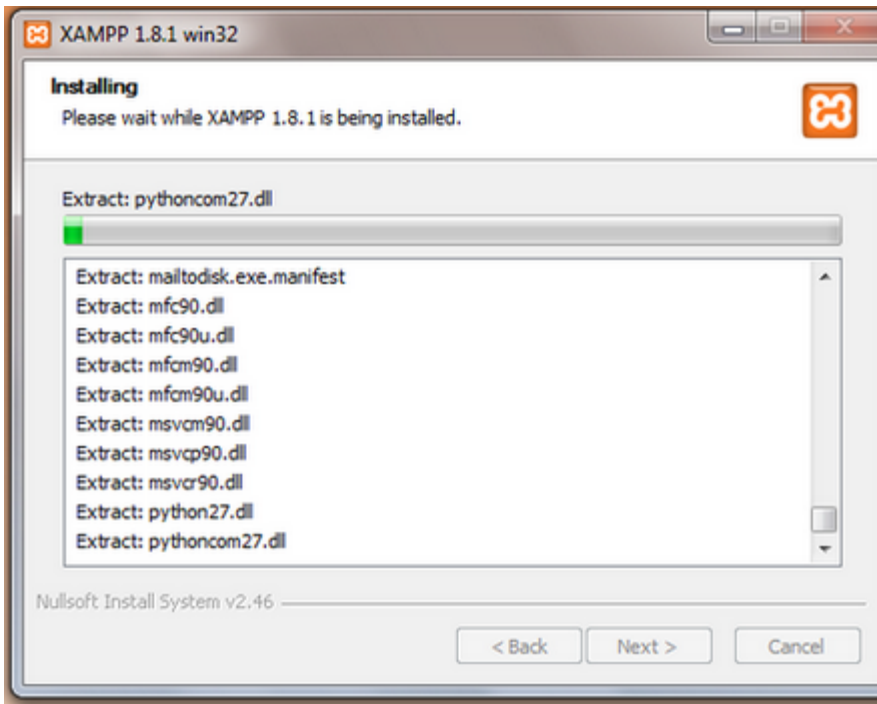
**Step 5:** The installation program will ask a location to save the files. C:\xampp is recommended as the installation folder.

**Step 6:** Allow the installer to install the following:

- XAMPP
- Server
- Program languages
- Tools

The list of installed programs may vary depending on the version of XAMPP. You can install every program the downloader suggests, but programs listed above are required.

**Step 7:** From screenshot 2 it can be seen what the installation progress will look like. The screenshot was taken in a Windows XP environment.



Screenshot 2: Installation

**Step 8:** After the installation program has finished, click "Finish". XAMPP asks if you want to start the Control Panel now. Click "Yes". The XAMPP Control Panel, as seen in screenshot 3, should appear on the bottom -right of the screen.

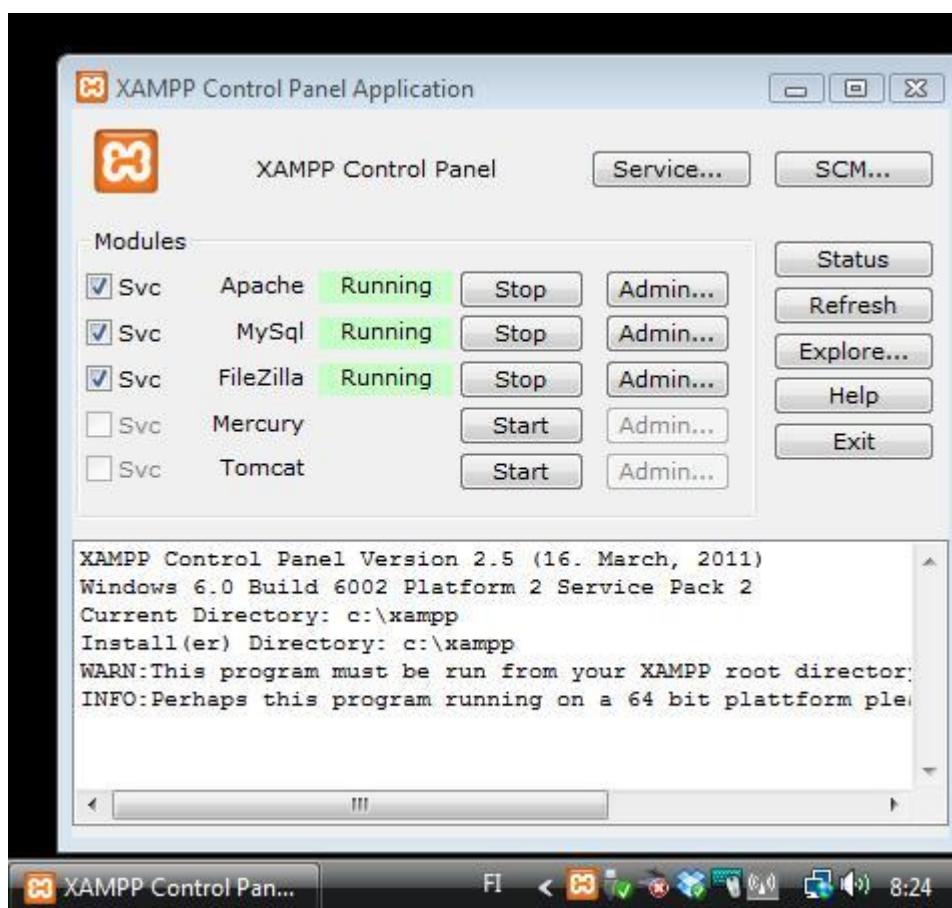
#### 4 After the installation

After the software is installed, the general settings can be adjusted. XAMPP is ready to use without configuration, but Drupal requires profile settings and a database should be created for the website files.

##### 4.1 XAMPP start-up and potential problems

If the XAMPP Control Panel is not open, it can be found on the Windows Taskbar, on the right side beside the clock. The bright orange button with a white bone on it should be easily recognized. If the icon is not there, XAMPP can be started from its own folder in the C drive (C:\xampp).

For Drupal to work, Apache and MySQL should be activated on the XAMPP control panel. As can be seen from screenshot 4 the green block beside the button refers if the program is running. Apache is web-server software program functioning as a part of Drupal. MySQL is the database for all data regarding the sites.



Screenshot 3: XAMPP control panel

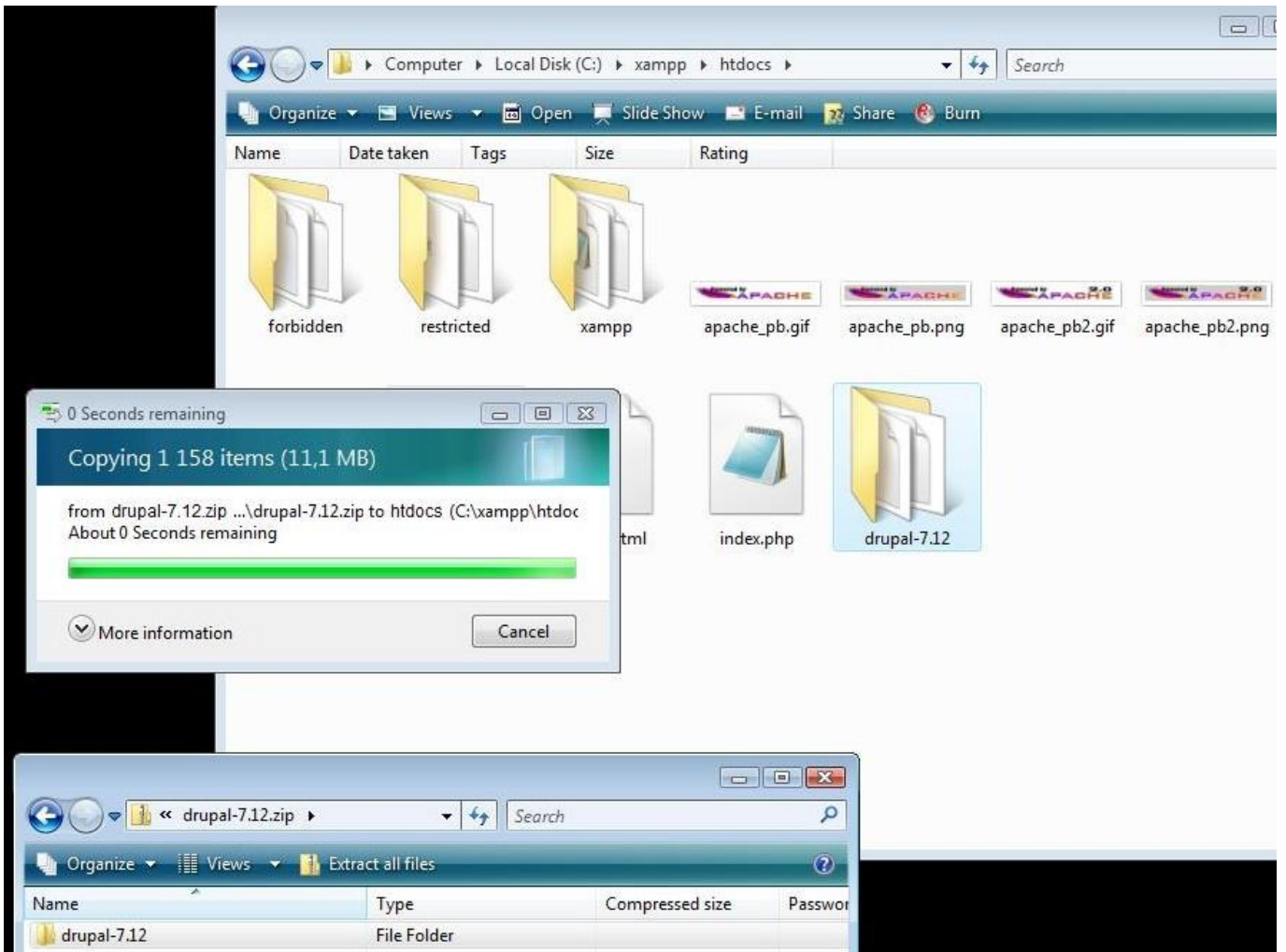
In the thesis project, there was a compatibility issue with the program Skype. XAMPP and Skype use the same ports, and while one is installed in the computer, the other cannot function properly. For XAMPP to work, it is advised to uninstall Skype if downloaded. Uninstalling Skype does not remove friends from the Skype account, and the program should remain the same when reinstalled. However, the latest version of Skype allows the user to use XAMPP simultaneously if the user is logged out from Skype.

In the thesis project was encountered several times with the XAMPP Control Panel in that Apache or MySQL would not start when clicking "start" at the Control Panel. Or the program would start running for a few seconds before closing again. To fix this, the program should be closed and restarted from the C drive. Notice: by clicking "X" (close) only closes the Control Panel and leaves XAMPP running. The program should always be closed from "Quit". If the problem remains, uninstall XAMPP from Windows Control Panel and reinstall XAMPP. Windows Control Panel can be accessed through Windows Start menu.

Possible problems encountered in Linux, Mac OS X or Solaris systems are beyond the scope of this document.

#### 4.2 Drupal 7 installation

Before starting the Drupal configuration, the "Drupal-7.12" folder (screenshot 4) should be renamed to be the same as the main database name. In this thesis project the name "rabbits" was used to identify the database from the others. Unless the name is changed, Apache cannot find files for the website in the computer. After this is done, the Drupal configuration can start. Folder "rabbits" is also the folder for copying for backup files from Drupal when there is a need to move files.



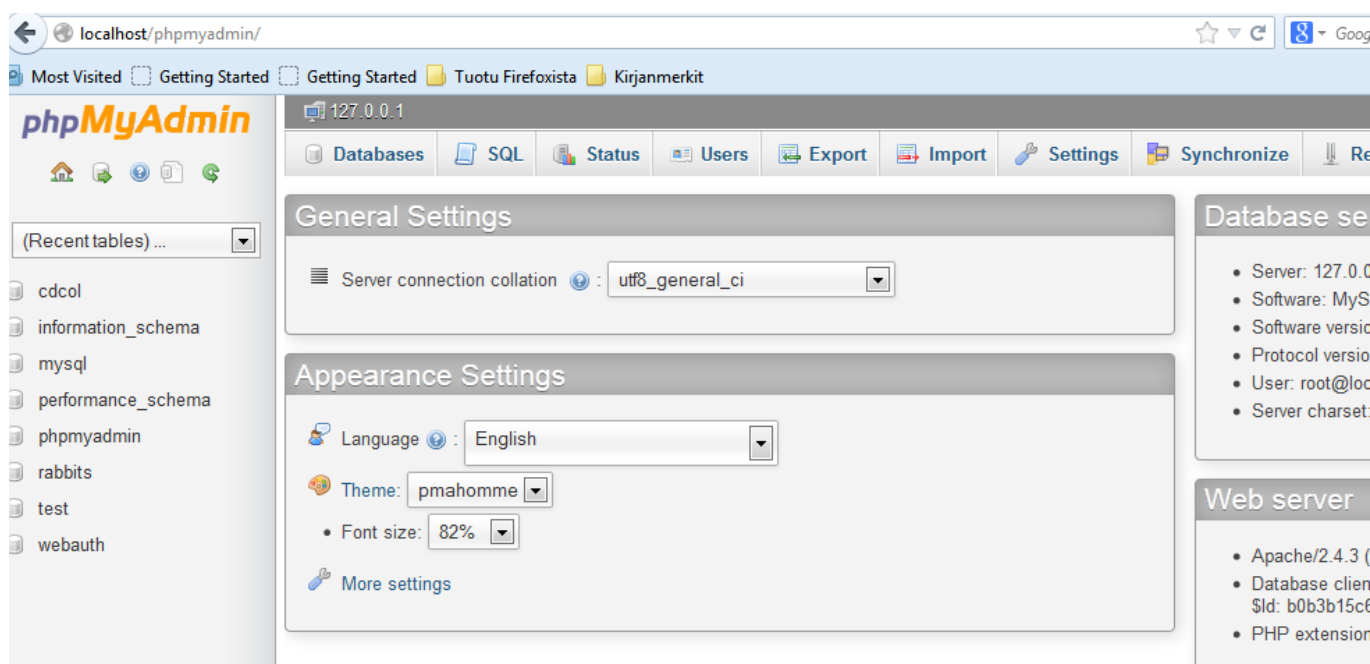
Screenshot 4: Copying the Drupal folder

### 4.3 Creating the database with phpMyAdmin

**Step 1:** Databases can be managed from phpMyAdmin main control panel. The main site of the control panel can be seen from screenshot 5. In case the control panel has wrong language setting, this can be changed in the Appearance Settings. The interface looks the same in a different language.

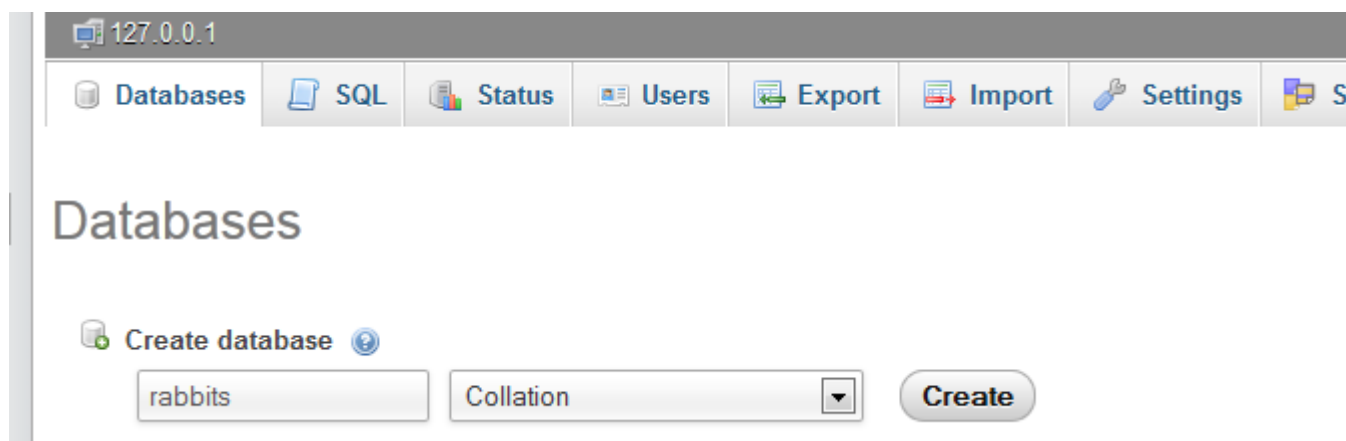
- phpMyAdmin can be accessed from the address <http://localhost/phpMyAdmin/>. This can be written on the address field on a internet browser.

If the control panel cannot be found or the site will not open, it should be ensured that XAMPP is running. For phpMyAdmin to function, Apache and MySQL are required to be running.



Screenshot 5: PhpMyAdmin control panel

**Step 2:** databases can be accessed from the main link bar link "Databases". The Database for the sites should be created. Notice that this name is needed later in Drupal settings. In screenshot 6 the database is named "rabbits". Database type should be set on "Collation".



Screenshot 6: phpMyAdmin creating databases

**Step 3:** After creating the database, access the "rabbits" database. The list of databases should be on the left side of the control panel.

**Step 4:** Click "Import" from the main link bar.

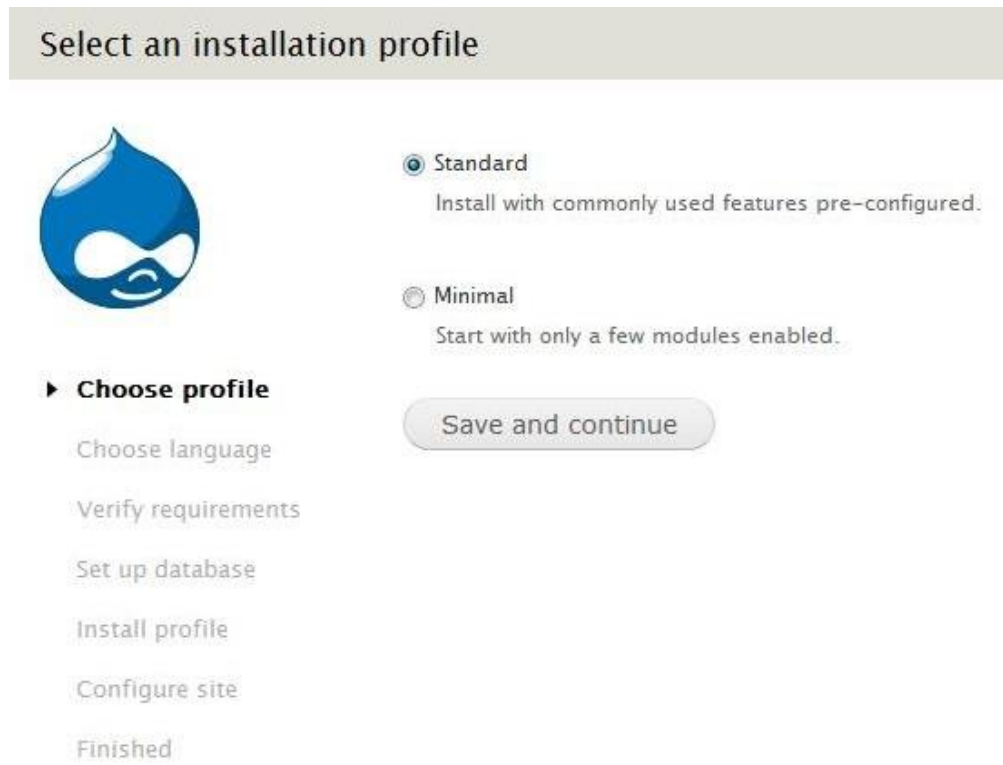
**Step 5:** Browse your computer to find the database. The latest version should be named in the same manner as: CityRabbits-2012-06-08T14-58-21.mysql (Notice that the name contains the date database was created. Name stated above is only an example.)

**Step 6:** By clicking "Go" the phpMyAdmin imports the database. There should be a notification "Import has been successfully finished."

**Step 7:** phpMyAdmin can now be closed after the importing has finished. Installing and configuring Drupal is covered in the next chapter.

#### 4.4 Drupal site configuration and confirming the database

Drupal installation is not required if htdocs folder and database are available. Htdocs folder refers the "rabbits" folder under C:\xampp\htdocs\ in this folder there is all information concerning the website. Database creation is done within phpMyAdmin and this is described at chapter 4.3.



Screenshot 7: Drupal configuration 1

**Step 1:** First the internet-browser should be opened. The browser used in this project is Mozilla Firefox.

- Drupal can be configured from the address <http://localhost/rabbits/install.php> This can be written on the address field in the internet browser.

**Step 2:** Choose a profile (screenshot 7). The option "minimal" should be chosen only if there are clear plans for which kind of modules are needed for the site.

**Step 3:** Choose language. English is a good option as much of the Drupal content in Internet is found in English. Also this option doesn't require any additional downloads. From the link "Learn how to install Drupal in other languages" can be found instructions for the other languages.

**Step 4:** In database creation there must be given the same name of the main folder that was renamed earlier. As can be seen from the picture "The name of the database your Drupal data will be stored in. It must exist on your server before Drupal can be installed.". Notice: If error message appears as seen in screenshot 8, re-installation is required.





In order for Drupal to work, and to continue with the installation process, you must resolve all issues reported below. For more help with configuring your database server, see the [installation handbook](#). If you are unsure what any of this means you should probably contact your hosting provider.

Failed to connect to your database server. The server reports the following message: *SQLSTATE[HY000] [1045] Access denied for user 'root'@'localhost' (using password: YES)*.

- Is the database server running?
- Does the database exist, and have you entered the correct database name?
- Have you entered the correct username and password?
- Have you entered the correct database hostname?

**Database type \***

- MySQL, MariaDB, or equivalent  
 SQLite

The type of database your Drupal data will be stored in.

Screenshot 8: Drupal error

**Database type \***

- MySQL, MariaDB, or equivalent
- SQLite

The type of database your Drupal data will be stored in.

**Database name \***

rabbits

The name of the database your Drupal data will be stored in. It must exist on your server before Drupal can be installed.

**Database username \***

root

**Database password**

••••••••

▼ **ADVANCED OPTIONS**

These options are only necessary for some sites. If you're not sure what you should enter here, leave the default settings or check with your hosting provider.

**Database host \***

localhost

If your database is located on a different server, change this.

**Database port**

If your database server is listening to a non-standard port, enter its number.

**Table prefix**

If more than one application will be sharing this database, enter a table prefix such as *drupal\_* for your Drupal site here.

Save and continue

Screenshot 9: Drupal configuration 2

**Step 5:** The database username and the password is the same that are used to log in Drupal when they require editing. These “admin” rights are only in use when logged in. The screen in Drupal configuration to give information can be seen from screenshot 9.

**Step 6:** Advanced options can be ignored if the files for the sites are located on the computer. This section is for those cases when the main database files are retrieved from some other source than the computer hardware or local network. If the files are transferred through the Internet, the address is added here.

**Step 7:** After the installation, it is time to add files to the rabbits folder on C:\xampp\htdocs\rabbits. This is done at chapter 4.2 Drupal configuration

#### 4.5 Addresses and other functions

- Drupal sites can be accessed from localhost- address <http://localhost/rabbits/>
- Admin sign -in can be accessed from localhost- address <http://localhost/rabbits/?q=user>

The address where Drupal sites are found is on localhost with the name of the database on it. In the thesis project it is <http://localhost/rabbits/>. This can be written on the address field on internet browser. The address takes the user to the front page of the sites and at the start of the project that is the only site there is. To get on the log in screen for Admin rights, this can be found from address <http://localhost/rabbits/?q=user>.

In below address, Administrator can login in the web server. More about cPanel web server can be found in the chapter 8.

SENSORED ADDRESS

#### 5 Backupping

It can never be over emphasized how important it is to take backups of the work. In case the new module confuses all interface or there is data corruption or breakups during the process, there is always the possibility to return to the last saved back-up file. Taking backups from the database is the most critical. Database folders hold all interface, functioning and cosmetic data of the sites. Before the working with website can begin, the first back-up should be taken from the database within phpMyAdmin. This is to ensure back -up function to run properly.

##### 5.1 Taking a backup from the Database (mySql)

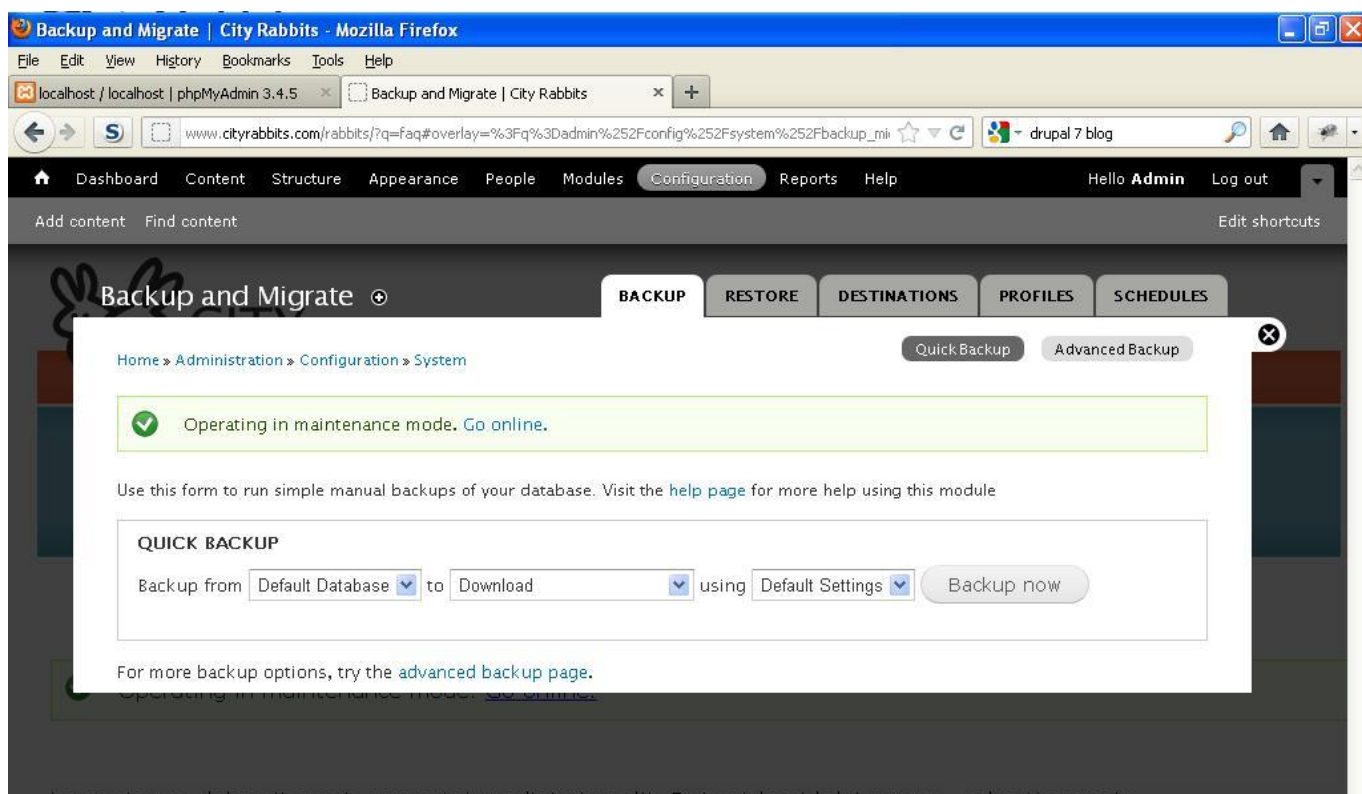
**Step 1:** Choose "Configuration" from the main link menu.

**Step 2:** Under SYSTEM, find link "Backup and migrate".

**Step 3:** Choose the backup from: Default Database (screenshot 10).

**Step 4:** Choose the location where to save the backup. The Manual Backups Directory option will save the backup at C:\xampp\backupit\backup\_migrate\_migrate\manual

"Download" option lets you choose your own destination, for example, the desktop of the computer.



Screenshot 10: Backup and migrate module 1

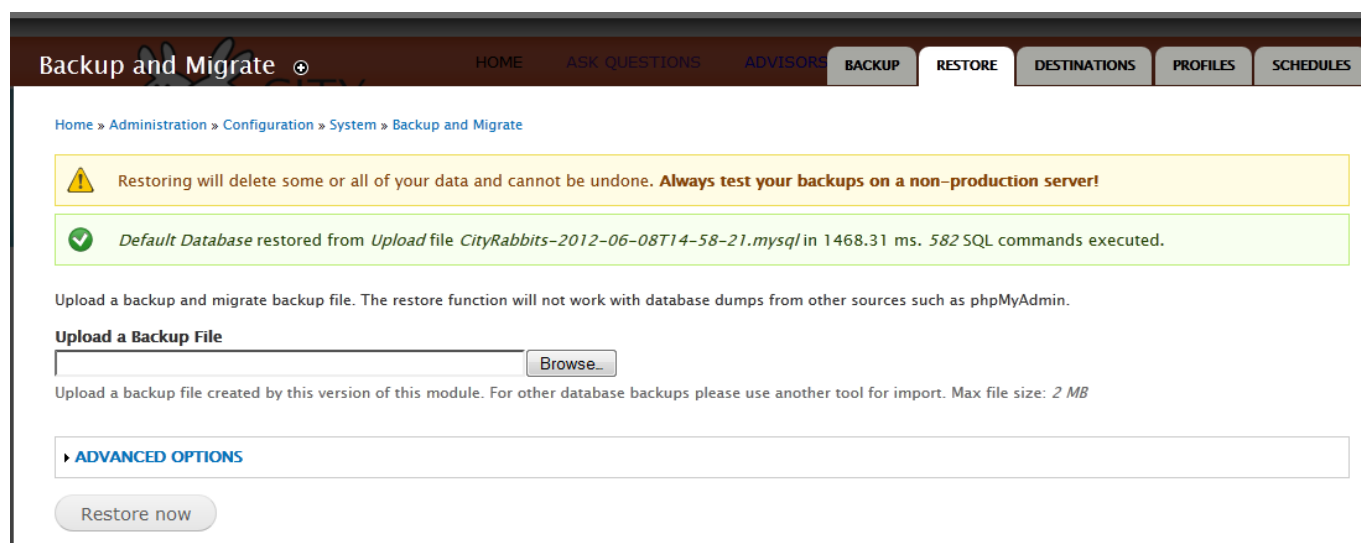
## 5.2 Restoring the database

The database can be restored within Drupal and this is recommended when building the sites with Drupal. The backup files can be saved on computer drives and external memory storages. External memory storages should be ensured that the contained data will not break or corrupt during the transportation.

**Step 1:** Choose "Configuration" from top at Drupal administration link bar.

**Step 2:** Under the link "System", there should be section for "Backup and migrate."

**Step 3:** Change the tab "Backup" to "Restore". Different tabs can be seen from the right top corner. Restore tab can be seen at screenshot 11.



Screenshot 11: Backup and migrate module 2

**Step 4:** From this tab click "Browse..." and find the backup database of the sites (as done at previous chapter).

**Step 5:** Click Restore. Database has been restored.

### 5.3 Drupal

When editing site on localhost, it is wise also to take backups from Drupal. The “rabbits” folder found from: c:\xampp\htdocs should be packed as a zip -file before moving. This zip -file can be saved on any desired destination.

**Step 1:** Right -click ”rabbits” folder in C drive.

**Step 2:** Choose option ”7-zip” (or an alternative zip-program) and click ”Add to packet ”rabbits.zip””.

**Step 3:** Rabbits.zip -file is created in the same folder as the original rabbits folder.

### 6 Modules

To make Drupal page more appealing and have more functions, there are numerous modules that can be downloaded and used for the sites. Using modules is highly recommended as the main functions in Drupal core have limited chances to modify what the sites will look like. Modules can be searched and downloaded from the drupal.org at the Download & Extend section. Also on Drupal forums (found at Community section) there are many users discussing and giving advice for good modules.

- Most trusted modules can be found at drupal.org/download. Google search is also viable tool for finding proper modules.

All modules are programmed at first for a certain version of Drupal (5,6,7 etc). In time, they are updated to the latest versions, but when downloading new module, it has to be made sure that it is meant for the same version as Drupal downloaded at the computer. The community gives feedback and ranking to the modules that can be found at drupal.org. A good rule to find good and functioning module is to ensure that there are recommendations and a great number of downloads on the module information. As there are great numbers of different coders and ideas how Drupal should work. This leads to the fact that not all modules in the drupal.org database are good/functioning/worth trying.

## 6.1 Modules used in Drupal

When managing modules, there are usually some modules that require another module to stay functioning. Nevertheless all those modules that are not in use should be removed. This should be done carefully as there is a chance at deleting something that causes the site or another module to crash.

**Panels** module divides the web page into panels or paragraphs better than Drupal with default settings. The baseline is to improve the layout and look of the page. With Panels module it is possible to choose the size and amount of the paragraphs for the new page. The pages in City Rabbits web sites are divided in 2 paragraphs with the ratio of 70% to 30%. All the newly created pages should be classified in the miscellaneous section to find the option for "rabbits style". This style should be applied for all pages to keep the layout united whole.

**Captcha** module is used at Contact us -page. This module is meant to prevent spam messages at the page where feedback messages are sent for the site owners. Captcha creates a collection of twisted letters and numbers beside the "send"-button. Before the message can be sent, user must insert correct letters in the given field.

**Facebook comments** module is the current solution to arrange chatting in the sites without requirement to log in to the service. In the chat site there is space similar to Facebook wall feed where users can post new messages. The posting requires existing account in Facebook, but this way City Rabbits does not need its own log in. From the module settings the option for "post to Facebook" is unchecked preventing the posts in sites to be posted in City Rabbits Facebook page.

**Facebook like button** is simple module that creates small "Like"-button at the City Rabbits front page. Through this link the user can like and follow City Rabbits news in Facebook.

**Chaos tools suite (CTools)** is module for managing modules and it is designed for administrators working with Drupal. This module does not affect in the site layout or users, but it makes the module control panel easier to manage by grouping downloaded modules in their own categories. Usually one module have several functioning parts and CTools keeps these parts together.

**Backup and Migrate** module is also for administrators working with Drupal. With this module it is possible to take backup-files from the sites and continue developing later or in a new location. Taking backup-files of the sites is fully described at the chapter 5.

**Views** module can be used to manage content in the sites to be hidden or visible for different users. For example, if there was several different user profiles in the sites, with Views parts of the content could be accessible for selected users. At the current sites there is only account for the administrator. Normal users are all equal and can access all public sites.

## 6.2 Installing modules

- All modules are downloaded as .zip files and should be unpacked and moved to the folder:  
C:\xampp\htdocs\rabbits\modules

After download, the computer needs a program to unpack the .zip files. For example Windows 7 has this function without additional downloads. After this in Drupal, the modules-page should be reopened and the downloaded modules should be found on the list. These still has to be turned on, by checking on the module name and scrolling down the page to turn them on. After this it is possible to start adjusting its settings as desired.

## 7 Content creation and editing

Content can be created and edited through cPanel and website. Structure changes such as creating new pages and editing content are done through the website when administrator is online. Trough cPanel it is possible to update database and all new pictures are brought to cPanel server before adding them to the sites.

At cPanel there is a login required before working with the City Rabbits site content. cPanel City Rabbits login can be found from the address below.

SENSORED

Username: SENSORED  
Password: SENSORED

Notice that this information is to be censored if the guide is presented to the other parties involved.

Content creation and editing is done through the website when the administrator is logged in to the sites. The link to the log in sites cannot be found from the sites that are meant to the normal users. The address to the log in is only presented below.

SENSORED

Username: SENSORED  
Password: SENSORED

Pictures for the sites are first downloaded at cPanel before they are used on the sites. Notice that these pictures should be formatted to the right size and form as there are very little chance to edit them after download. The address for the pictures is presented below.

SENSORED

## 7.1 Create a new page

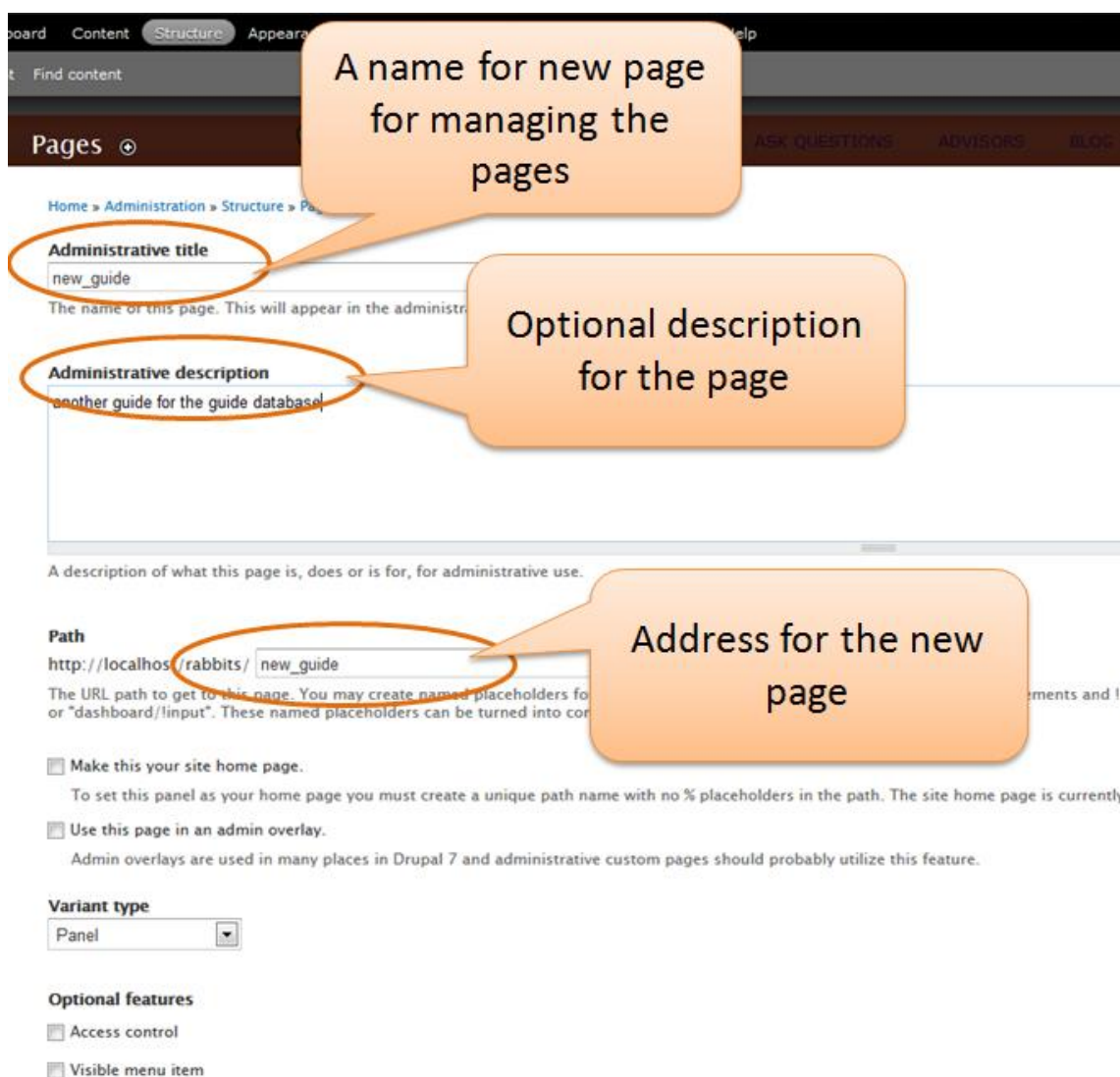
When there is a need for new pages in the websites such as new advisor profiles or other info pages, they can be created when Administrator is logged on. New pages are created in Drupal from Structure > Pages list. There is link to create new page and the first page is as shown in the screenshot 12.

**Step 1:** Give title, description and address. Then click "continue".

Notice that the title of the page is the one shown when managing the pages overall in administrator mode. On description field can be briefly described what the page contains (do not just copy page text content here). When viewing the list of existing pages on the sites, description is shown beside the title. Title and description will not show to normal users.

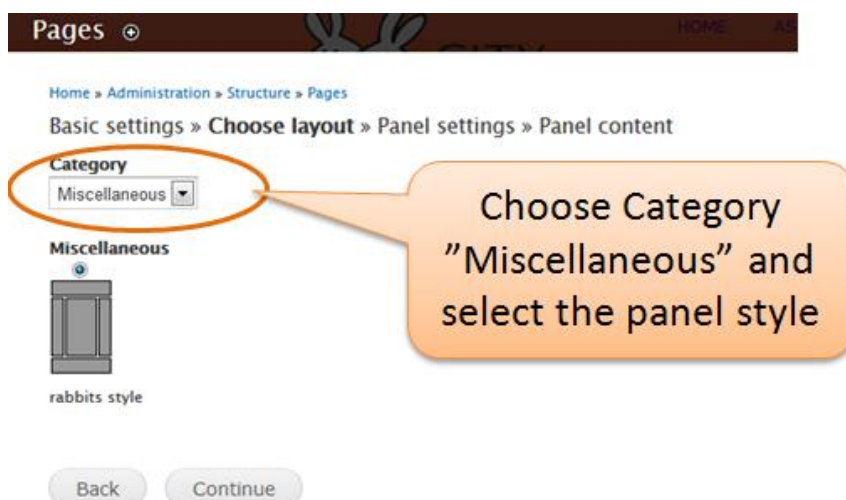
When creating address for the new page, notice that this will be shown to the normal users on the address field. The start of the address is presented and the part concerning the new page can be chosen. The name should describe the new site such as /info, /contact\_us or /advisor\_harry\_potter.

- Notice that with Finnish advisor names, letters such as Ä, Ö and Å must be changed into A and O in the address. For example, /advisor\_maija\_meikalainen



Screenshot 12: Create new page (title, description, address)

**Step 2:** Choose panel style for the page. Miscellaneous and "rabbits style". Continue.

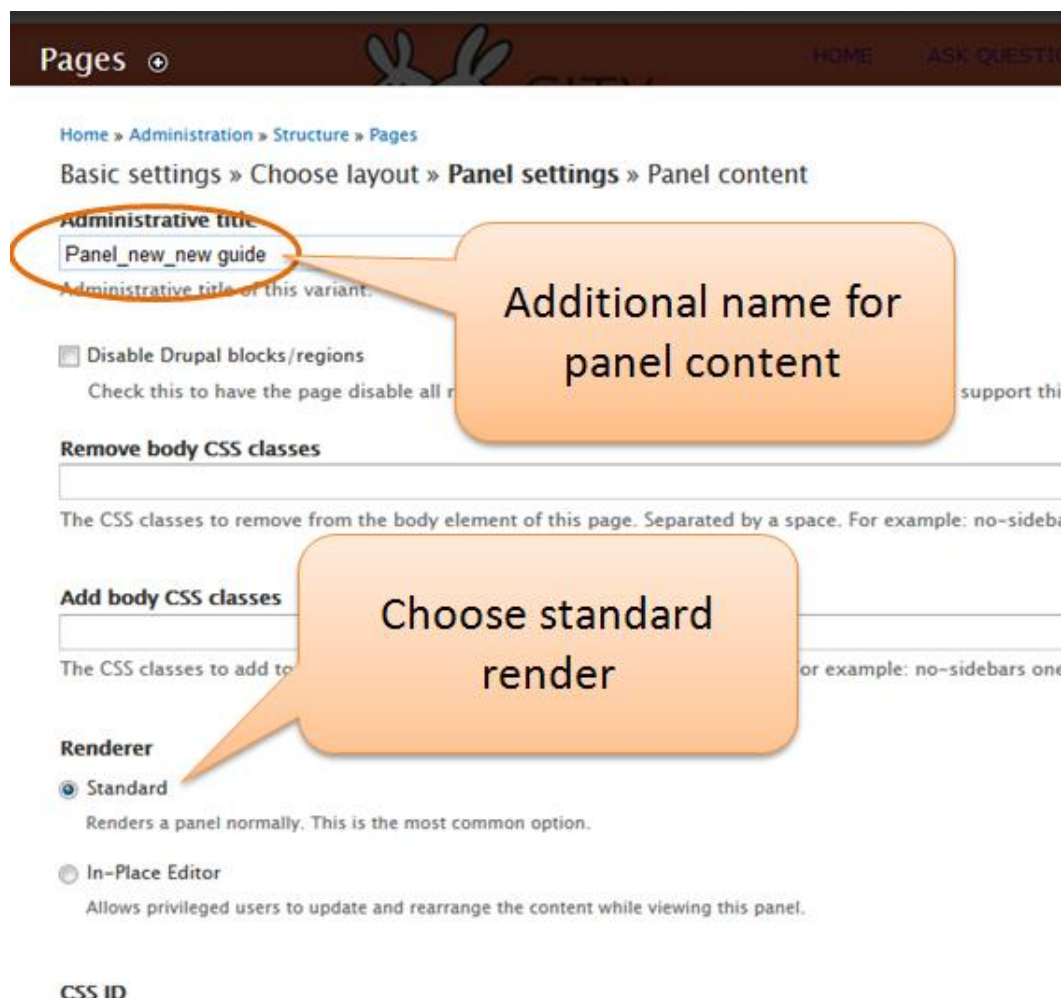


Screenshot 13: Create new page (choose panel style)

**Step 3:** Panel content name, choose standard render. Click continue. New page is now created with general appearance of the normal page with the header and the footer, still without content. Content can be created lat-



er, but before that the page should be found from somewhere ie. there should be link somewhere to lead to the new page. This can be done by editing an existing page and adding the link in their main content. Such links can be "Enter the gallery here" or "Site Advisors".



Screenshot 14: Create new page (panel content)

## 7.2 Editing an existing page

To edit existing pages such as Advisor -profiles and basic content sites is done with administrator rights from any computer. When the administrator is logged in, every page has a black bar at the bottom of the sites with links "Customize this page" and "Change layout" as can be seen from screenshot 15. To edit the text content, "Customize this page" link should be chosen.

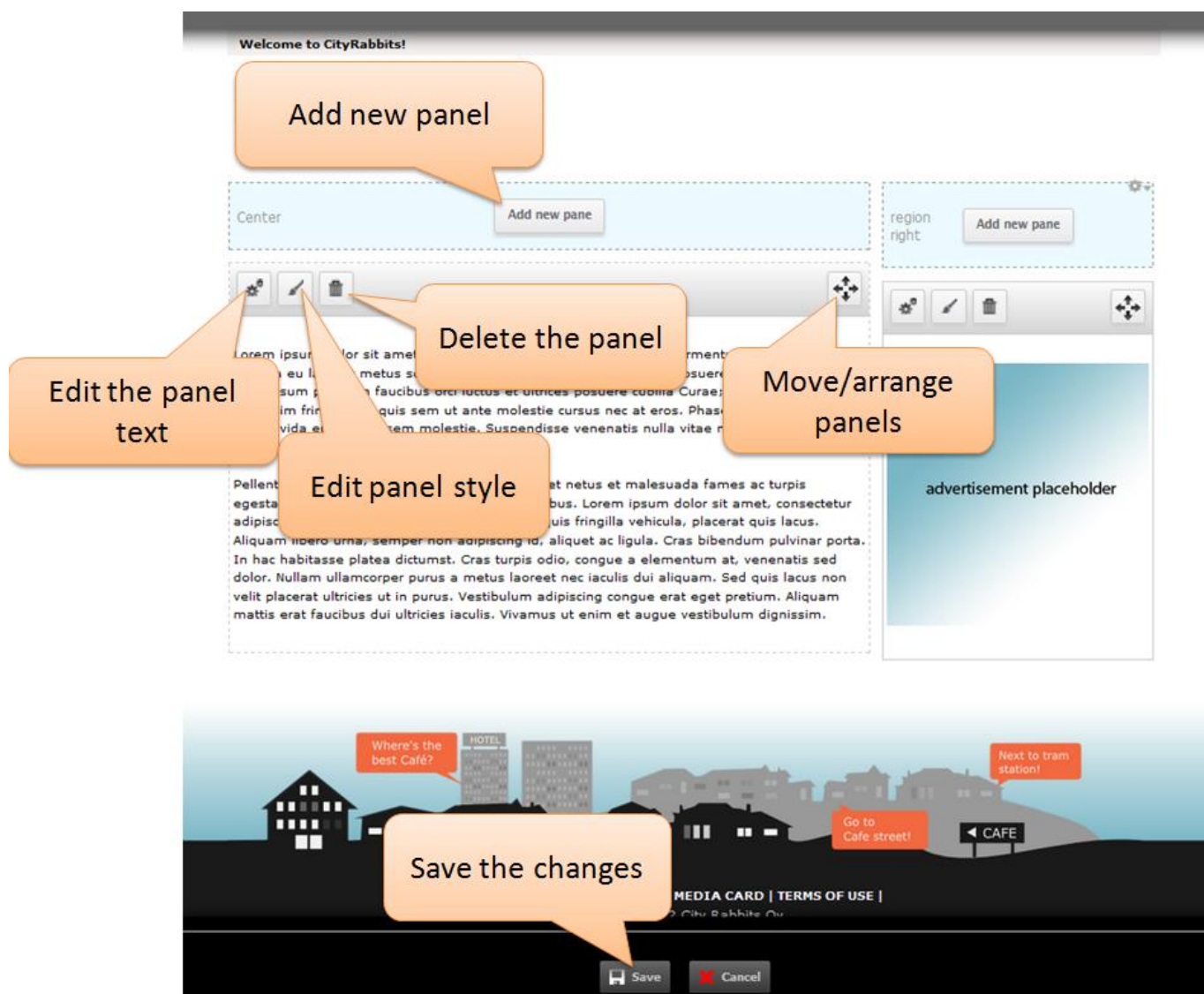
From the link "Change layout" can be changed the panel structure of the page. As all City Rabbits pages are set on special rabbits style of 70%:30% panels, it is not advised to change this.



Screenshot 15: Edit site content

From the customization mode the page content can be changed. Every page in this mode has the same edit options. As can be seen from screenshot 16, the panels can be edited in the following methods:

- Add new panel -button at the top of the panel adds another panel at the top of previous panel. This can be used, if there is a specific need to separate content on the panel in more clear way than "hitting several enters" on the content.
- Edit the panel text -button can be accessed the main text edit field to enter desired text into the current panel. Notice that when the page is set on "Full HTML -mode" this text field executes also scripts of HTML -code.
- Edit the panel style -button allows to change font and panel background colours. The colours are entered on their own fields as colour codes. Notice that these settings should be left on their default values if there is not any specific need to change the normal look.
- From delete the panel -button simply deletes the single panel.
- From move/arrange -button the panel order can be adjusted. The panel moves simply just clicking the button and dragging with the mouse.



Screenshot 16: Edit the site content, customization mode

All the changes become available to normal users after saving the changes. Also notice that by clicking "Cancel" after editing, all the changes done in that session are lost.

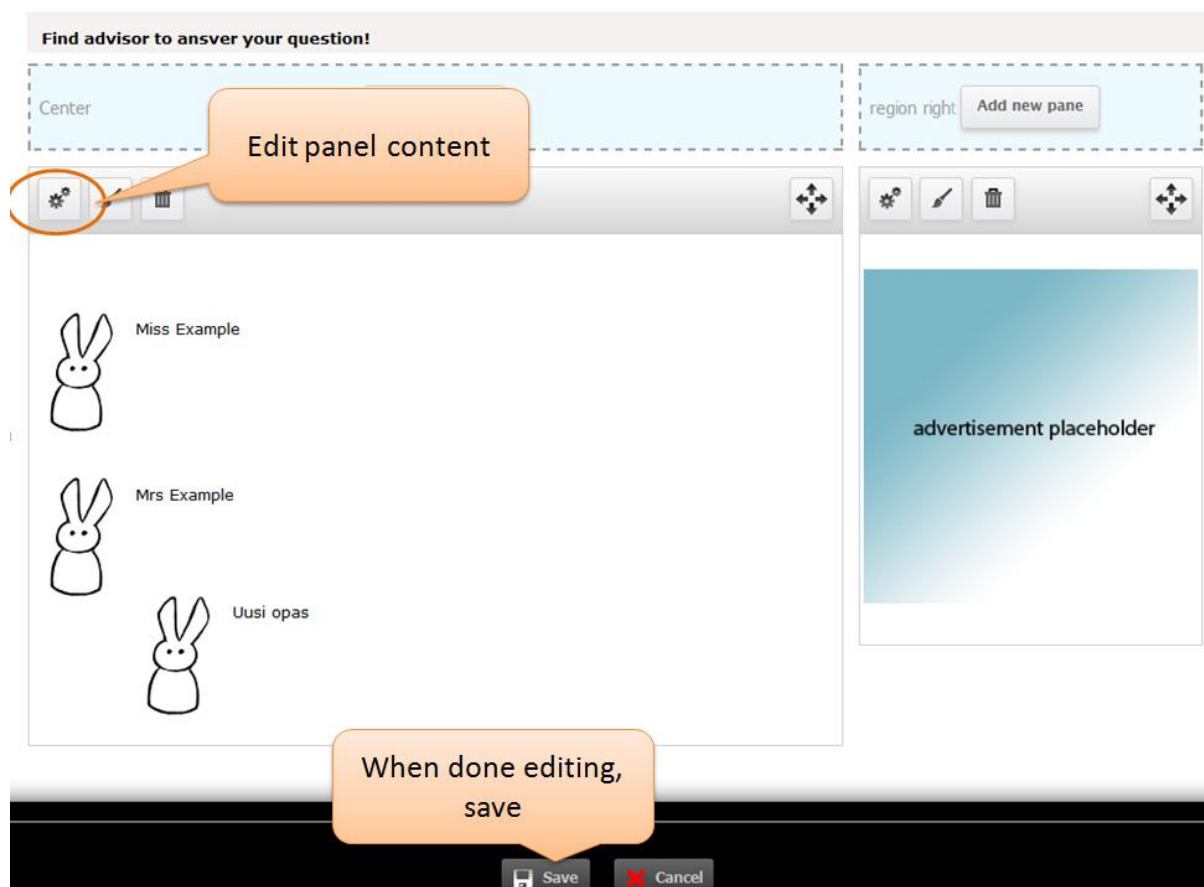
Advertisement placeholder at right region panel is an image that is brought to the panel by HTML code. Changing the code is covered on the next chapter 7.3.

### 7.3 Linking the new sub -page to the existing pages, editing HTML code

Currently there is not much sub page content in the sites. In case the need for them becomes relevant, subpages should be linked to the main structure i.e. Main page should contain a link to the subpage. Otherwise the new subpage technically exists, it has an address, but the user does not have "a way leading" to this page. This leads to that the user does not know that the page even exists.

- In here is described how to link new advisor profiles to the profiles main page. Also this contains instructions to edit HTML code in any panel.

To make link from Advisors main page, the user should be first on that page. When admin is signed in, there is black bar of 2 links in the bottom. "Customize this page". You get view as shown in the screenshot 17.



Screenshot 17: Edit panel content

- Actual changes to the site appearance is done through HTML-code that is embedded in the paragraph body. This method can be used in all paragraphs on the sites. Also the "advertisement placeholder" image can be replaced with this.
- Good way not to cause any damage to the original code is to copy it for example in Notepad and keep this as a back-up file. In this way even if new changes do not work within the new HTML code, there is always chance to return to the original point. As it is real computer code edited in this part, certain carefulness should be taken with punctuation marks as leaving them out will result as a broken code.
- The changes require saving in the paragraph customizing page.

To bring new advisor to the Advisors main page, the first advisor link-code can be copied to the list and change the advisor profile address, advisor image and advisor name as required. Also deleting the code concerning the advisor link and picture, it is possible to remove older advisors.

Similar to the following HTML-code can be found from Advisors main page on center paragraph. This part of the HTML code contains one advisor image, advisor profile link and advisor name. Certain key parts are highlighted for better emphasis.

```
<a href="http://localhost/rabbits/miss_example">

</a>&nbsp;&nbsp;&nbsp;Miss Example
<br/>
<br/>
```

<br/> tag refers to the text continuing from the next line after the tag. This has the same effect as pressing "enter" on a normal text editor. Writing code with tags on their own line only makes the code more easy to read. Take a notice that there is always two "enters" after the advisor name.

`<a href="http://localhost/rabbits/miss_example">` refers to the address the advisor link will lead to. By changing this, the link on the picture will take the user to the address which is given here. The full address is highlighted in light blue on the example.

As the link itself is the picture of the advisor, the sites has to be told what kind of image it is. And this image source should be coming from cPanel picture index (see chapter 8.1 Adding pictures to cPanel). With the command `` the site shows picture of the advisor. The yellow highlight refers to the address of the picture in cPanel picture index. After the apostrophe the width and height of the image is defined. To keep the site symmetrical, it is advised to keep all the images on same size. In this case 80 x 50 pixels. The align-command groups the images on the left side of the paragraph.

`</a>&nbsp;Miss Example` -command creates the name of the advisor next to the picture. All the marks written after semicolon will appear as they are in the code. The name of the advisor is highlighted here with green color.

The Advertisement placeholder -image can also be changed in the same method. Currently the text body field contains HTML -code

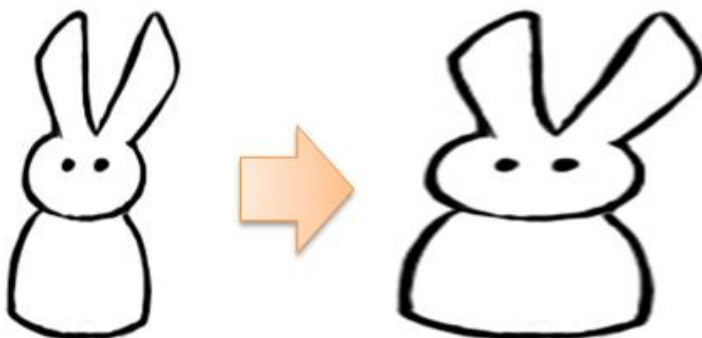
```

```

which refers to the image and its width and height. If this image were needed to change into a link to other page (for example, to the advertisers home page) the HTML code should be added with the link source part making the link as follow

```
<a href="enter the link here">  
  
<br/>
```

The light blue highlights the place for advisor site address and the yellow highlights the image address from cPanel. Notice that the width and height numbers are size of the image in pixels. This is definite setting for the image dimensions and should always be set on the real size of the image. If the dimensions are set wrong, the resulting image would look stretched like in the screenshot 18.



Screenshot 18: Wrong image dimensions in HTML code

The correct image width and height can be found from image details in cPanel. Uploading images to cPanel, finding image information and precise image addresses are instructed at the chapter 8.1.



After you have created content, you can find pages from Content - Find content. Pages are listed on this page (screenshot 19). You can edit or delete pages through this page.

The screenshot shows the Drupal administration interface for content management. At the top, there is a navigation menu with options like Dashboard, Content, Structure, Appearance, People, Modules, Configuration, Reports, and Help. The 'Content' tab is active. Below the navigation, there is a search bar with 'Find content' and a 'CONTENT' button. The main content area has a breadcrumb 'Home > Administration' and a '+ Add content' link. There are two filter sections: 'SHOW ONLY ITEMS WHERE' with dropdowns for 'status' and 'type' (both set to 'any') and a 'Filter' button; and 'UPDATE OPTIONS' with a dropdown for 'Publish selected content' and an 'Update' button. Below these is a table listing content items.

<input type="checkbox"/>	TITLE	TYPE	AUTHOR	STATUS	UPDATED	OPERATIONS
<input type="checkbox"/>	Front page <span style="color: red;">updated</span>	Basic page	Admin	published	04/20/2012 - 11:44	<a href="#">edit</a> <a href="#">delete</a>
<input type="checkbox"/>	Terms of use	Basic page	Admin	published	04/18/2012 - 12:20	<a href="#">edit</a> <a href="#">delete</a>
<input type="checkbox"/>	Media card	Basic page	Admin	published	04/18/2012 - 12:20	<a href="#">edit</a> <a href="#">delete</a>
<input type="checkbox"/>	Contact us	Basic page	Admin	published	04/18/2012 - 12:20	<a href="#">edit</a> <a href="#">delete</a>
<input type="checkbox"/>	About us	Basic page	Admin	published	04/18/2012 - 12:06	<a href="#">edit</a> <a href="#">delete</a>
<input type="checkbox"/>	Blog	Blog entry	Admin	published	04/18/2012 - 10:42	<a href="#">edit</a> <a href="#">delete</a>
<input type="checkbox"/>	Ask questions	Basic page	Admin	published	04/18/2012 - 10:41	<a href="#">edit</a> <a href="#">delete</a>
<input type="checkbox"/>	Advisors	Basic page	Admin	published	04/18/2012 - 10:41	<a href="#">edit</a> <a href="#">delete</a>
<input type="checkbox"/>	FAQ	Basic page	Admin	published	03/30/2012 - 10:17	<a href="#">edit</a> <a href="#">delete</a>

Screenshot 19: Content list

## 7.5 Creating new blog entry

New blog posts can be done from any computer when the administrator have logged in without working with Drupal. Basically the new created page is which is categorised as a blog page. All older blog posts are listed by post-date on blog main page. To read the full blog post, user can follow the blog post link to access whole post.

**Step 1:** Click content (top of screen).

**Step 2:** Click Add content.

**Step 3:** Choose Blog entry as a content type.

**Step 4:** Give your blog entry a title. This title is shown at the blog post list at the blog main page.

**Step 5:** Write your blog entry on the content field. (Or copy the text from a text editor)

**Step 6:** Choose Text format: Full HTML.

**Step 7:** Enable Facebook comments if needed.

**Step 8:** Check Publishing options. "Published" option should be checked.

**Step 9:** By clicking "save", new blog entry has been created.

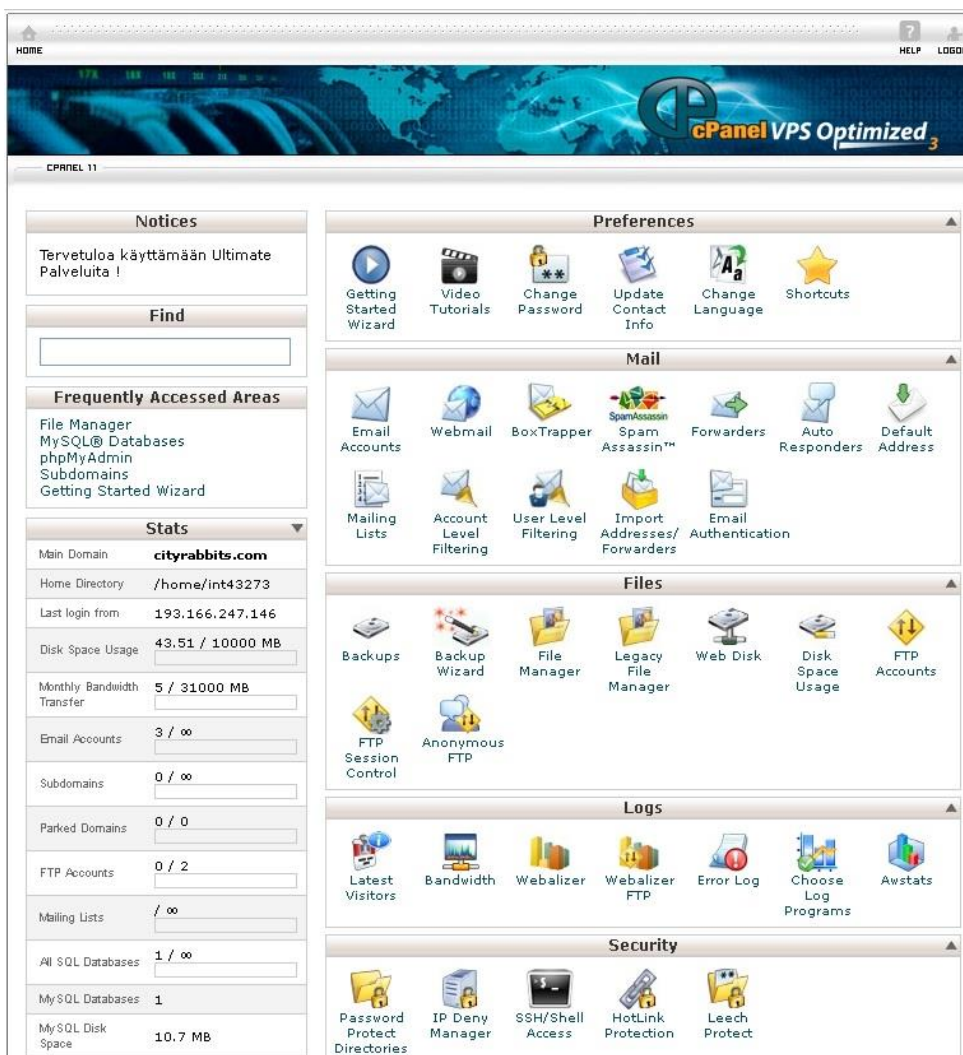
## 8 Web Hosting Management: Cpanel

cPanel is a Web hosting management system. All needed data to run the website, is stored in there; Drupal system files, database and pictures. Drupal is transferred to a web server, for Administrator updating and adding content directly for website. Notice that cPanel is used when the whole site content needs to be updated from localhost to the server. Such cases could be, for example, updating the look of the sites or bringing great structure changes online.

When uploading any images to the sites, they have to be uploaded in cPanel before adding image on a site. Uploading images to the server is instructed at chapter 7.1.

- cPanel can be accessed on the address SENSORED ADDRESS

Username: SENSORED  
Password: SENSORED



Screenshot 20: cPanel home page

MySQL Database is already created on server. Database backup is done in localhost.

## 8.1 Adding pictures to cPanel

- cPanel for the City Rabbits sites can be found from address SENSORED

**Step 1:** Log in to the cPanel (Log in information can be found at the start of chapter 8)

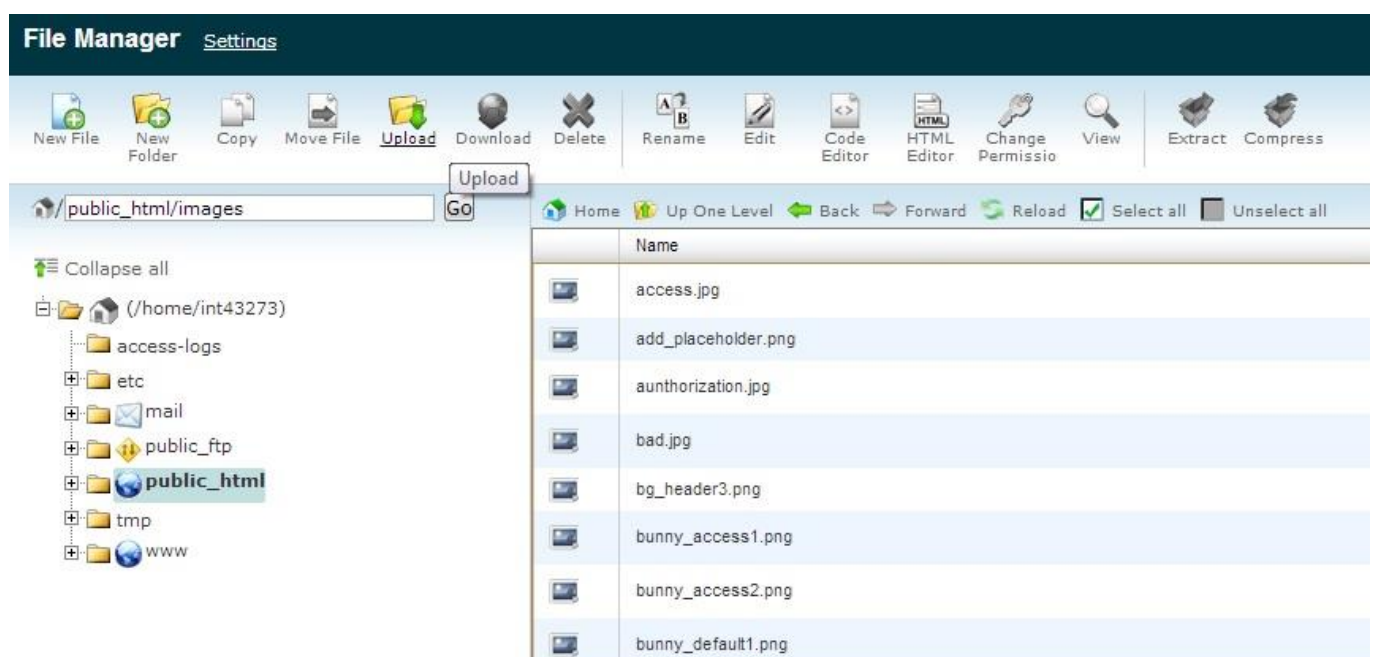
**Step 2:** Go to Files and choose File Manager. This will open a pop-up window. Select Web Root (public\_html/www). The pop-up window can be seen from screenshot 21.



Screenshot 21: Directory Selection

**Step 3:** Continue by clicking “Go”.

**Step 4:** Under Web root folder, is folder named ”images”. Double click it to open it. A list of all images on City Rabbits domain will appear.



Screenshot 22: File Manager

**Step 5:** New pictures can be loaded from "Upload" button at the top of the page (screenshot 22).

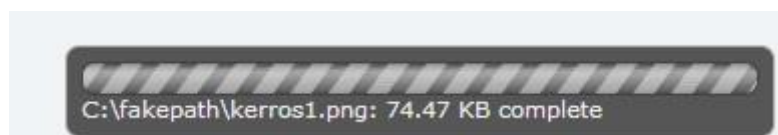


**Step 6:** New site will open as seen from the screenshot 23. Choose the files by clicking "choose files" ("Valitse tiedosto" in Finnish operating systems). Find the picture from computer for uploading and continue by clicking "save".



Screenshot 23: Choosing the picture for upload

**Step 7:** On the lower part of page appears notification, as seen from screenshot 24, when the upload is finished.



Screenshot 24: Picture upload ready

After the upload is complete, the user can choose to upload more pictures in the same way or return to the File Manager by clicking the address link "SENSORED". New uploaded pictures can be seen from the list. By right clicking the picture file, there are various options for the files. For example through this, it is possible to view or delete the picture.

To bring image as a part of the site content, the precise image address has to be used in the code. The following steps are one way to find out the address of a single image. How to add the image to the sites is described at the chapter 7.3.

**Step 1:** From cPanel front page, choose "Legacy File Manager".

**Step 2:** Choose the link Web Root (public\_html/www).

**Step 3:** Click "Go".

**Step 4:** Go to "images" folder by clicking small folder picture next to folder name.

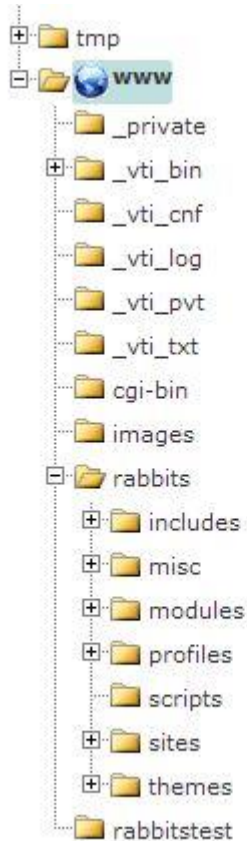
**Step 5:** Click the name of the image.

**Step 6:** The address of the image appears to right of the screen, right above icon "Empty Trash".

**Step 7:** By clicking the URL address, image can be viewed.

## 8.2 Uploading the Drupal files to the server

Uploading the Drupal files to the cPanel server replaces all the existing content online. This includes all advisor profiles and blog posts if they are not included in the new uploaded version. In this method the files are brought from the localhost to the server. Localhost updates are mainly needed for appearance updates and other changes that require testing before uploading online. Notice that the sites can be turned into maintenance mode during the "online testing". The maintenance mode is instructed at the chapter 9.3.



Screenshot 25: Rabbits -folder

**Step 1:** To upload Drupal files from localhost to the server, from home page (screenshot 20) under Files choose File Manager.

**Step 2:** Choose Home Directory.

**Step 3:** On the left side of the screen (screenshot 25) find folder named as www.

**Step 4:** Create folder where to copy Drupal files. Check that "www" folder is chosen and click new folder icon from top of the screen.

**Step 5:** Give the folder a name (rabbits).

**Step 6:** Now you can upload your Drupal files to the created folder. Check that the folder is chosen and click Upload icon from the top of screen.

**Step 7:** Upload Drupal files as seen pictures at chapter 8.1. Drupal files must be on one zip-file.

**Step 8:** Now Drupal sites can be found at the address: [www.cityrabbits.com/rabbits](http://www.cityrabbits.com/rabbits)

## 9 Updating

Updates for the website are required in the future in case of the further website developing. Depending of the newest Drupal version, the majority of available modules are designed to function with it. XAMPP also has its maintenance time to time and there will be newer versions of it. Updating XAMPP does not affect in the functioning of Drupal or modules. Information about updating Drupal 7 to newer versions will be available after new version release.

### 9.1 Updating Drupal versions

Drupal has different versions as many other programs. During the thesis project, Drupal 6 was still in use in many cases, but Drupal 7 was already out. Also there was the next version, Drupal 8, coming up. Updating Drupal versions did never take place when building the City Rabbits- website.

From NTAP Scott Friday's Blog and also from Drupal.org can be found step by step instructions how Drupal can be updated from 6 to 7. As it is technically possible, it is highly recommended to make backup files of the old work before starting to update. As Scott Friday's Blog is stating, updating the whole program is not easy and there are many problems concerning modules and their functioning in the next version. It is very likely that similar guides can be found from Drupal forums after Drupal 8 is released.

### 9.2 Updating modules

Every time there is new module downloaded, it has to be taken care that the right version for Drupal 7 is chosen. Modules for older version of Drupal will not function in the new version. This causes continuity problems for later work if there is a need for updating the main program.

In the case new versions cannot be found for modules after Drupal update, it is very likely that the feature has to be built again in an alternative manner. This will cause more work for further developing. Existing modules do not require updating even if the module receives newer versions.

### 9.3 The maintenance mode

Maintenance mode takes the site offline for maintenance or brings it back online. This option is situations where there is a need for denying user access to the web sites. Maintenance mode shows only a front page notification that the sites are under maintenance.

**Step 1:** Click Configuration on top of the screen.

**Step 2:** Find Development section.

**Step 3:** Choose Maintenance mode.

**Step 4:** Either take the site offline for maintenance or bring it back online (screenshot 26).

The screenshot shows the 'Maintenance mode' configuration page in Drupal. At the top, there is a dark navigation bar with the title 'Maintenance mode' and several menu items: HOME, ASK QUESTIONS, ADVISORS, BLOG, and FAQ. Below the navigation bar, a breadcrumb trail reads 'Home » Administration » Configuration » Development'. The main content area contains a paragraph explaining that users may need to run an 'update script' when upgrading. A checkbox labeled 'Put site into maintenance mode' is checked. Below this, a text box explains that only users with the 'Use the site in maintenance mode' permission can access the site, while others see a maintenance message. A text area labeled 'Maintenance mode message' contains the text: 'City Rabbits is currently under maintenance. We should be back shortly. Thank you for your patience.' Below the text area, a label reads 'Message to show visitors when the site is in maintenance mode.' At the bottom of the page, there is a 'Save configuration' button.

Screenshot 26: Maintenance mode

## References

ApacheFriends homepage, [Apachefriends.org](http://www.apachefriends.org), XAMPP download.  
<http://www.apachefriends.org/en/xampp.html>  
Cited 23.9.2012.

Drupal.org, Drupalin kotisivut. Drupal download .<http://drupal.org/project/drupal>,  
viitattu 23.9.2012

Toggle.com. Free program downloads. DownloadWinRAR.  
<http://winrar.finnish.toggle.com/>, Cited 27.9.2012

Legal Services NTAP, National Technology Assistance project- homepage  
<http://lsntap.org/home> Scott friday's blog. Upgrading to Drupal 7. Cited  
14.11.2012  
<http://lsntap.org/content/upgrading-drupal-7>

Jouttunpää, E. 2012a. Verkostopalveluliiketoiminnan toteutussuunnitelma. Cited  
3.3.2013.  
<https://publications.theseus.fi/bitstream/handle/10024/40650/Elisa.Jouttunpaa.2012.pdf?sequence=1>

## Screenshots

Screenshot 1: XAMPP notification.....	43
Screenshot 2: Installation .....	44
Screenshot 3: XAMPP control panel .....	45
Screenshot 4: Copying the Drupal folder .....	46
Screenshot 5: PhpMyAdmin control panel .....	47
Screenshot 6: phpMyAdmin creating databases .....	47
Screenshot 7: Drupal configuration 1 .....	48
Screenshot 8: Drupal error .....	49
Screenshot 9: Drupal configuration 2 .....	50
Screenshot 10: Backup and migrate module 1.....	52
Screenshot 11: Backup and migrate module 2.....	52
Screenshot 12: Create new page (title, description, address) .....	56
Screenshot 13: Create new page (choose panel style).....	56
Screenshot 14: Create new page (panel content) .....	57
Screenshot 15: Edit site content .....	58
Screenshot 16: Edit the site content, customization mode .....	59
Screenshot 17: Edit panel content.....	60
Screenshot 18: Wrong image dimensions in HTML code .....	61
Screenshot 19: Content list .....	62
Screenshot 20: cPanel home page .....	63
Screenshot 21: Directory Selection .....	64
Screenshot 22: File Manager .....	64
Screenshot 23: Choosing the picture for upload .....	65
Screenshot 24: Picture upload ready .....	65
Screenshot 25: Rabbits -folder .....	66
Screenshot 26: Maintenance mode .....	68