

Timo Partanen

SATAKIRI-JULKAISUALUSTAN SUUN- NITTELU JA TOTEUTUS

Opinnäytetyö
Tietojenkäsittelyn koulutusohjelma

2017



**Kaakkois-Suomen
ammattikorkeakoulu**

Tekijä/Tekijät	Tutkinto	Aika
Timo Partanen	Tradenomi (AMK)	Joulukuu 2017
Opinnäytetyön nimi		31 sivua
Satakiri-julkaisualustan suunnittelu ja toteutus		
Toimeksiantaja		
Mikkelin kaupunginkirjasto		
Ohjaaja		
Arto Väätäinen		
Tiivistelmä		
<p>Opinnäytetyön aiheena on Satakiri-julkaisualustan suunnittelu ja toteutus. Opinnäytetyön toimeksiantajana toimi Mikkelin kaupunginkirjasto.</p> <p>Opinnäytetyön toimeksiantaja Mikkelin kaupunginkirjasto tarvitsi toimivan julkaisualustan satavuotiaan Suomen kunniaksi järjestettyä koululaisille suunnattua kilpailua varten. Kilpailussa koululaisia sekä toisen asteen opiskelijoita kehoitettiin lähettämään kaikenlaisia mediatuotoksia, mitkä heidän mielestään kuvastavat satavuotiasta maataamme. Tämä tarkoitti, että julkaisualustan täytyi pystyä vastaanottamaan videoita, ääntä, kuvia ja tekstiä sekä käsittelemään niitä järkevästi ylläpidon työkalujen avulla. Kilpailun lopuksi kaikki julkaisuluvan saaneet työt olisi myöskin tarkoitus näyttää julkisesti palvelun sivuilla. Työhön kuului myöskin sopivan webhotellin löytäminen sekä sivujen ja tietokantojen siirto palvelimelle.</p> <p>Opinnäytetyössä käydään läpi kaikki tärkeät tekniikat, ohjelmistot ja menetelmät, joita työprosessin aikana käytettiin. Nämä vaihtelevat cPanelista ja Bootstrapistä aivan YouTuben rajapintoihin saakka. Opinnäytetyössä käydään läpi myöskin työprosessivaihe vaiheelta, minkä jälkeen perehdytään tarkemmin opinnäytetyössä kohdattuihin haasteisiin ja saavutuksiin, kuten ylläpitopuolen työkaluihin ja salaukseen. Lopuksi opinnäytetyössä pohditaan mahdollisia muita toteutustapoja sekä asioita joita olisi voitu toteuttaa eri tavoin.</p>		
Asiasanat		
Ohjelmointi, WWW-sivut, Bootstrap, YouTube API, raportointi		

Author (authors)	Degree	Time
Timo Partanen	Bachelor of Business Administration	December 2017
Thesis Title		31 pages
The design and implementation of the Satakiri publishing platform		
Commissioned by		
Mikkeli city library		
Supervisor		
Arto Väätäinen		
Abstract		
<p>The subject of this thesis was to design and create a publishing platform for the Satakiri project. The thesis was commissioned by the city library of Mikkeli, that needed a publishing platform for their Satakiri competition to celebrate Finland's 100 years of independence in which school aged young people could participate. In the competition students ranging from elementary students all the way to college students could participate by sending all types of media that in their opinion represented our 100-year-old nation.</p>		
<p>This means that the publishing platform must be able to receive videos, sound files, images and text and be able to handle them correctly with administration tools. At the end of the competition every artwork with publishing permission should be available for public viewing at the Satakiri website. Also, one part of the commission was to find a suitable web hosting service for the Satakiri project and transfer all the files and databases to its server.</p>		
<p>The thesis contained all the important techniques, software and solutions which were used during the work process. These ranged from cPanel and Bootstrap all the way to YouTube's application programming interfaces. The thesis also introduced the work process step by step and right after that the thesis took a more serious look at all the challenges and achievements encountered during the process like administration tools and encryption. The end of the thesis discussed alternative execution methods and evaluated the thesis process.</p>		
Keywords		
Programming, web site, Bootstrap, YouTube API, reportage		

SISÄLLYS

1	JOHDANTO.....	5
2	FRONT-END WEBKEHYKSET	6
2.1	Ominaisuudet ja käyttötarkoitus.....	6
2.2	Bootstrap	8
3	KÄYTETYT OHJELMISTOT	10
3.1	MySQL Workbench.....	10
3.2	cPanel.....	12
4	JULKAISUALUSTAN TOTEUTUS.....	14
4.1	Tavoitteet.....	15
4.2	Prosessi.....	16
4.3	Käyttäjäkokemus	18
4.4	Ulkoasu.....	19
4.5	Tiedon tallentaminen	20
4.5.1	Youtube API:n käyttö palvelussa	22
4.5.2	Salaus ja yksityisyys	23
4.6	Ylläpidon kirjautuminen.....	24
4.7	Ylläpidon työkalut.....	24
5	PÄÄTÄNTÖ	27
	LÄHTEET.....	30

1 JOHDANTO

Opinnäytetyön toimeksiantajana toimii Mikkelin kaupunginkirjasto, joka on Mikkelin kaupungin ylläpitämä julkinen organisaatio. Kirjasto oli suunnitellut järjestävänsä syksyllä 2017 koululaisille suunnatun Satakiri-kilpailun, jonka ideana olisi toteuttaa kuvia, videoita, ääntä ja kaikenlaista mediaa, mikä edustaisi satavuotiasta maamme. Tämän kilpailun järjestämiseen Mikkelin kaupunginkirjasto tarvitsi henkilökuntansa lisäksi hanketyöntekijän sekä verkkosivujen luomiseen erikoistuneen harjoittelijan tuottamaan verkossa toimiva julkaisualusta, johon kaikki kilpailijat voisivat helposti lähettää työnsä.

Opinnäytetyön tavoitteena on esitellä miten ja millä tavoin Satakiri-julkaisualusta toteutettiin. Käydä myöskin läpi työssä käytetyt ohjelmistot ja tekniikat sekä perehtyä tarkemmin työprosessiin ja sen aikana kohdattuihin haasteisiin. Opinnäytetyö rajaa pois vaihtoehtoisia toteutustapoja sekä lopuksi pohdiskelee mitä asioita olisi voitu toteuttaa eri tavoin.

Opinnäytetyön luvussa kaksi ja kolme käydään pääasiassa yleisesti läpi avainmenetelmiä, mitä hyödynnettiin opinnäytetyötä tehdessä. Menetelmät vaihtelevat verkkosivujen kehyksistä, ylläpitosovelluksiin ja ohjelmistoihin saakka. Opinnäytetyössä ei nähty tarvetta esitellä perusteellisesti käytettyjä ohjelmointikieliä vaan ne on ainoastaan mainittu kolmannen luvun alussa.

Luvut neljä ja viisi käsittelevät perusteellisesti itse opinnäytetyön luomista ja sen prosessia. Luku kolme kattaa kaikki vaatimukset, jotka esiteltiin joko työn alussa, tai sen aikana, jotka vaativatkin joskus laajoja muutoksia työtä tehdessä. Kyseiseen lukuun on myös sisällytetty tekijän itse asettamat vaatimukset, jotka sopivat toimeksiantajalle hyvin mainiosti. Luvussa 4.1 pyritään luomaan opinnäytetyön lukijalle kuva työn etenemisestä ja sen loogisesta järjestyksestä, kuinka opinnäytetyö luotiin, kuten taas muut luvun neljä kohdat keskittyvät enemmän opinnäytetyön yksittäisiin haasteisiin ja mainitsemisen arvoisiin seikkoihin. Viimeinen luku kattaa tiivistelmän opinnäytetyön onnistumisesta, vastaanotetusta palautteesta, henkilökohtaisesta kritiikistä sekä pohdintaa mahdollisista muista keinoista, joita oltaisiin voitu käyttää opinnäytetyön aikana.

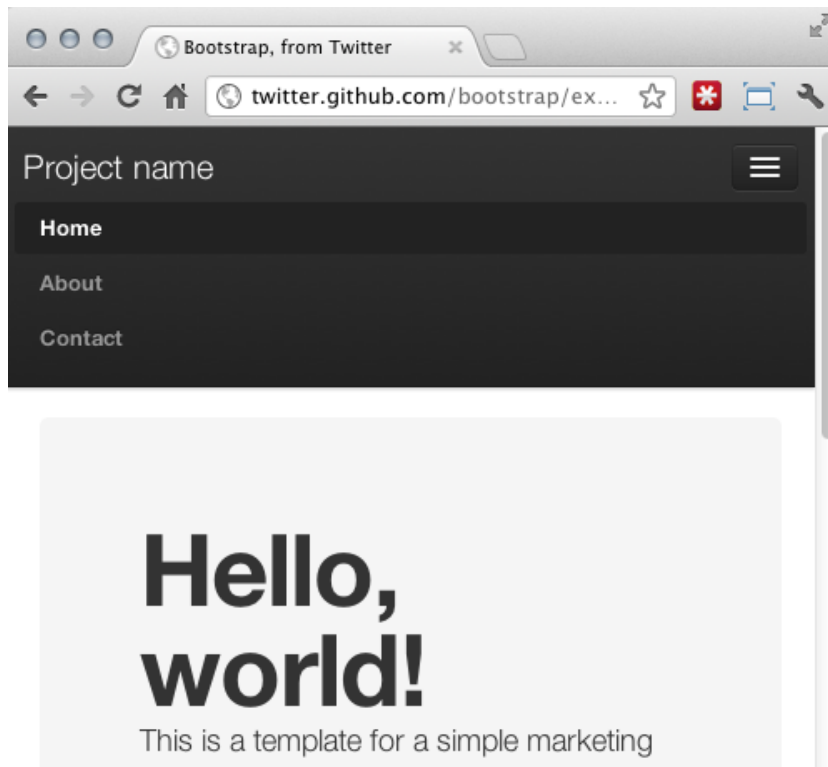
2 FRONT-END WEBKEHYKSET

Tässä luvussa esitellään yleisesti front-end webkehysten tyypillisiä ominaisuuksia ja käyttötarkoituksia. Väärinkäsityksiä välttämällä luvussa ei käsitellä Javascript pohjaisia websovelluskehystiä, kuten Angularia ja Reactia vaan ainoastaan CSS pohjaisia webkehystiä.

2.1 Ominaisuudet ja käyttötarkoitus

Front-end webkehystet ovat valmiita WWW-sivujen runkoja, jotka ovat suunniteltu siten, että käyttäjä pääsisi suoraan tuottamaan sisältöä verkkosivuilensa ilman, että hänen tarvitsisi luoda mittavaa pohjatyötä. Webkehystiä voidaan hyödyntää monenlaisissa tarkoituksissa staattisista ja dynaamisista sivuista websovelluksiin saakka. Esimerkkejä valmiiden front-end webkehysten käytöstä voi nähdä nykyään kaikkialla erilaisissa verkkokaupoissa, blogisivustoissa, yritysten kotisivuissa ja monissa muissa verkossa toimivissa palveluissa.

Front-end webkehysten ominaisuuksiin kuuluu HTML/CSS:llä rakennettu runko sekä tyypillisesti myöskin teemaan sopivat fontit, painikkeet, navigaatiopalkki, useita kuvakokoja tukeva responsiivisuus sekä Javascript laajennuksia (Guardian 2010). Myöskin joko ulkoa, tai kehysten mukana latautuneet kirjastot tuovat erilaisia ikoneita ja tyylejä kehittäjän käytettävyyteen. Javascript laajennukset tuovat yleensä nykyaikaisilta sivuilta tuttuja pieniä animaatioita ja siirtymisiä, jotka parantavat käyttäjäkokemusta niiden sulavuudellaan, mutta Javascriptiä ja JQueryä näkee joidenkin suurempien sivujen elintärkeissä toiminnoissakin, kuten verkkokauppojen ja valikoiden toiminnallisuudessa.



Kuva 1. Esimerkki mobiiliystävälliseksi muuttuvasta navigaatiopalkista Bootstrap ympäristössä.

Webkehyskiä käyttämällä sivut saavat yleensä tyylikkään ulkoasunsa lisäksi niin sanotun ammattimaisen rakenteen, jota muidenkin mahdollisten WWW-kehittäjien on jatkossakin helppo säätää ja muokata. Monet front-end webkehukset pyrkivät myöskin nopeaan suoritustehoon, mikä tarkoittaa lyhyempää aikaa sivustojen latauksessa ja toimintojen tapahtuessa (Red Frog Consulting 2011). Myöskin useiden kehysten mukana tuleva täysin responsiivinen rakenne erilaisille kuvasuhteille on aina hyödyksi nykypäivänä, kun mobiililaitteiden käyttö on erittäin yleistä. Monet webkehukset juuri sen takia tuovatkin nykyään täysin muuttuvat navigaatiopalkit, jotka muuttuvat mobiiliystävällisemmiksi kyseisillä kuvasuhteilla.

Opinnäytetyön alkupuolella käsiteltiin useita webkehyskiä julkaisualustan runkoksi, mutta lyhyen pohdinnan jälkeen projektissa päädyttiin käyttämään Creativen Bootstrap pohjaa, joka sopi niin ulkoasun kuin rakenteenkin kautta projektiin täydellisesti.

2.2 Bootstrap

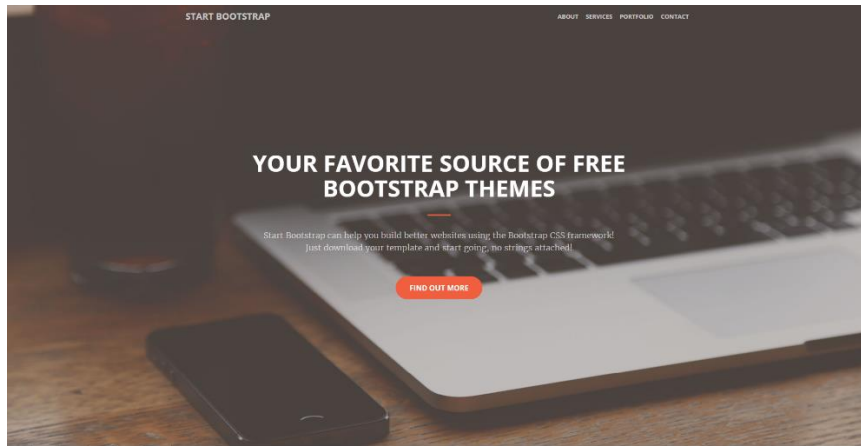
Bootstrap on ilmainen avoimen lähdekoodin webkehys, joka on suunniteltu toimimaan WWW-sivujen ja websovellusten front-end rakenteena (Get Bootstrap 2017). Bootstrap sisältää pääosin HTML ja CSS pohjalle tehtyjä valmiita malleja ja rakenteita, joita webkehittäjä voi muokata mieleiseksensä.



Kuva 2. Yksi useista Bootstrapin käyttämistä logoista

Bootstrap, joka alun perin tunnettiin nimellä Twitter Blueprint, on Mark Otton ja Jacob Thorntonin kehittämä webkehys, joka julkaistiin Bootstrap nimellään elokuussa 2011. Julkaisun jälkeen pieni ryhmä Jacobin ja Otton lisäksi on jatkanut Bootstrapin aktiivista kehitystä useiden versioiden läpi, jotka ovat tuoneet monien ominaisuuksien lisäksi myös tukia useimmille selaimille (Get Bootstrap 2017).

Nykypäivänä Bootstrap houkuttelee suurimman osan käyttäjistään helppokäyttöisyydellään, responsiivisellä rakenteellaan sekä selain- ja mobiili yhteensopivuudellaan (W3Schools). Myöskin monet Bootstrapillä tehdyt valmiit WWW-sivupohjat tuovat aiemmin mainittujen ominaisuuksien lisäksi, tyylikkäitä ulkoasuja, jotka ovat myöskin erittäin helposti muokattavissa pienillä HTML/CSS muutoksilla. Bootstrap on myöskin todennäköisesti maailman käytetyin WWW-sivujen teossa käytetty webkehys ja sitä hehkutetaan muun muassa sen selkeydestä sekä laajasta lisäosa tarjonnasta. Myöskin Bootstrapin mobiiliystävällinen ympäristö, responsiivinen grid sekä jQuery pluginit antavat selvän edun Bootstrapille kilpailijoihinsa nähden (GetBootstrap 2017).



Kuva 3. Creative Bootstrap esimerkki responsiivisesta ja modernista WWW-pohjasta

Kuvassa 3 on kuvakaappaus projektissa käytetystä Creative Bootstrap Onepage teemasta, joka osoittautui työssä erittäin hyödylliseksi työkaluksi. Teema sisältää tuen kaikille käytetyimmille selaimille (Explorer, Edge, Firefox, Chrome, Opera yms.) sekä täyden responsiivisuuden, joka osaa asetella teeman ”palaset” paikalleen kuvaruudun koosta riippumatta. Creative teema sisältää myöskin Javascriptillä animoituja osia navigaatiopalkissa sekä muuttamissa yksittäisissä napeissa, jotka reagoivat, kun käyttäjä siirtyy niiden kohdalle. Myöskin teeman moderni suunnittelu, joka sisältää suuren 1920x1080 kuvan aivan etusivun ylälaidassa on nykyään lähes aina vakio monien yrityksen sivuilla, mikä perustuu täysin sen luomaan positiiviseen käyttäjäkokemukseen ja avaraan tunteeseen (Start Bootstrap 2017).

```

color: #222222;
border-color: #e0e0e0;
background-color: #e6e6e6; }
.btn-default.active.focus, .btn-default.active:focus, .btn-default.active:hover, .btn-default:active.focus,
.open > .btn-default.dropdown-toggle.focus,
.open > .btn-default.dropdown-toggle:focus,
.open > .btn-default.dropdown-toggle:hover {
  color: #222222;
  border-color: #bfbfbf;
  background-color: #d4d4d4; }
.btn-default.active, .btn-default:active,
.open > .btn-default.dropdown-toggle {
  background-image: none; }
.btn-default.disabled.focus, .btn-default.disabled:focus, .btn-default.disabled:hover, .btn-default[disabled]
fieldset[disabled] .btn-default.focus,
fieldset[disabled] .btn-default:focus,
fieldset[disabled] .btn-default:hover {
  border-color: white;
  background-color: white; }
.btn-default .badge {
  color: white;
  background-color: #222222; }

```

Kuva 4. Kuvakaappaus Creativen CSS tiedostosta

Creative Bootstrap -teeman muokkaaminen on myöskin pääosin helppoa. HTML runko ja Javascript osat on aseteltu erittäin selkeästi teemassa loogisiin

paikkoihinsa sekä niiden sisällöt ovat helppoa luettavaa. HTML puolella kaikki osa-alueet on lajiteltu aikaisemmin mainittuihin ”paloihin” eli sectioneiden ja Divien kautta luotuihin lokeroihin. Javascript puolen pystyy myöskin aloitteleva käyttäjä ymmärtämään suhteellisen nopeasti kommentoitujen neuvojen ja selkeästi nimetyn koodin kautta. Ainoastaan CSS puoli kuvan 4 mukaan on toteutettu hieman vähemmän aloittelijaystävällisesti. Vaikka kaikkien nappien toiminta onkin nimetty niitä kuvaavilla termeillä, niin saattaa käyttäjälle tulla monesti hetkiä, jolloin jonkun tietyn asian löytäminen saattaa olla vaikeaa erittäin tiheään kirjoitetun 400 koodirivin joukosta.

3 KÄYTETYT OHJELMISTOT

Ohjelmassa toimivat lukujen 3.1 ja 3.2 mainittujen työkalujen lisäksi ohjelmointi, -ja merkkikielinä HTML5, CSS3, PHP, SQL sekä Javascript. Suurin osa palvelun toiminnasta luotiin PHP-ohjelmointikielellä, kun taas ulkoasuun ja käyttäjäkokemukseen pyrittiin vaikuttamaan pääosin CSS3:n sekä Javascriptin ominaisuuksilla.

Tässä luvussa esitellään yleisesti kaksi eniten käytettyä ohjelmaa, jotka näkivät paljon käyttöä opinnäytetyöprosessin aikana. Opinnäytetyössä ei nähty tarpeelliseksi esitellä pienempiä sovelluksia, kuten Sublime Text -tekstieditoria tai vähemmän käytettyjä sovelluksia, kuten GIMP -kuvankäsittelyohjelmaa.

3.1 MySQL Workbench

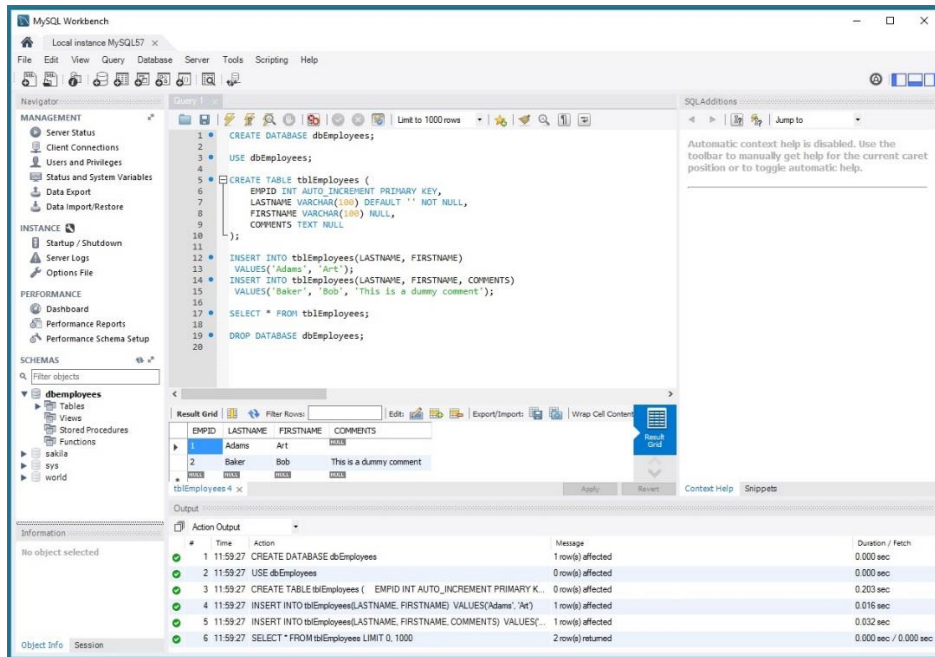
MySQL Workbench on visuaalinen tietokannan suunnittelutyökalu, joka sisältää tietokantojen kehitys, hallinta ja ylläpito mahdollisuudet. MySQL Workbenchin kehitys aloitettiin vuonna 2002 fabFORCE tiimin nimeämällä DBDesigner4 tittelillä, joka kuitenkin yhdistyi vuoden päästä MySQL AB yhtymän kanssa (fabFORCE 2003). Muutamien vuosien hankaluuksien jälkeen MySQL Workbench sai kuitenkin lippulaivan aseman yrityksessään, joka myöhemmin liittyi ruotsalaiseen Sun Microsystemsin kanssa (Cision PrWeb 2008).



Kuva 5. MySQL Workbench:n logo

MySQL Workbench jakaa toiminnallisuutensa pääosin viiteen kategoriaan. Kehitykseen, tiedon näyttämiseen (data modelling), serverien hallinnoimiseen, datan siirtämiseen sekä apuohjelmiin (MySQL 2017). Kehitys osa-alue pitää sisällään perinteisen tietokantojen luomisen, niiden yhteyksien säätämisen sekä sisäänrakennetun SQL-editorin käyttämisen. Tiedon näyttäminen tarkoittaa pääosin tietokantaa kuvaavien graafisien mallien luomista sekä taulu editorin (table editorin) käyttämistä. Servereiden ylläpito osa-alue kattaa nimensä mukaisesti käyttäjätunnuksien hallitsemista, tiedon seuraamista sekä varaversioiden luomista ja niihin palaamista. Viimeiset kaksi osa-aluetta tiedon siirtäminen sekä apuohjelmien käyttämisestä koskee pääasiassa MySQL Workshopin seurustelemista muiden ohjelmien kanssa (MySQL 2017). MySQL Workshopille on tehty useita tukevia ohjelmistoja, jotka voivat tukea käyttäjän toimintaa monilla tavoin. Palvelusta on myöskin helppo siirtää kokonaisia tietokantoja muihin palveluihin ainoastaan muutamalla hiiren klikkauksella.

MySQL Workbench on myöskin täysin Python ja Lua skriptausta tukeva ympäristö, johon on mahdollista hakea lukuisia muiden käyttäjien tekemiä plugineita tukemaan omaa projektiaan (Mysql Workbench 2010).



Kuva 6. Kuvakaappaus MySQL Workbench:n ohjelmistosta

Projektissa MySQL Workbench tuli pääosin aktiivisesti esiin työprosessin alkupuolella, kun tietokannat tauluineen ja tietoineen täytyi syöttää manuaalisesti paikoilleen. Myöhemmin Workbenchin käsikäyttö käyttö väheni pääosin julkaisualustan ylläpidon valmistuneiden CRUD-toimintojen ansioista, joka mahdollisti suurimman osan testaustyöstä niin palautusten, kuten käyttäjätunnuksien kanssa, mutta silti työprosessin aikana tuli hetkiä, milloin täytyi palata käyttämään Workbenchin ohjelmistoa. MySQL Workbenchin kokonaisien tietokantojen siirtotyökalu osoittautui myöskin hyödylliseksi, kun projekti siirrettiin cPanelin phpMyAdmin -palvelun kautta verkkoon.

3.2 cPanel

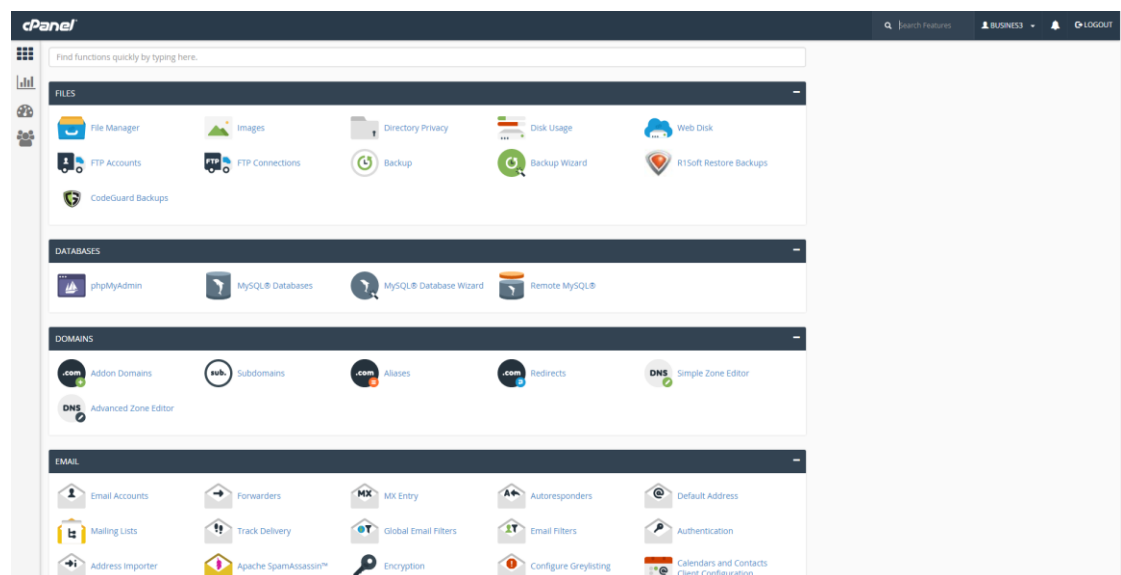
cPanel on Perl ohjelmointikielellä kirjoitettu Linux-pohjainen verkkoylläpidon hallintapaneeli, joka tarjoaa graafisen käyttöliittymän sekä työkalut sivun ylläpitoa varten. cPanelissa on kolmikerroksinen rakenne, joka tarjoaa erilaisia työkaluja ylläpitäjälle, jälleenmyyjälle kuten myöskin verkkosivun omistajalle, jotka voivat käsitellä sivun ominaisuuksia monilta eri alueilta (Forum cPanel 2011). Serverin ylläpitäjätökalut mahdollistavat muun muassa minkä tahansa tasoisten tunnuksien lisäämisen, muokkaamisen ja poiston, kun taas

jälleenmyyjien työkalut pystyvät ainoastaan luomaan käyttäjätason (User Account) tunnuksia. Jälleenmyyjällä on muokkaamisoikeudet ainoastaan heidän palvelunsa alle luotuihin tunnuksiin. Käyttäjätason työkalut ovat perinteisimpiä tunnuksia, jotka maksavat kuukausimaksua verkkopalvelunsa isännöinnistä. cPanel mahdollistaa kuitenkin useiden nettisivujen ylläpidon samoilta tunnuksilta, ettei käyttäjä tarvitse vaivautua luomaan eri tunnukset jokaiselle isännöidylle webpalvelulle (Tutorialspoint 2016).

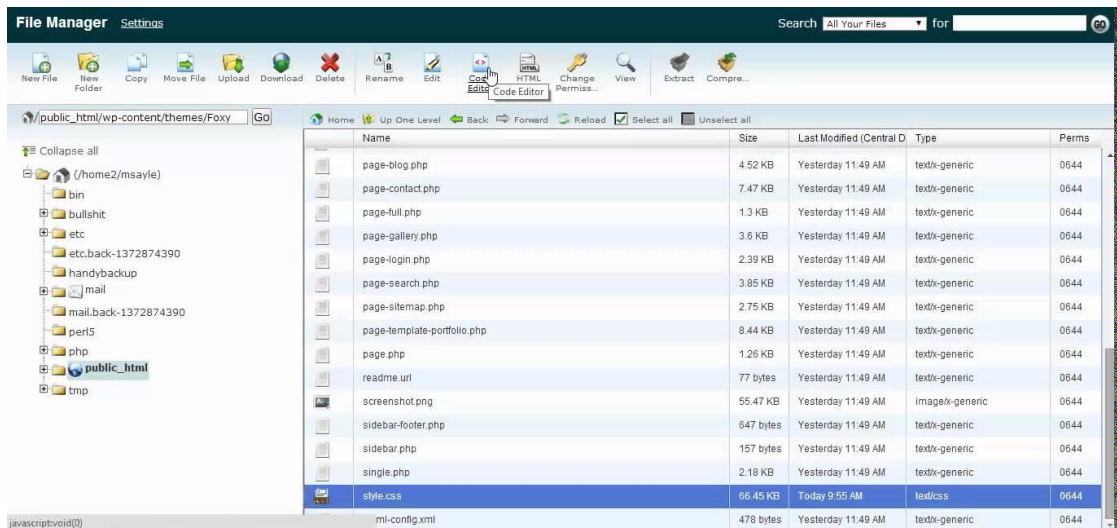


Kuva 7. cPanelin logo

Verkkosivun omistaja pystyy esimerkiksi lataamaan omatekemänsä WWW-sivut, tietokannat sekä mahdolliset laajennukset helposti verkkoon, mikä päivittää sivun yleensä välittömästi. Lisäksi cPanel tarjoaa ylläpitäjälle tietoa WWW-sivuston dataliikenteestä sekä vierailijamäärästä, mitkä saattavat olla tärkeitä monille yrityksille ja seuroille. Myöskin sähköpostituki ja sovellusrajapinnat useille palveluille (Apache), ovat mahdollisia Cpanelin kautta.



Kuva 8. Kuvakaappaus cPanelin hallintapaneelistä



Kuva 9. cPanelin tiedostonhallintatyökalu

Projektissa cPanel tuli varsinaisesti käyttöön työn loppupuolella, kun palvelulle valittiin webhotelli. Useiden ehdokkaiden jälkeen webhotelliksi valittiin suomalainen domainhotelli.fi, joka tarjosi asiakkailleen cPanelin käyttöliittymän sivustojen hallitsemista varten. cPanel osoittautui erittäin helppokäyttöiseksi tiedostojen siirrosta tietokantoihin asti, kun nekin pystyttiin hoitamaan sisäänrakennetun phpMyAdmin tietokantojenhallintatyökalun kautta. cPanelin tiedostojen hallintatyökalu mahdollisti kokonaisien kansiorakenteiden siirron helposti palvelimelle ainoastaan muutaman hiiren painalluksen kautta sekä ohjelmistoon tehty sisäinen koodieditori auttoi nopeiden muutoksien luomisessa, ilman että ylläpitäjän olisi tarvinnut manuaalisesti käydä lataamassa uusi tiedosto ainoastaan muutaman koodirivin takia.

4 JULKAISUALUSTAN TOTEUTUS

Tämä luku käsittelee julkaisualustan toteutusta alusta loppuun saakka. Luku 4.1 luettelee kaikki vaatimukset ja tavoitteet, joita asetettiin niin työprosessin alussa, kuten sen aikanakin. Luku 4.2 perehtyy tarkemmin työprosessiin, josta lukija saa parhaimman käsityksen työjärjestyksestä ja sen kulusta, kun taas muut luvut keskittyvät enemmän verkkopalvelun avaintekijöihin ja sen haasteisiin, joihin perehdytään yksityiskohtaisemmin.

4.1 Tavoitteet

Opinnäytetyön tehtävänä oli luoda koululaisille suunnattu julkaisualusta, johon kaikki Etelä-Savon nuoret pystyisivät lähettämään kuvia, videoita, tekstiä ja ääntä ja samalla osallistua kilpailuun, jonka rahapalkintojen suuruudet vaihtelivat sadoissa euroissa. Koululaisten tehtävänä oli luoda valitsemansa mediatyypin kautta kaikenlaista sisältöä, mitä arvostaa Suomessa, niin luontokuvista ja videoista, runoihin ja lauluihin asti.

Ulkoasuun haluttiin sinivalkoista selkeää teemaa kuvastamaan satavuotiasta maataamme. Kuitenkin taustakuva, muutamat ikonit sekä Mikkelin poikateatterin video tulivat ulkopuolelta, joten ainoastaan niiden asettelu kuului työtoimikuvaan. Ulkoasuun haluttiin myös sulauttaa Satakiri-hanketta tukevien yritysten logot sekä yhteyshenkilön yhteystiedot.

Palvelulla kuului olla myöskin salasana suojattu ylläpitopitopuoli, missä täytyi olla monenlaisia toimintoja kuten tietokantanäkymä, töiden tarkistaminen ja arviointi, töiden julkaisu, ylläpitäjien hallinta sekä loki ylläpidon kirjautumisista. Tietokantanäkymään vaadittiin perinteinen CRUD toiminnallisuus, joka mahdollistaa helposti pienien muutoksien tekemisen niin palautettuihin töihin kuten ylläpidon käyttäjätunnuksiinkin. Raati työkalulle ainoat vaatimukset olivat vain, että arvostelijoiden pitää pystyä näkemään työt selkeästi sekä antamaan niille arvosana jonkinlaisella asteikoilla pienen kommenttikentän kera. Viimeinen vaatimus ylläpitopuolelle oli, että palautetut eivät saisi näkyä julkisesti ennen ylläpidon hyväksyntää, joten sille täytyisi luoda myöskin oma työkalunsa.

Palvelulla täytyi olla myös julkisien töiden selaussivu, johon kaikki palautetut työt, joilla olisi lupa pystyttäisiin esittämään selkeästi kilpailun jälkeen. Selaussivulla täytyi olla myöskin hakutoiminto mahdollisuus yksittäisten töiden etsimistä varten sekä sivutusratkaisu, etteivät kaikki työt latautuisi yhdellä kertaa.

Palvelu tulisi myöskin tarvitsemaan webhotellin, joka ylläpitäisi WWW-sivuja verkossa. Tehtävään kuului siis sopivan edullisen, mutta myös tallennustilaltaan laajan webhotellin löytäminen lukuisien toistensa kanssa kilpailevien pal-

veluiden joukosta. Webhotellin hallintapaneelista täytyisi ottaa myöskin selvää, mutta suositukseksi sanottiin perinteinen cPanel. palvelulla kuului myöskin olla kaikki käyttäjätunnukset sekä salasanat lähetettynä ylläpitäjille käyttöohjeiden kera, jotta heillä olisi yhteydet ohjelman lisäksi muihinkin tärkeisiin toimintoihin, kuten tietokantoihin sekä webhotelliin.

4.2 Prosessi

Palvelun toteuttaminen aloitettiin alustan valinnalla, jossa harkittiin useita termejä kuten Bootstrap, Wordpress, Drupal, Wix sekä raaka koodin käsikirjoittaminen tekstieditorilla. Jokainen kyseisistä palveluista ja tekniikoista tulisi silti käyttämään MySQL Workbench:n tukea tietokantoja käsitellessä. Ensimmäiseksi hylättiin Wix, koska kyseisellä palvelulla ei ollut tukea Linux alustalle, jolla palvelu oli tarkoitus luoda. Wordpress ja Drupal olisivat myöskin varmasti tuoneet helpottavia ominaisuuksia, mutta samalla ne olisivat myöskin rajoittaneet projektin toimintaa erilaisin keinoin. Kyseisien rajoitusten jälkeen projektissa päädyttiin ns. raakaan koodin käsin kirjoittamiseen, joka tulisi hyödyntämään Bootstrapin tarjoamaa Creative webkehystä, mikä olisi täysin tekijän muokattavissa, mutta samaan aikaan tarjoaisi upean ulkoasun ja responsiivisuuden WWW-sivulle.

Ensimmäiseksi projekti aloitettiin simpeleillä rautalankamalleilla ja vaatimusmäärittelyllä. Voltairin kortit sekä havainnollistavat rautalankamallit auttoivat hahmottamaan projektin laajuutta sekä muistioon kirjoitetut huomioon otettavat seikat helpottivat projektissa alkuun pääsemistä. Myöskin projektin alussa kerätyt hyödylliset www-linkit avustivat työprosessia.

Vaatus: 03	Vaatumuksen tyyppi: Toiminnallinen	Toiminto/käyttötapaus
Kuvaus: Ylläpidon tietokantanäkymä (admin)		
Vaatumuksen perustelu: Helpottaa palautettujen töiden seuraamista.		
Prioriteetti: 1		
Hyväksymiskriteeri: Toimiva kirjautumisruutu, joka ohjaa ylläpitosivulle. Ylläpitosivu sisältää CRUD-toimenpiteet, ja materiaalin hyväksymisen.		
Vaikutus asiakkaan tyytyväisyyteen		Vaikutus asiakkaan tyytymättömyyteen
Riippuvuudet		Ristiriidat
Tukimateriaali		
Historia		
Lähde:		

Kuva 10. Esimerkki käytetystä Voltairen-kortista

Suunnittelun jälkeen, ensimmäiseksi palveluun luotiin tietokanta, jonka jälkeen kehitettiin tietokantaan sekä tiedostorakenteeseen tallentavaa lomake, joka on täysi perusedellytys palvelun toimivuudelle. Toimivan lomakkeen jälkeen ylläpidon työkalut alkoivat kasautua paikalleen. Tietokantanäkymä CRUD toiminnollisuuksineen sekä töiden arvostelusivu kehittyivät suhteellisen nopeasti. Sen jälkeen syntyi julkisien töiden selaussivu ja sen logiikan minkä mukaan ylläpidon täytyy hyväksyä jokainen työ joka voi näkyä julkisesti kaikille. Myöskin ylläpidon turvallinen kirjautuminen ja ylläpitäjien hallinta saivat muotonsa projektin alkupuolella.

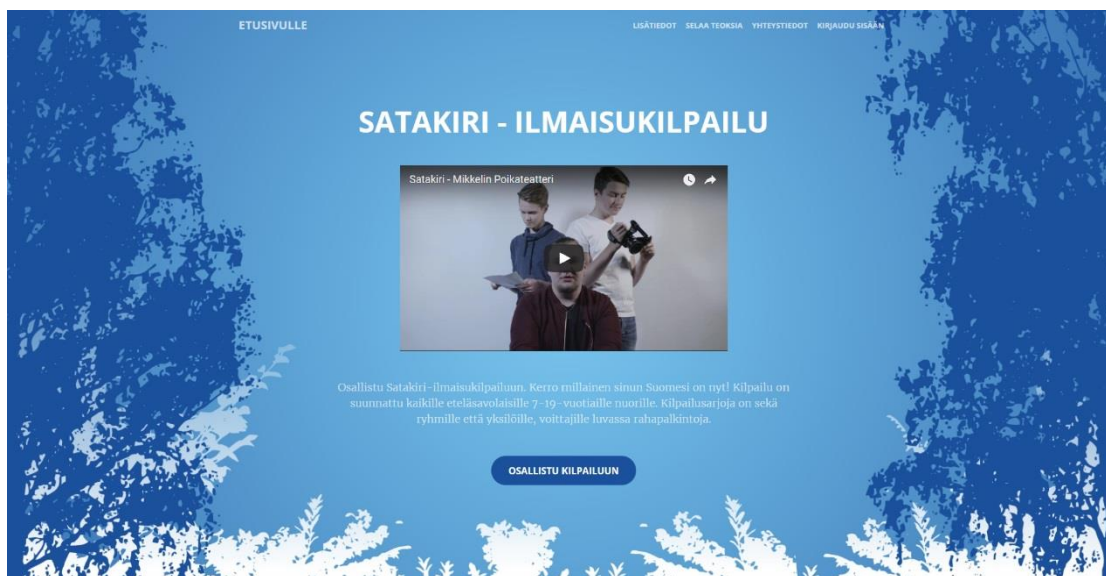
Tärkeiden ominaisuuksien jälkeen palveluun pystyttiin alkaa kehittää tukevia tekijöitä kuten ulkoasua, lokitietoja, hakutoimintoja, tukevien yritysten logojen asettelua ja muuta samanlaista vähemmän elintärkeää sisältöä ohjelmalle.

Ohjelman tunnettua valmiilta, sille luotiin vielä 4-sivuinen käyttöopas suurikokoisine kuvineen helpottamaan tulevia ylläpitäjiä työssään. Käyttöopas ohjaa ylläpitäjää kädestä pitäen kaikkien työkalujen läpi sekä sisältää ehdotuksia, kuinka käyttää ylläpitotyökaluja mahdollisimman tehokkaasti.

4.3 Käyttäjäkokemus

Käyttäjä voi löytää Satakiri-palvelun monella tapaa. Sadat fyysiset painokset, jotka sisältävät kännykällä luettavan QR-koodin sekä mainokset Mikkelin radiossa ja Mikkelin kaupunginkirjaston sivuilla ohjaavat käyttäjän palvelun etusivulle. Lisäksi simppele ”googletus” näyttää palvelun ensimmäisten hakutulosten joukossa.

Sivulle saapuessaan käyttäjä näkee muokatun Creative Bootstrap One Page mallin, joka on samaan aikaan sekä avara, että selkeä. Osallistu kilpailuun painike on asetettu nähtäville suoraan Mikkelin Poikateatterin intro videon alapuolelle kirjaston vaatimusten mukaisesti. Yläpalkissa olevat painikkeet auttavat käyttäjää navigoimaan palvelun etusivulla, joka samalla sisältää myöskin ylläpidolle linkin ylläpidonkirjautumisruutuun.



Kuva 11. Satakiri-palvelun etusivu

Alaspäin mentäessä sivu sisältää kaikenlaista infoa, mitä kilpailuun osallistuva voi tarvita ja kyseisiä infopalkkeja pystyy avaamaan, jos kaipaa vielä lisätietoja. ”Selaa teoksia” kohdassa käyttäjää muistutetaan vielä mahdollisista mediamuodoista sekä sen alapuolella oleva painike johtaa käyttäjän suoraan käyttäjäpalautusten sivulle. Selaa teoksia sivulla näkyvät kaikki ylläpidon hyväksymät työt, joille käyttäjä on antanut luvan näkyä julkisesti kilpailun jälkeen. Käyttäjä pystyy simppelellä katselemaan muiden palautuksia painamalla tie-

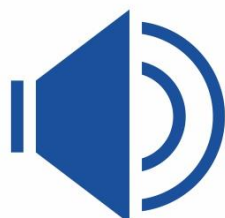
dostomuotoa esittävästä ikonista, jolloin teos aukeaa Lightbox-ikkunaan mediamuodosta piittaamatta. Lisäksi käyttäjä pystyy etsimään hakupalkin kautta töitä, jolloin palvelu rajaa tulokset hakuehtojen mukaisesti.

Kilpailuun osallistuessa käyttäjä painaa etusivulla olevaa ”Osallistu kilpailuun” painiketta, jolloin palvelu ohjaa muutaman kysymyksen kautta käyttäjän lomakesivulle. Lomakkeen kaikki infokentät on asetettu pakollisiksi, joiden lähettämisen jälkeen palvelu kiittää käyttäjää ja ohjaa hänet takaisin etusivulle. Virhetilanteessa kuten väärän tiedostomuodon tapauksessa, palvelu kertoo käyttäjälle virheestä ja ohjaa takaisin lomakesivulle.

4.4 Ulkoasu

Palvelu kehitettiin Creative Bootstrap pohjalle, jonka ulkoasua on muokattu erittäin paljon. Sivuston kokonaisrakennetta on järjestelty sopivammaksi sekä sivun oranssi väriteema on vaihdettu sinivalkoiseen satavuotiasta edustavaan Suomi tyyliin. Sivuston taustakuva sekä muutamat ikonit olivat kirjaston ulkopuolelta tilaamia teoksia, jotka täytyi asetella sivulle järkevästi. Myöskin tukevien yritysten logot voi löytää ruudun alareunasta linkkiensä kera.

Ylläpitopuolelle luotiin simppele äylaitetyylinen kuusivalintainen ruutu, jonka pitäisi olla helppokäyttöinen jo ensimmäisellä kerrallakin. Jokaisella ruudulla on oma ikoninsa ja lyhyt kuvaus, joka kertoo työkalun toimintaperiaatteen lyhyesti.



Kuva 12. Esimerkki käytetystä ikonista

Palautettujen töiden sivu käyttää yhdeksän ruutuista mallia, joita klikkaamalla käyttäjä saa haluamansa työn auki. Jokaisella työllä on sen tiedostomuotoa

esittävä kuva, joka kertoo käyttäjälle jo ennen tiedoston avaamista, minkälaista mediaa kyseinen tiedostomuoto sisältää ja kun käyttäjä asettaa hiiren työn päälle hän näkee kyseisen työn nimen sekä tekijän. Kaikki työt avautuvat Lightbox-ikkunaan, joka asettuu siististi näytön keskelle.

4.5 Tiedon tallentaminen

Lomakkeen lähettämisen jälkeen linkitetty tiedosto tallentuu palvelun kansiorakenteeseen, kun taas kaikki lomakkeelta kerätty tieto varastoidaan SQL-tietokantaan. Palvelu luo käyttäjän etunimestä ja teoksen nimestä tunnisteiden, jonka eteen se arpoo 6 numeroisen numerosarjan estääkseen mahdollisia virheitä samannimisien henkilöiden kohdalla. Sen jälkeen ohjelma asettaa halutun tiedostosijainnin sekä tunnisteiden samaksi pötköksi ja luo kyseisten ohjeiden perusteella uniikin kansion, johon käyttäjän liite tallennetaan.

```
//JOS TEKSTI
} else {
    $teksti = $_POST['teksti'];
    $numero = rand(1,999999);
    $tunniste = $etunimi . ";" . $numero;
//LUODAAN KANSIO
if (!is_dir('/var/www/html/img/teokset/' . $tunniste . '.' . $numero)) {
    // Jos kansio ei ole olemassa, luo se
    mkdir('/var/www/html/img/teokset/' . $tunniste . '.' . $numero , 0777, true);
} else{
    echo "kansio on jo olemassa";
}
file_put_contents("/var/www/html/img/teokset/" . $tunniste . "." . $numero . "/" . $etunimi . ".txt" , $teksti);
```

Kuva 13. Tiedon tallentaminen kansioon

Kansion luotuaan ja useiden tarkastusten, kuten kelvollisuuden ja kokotarkistuksien jälkeen, liite tallennetaan uuteen vastaluoettuun kansioon.

```
// Jos ok, ladataan
} else {
    if (move_uploaded_file($_FILES["fileToUpload"]["tmp_name"], $target_file)) {
        ?>
```

Kuva 14. Lähetetyn tiedoston siirtäminen vastaluoettuun kansioon

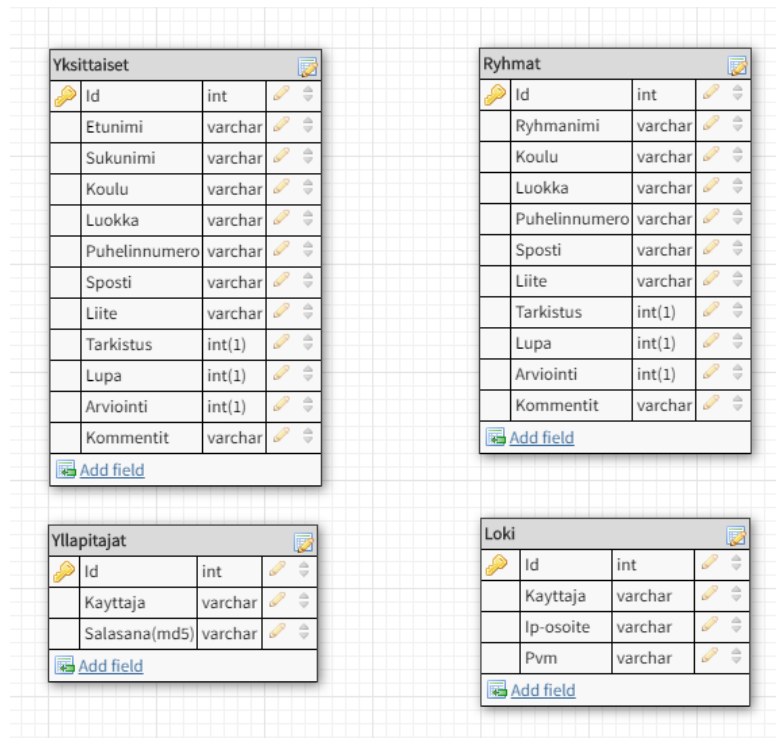
Samaan aikaan ohjelma tallentaa myöskin tärkeät tiedot tietokantaan tunnisteineen. ”Liite” kohta on polku käyttäjän tiedostoon, jota monet ylläpidon työkalut viittaavat aktiivisesti muutoksia tehdessä.

```

if($uploadOk!=0) {
//Sytetään tiedot tietokantaan
$sql="INSERT INTO Yksittaiset (etunimi, sukunimi, koulu, luokka, sposti, puh, liite, tarkistus, lupa, arviointi)
VALUES
('$etunimi','$sukunimi','$koulu','$luokka','$sposti','$puh','$target_file',0,'$lupa', '$Ei arvioitu')";

```

Kuva 15. Lomakkeelta kerättyjen tietojen tallentaminen tietokantaan



Kuva 16. Tietokantakaavio palvelun käyttämästä tietokannan tauluista

Yläpuolisen kuvan 16 kaavio luo selkeän kuvan ohjelmiston tietokannasta ja sen toiminnasta. Tietokannan tauluilla ei ole relaatioita, koska palvelua suunniteltaessa nähtiin, ettei näin yksinkertaisessa tallennusmallissa olisi tarvetta niille ja ne olisivat ainoastaan mutkistamassa asioita.

Kaikki yksinkertaiset tiedot, kuten sähköposti ja puhelinnumero tallentuvat normaalisti paikoilleen, mutta muutamat kohdat kuten "tarkistus", "lupa" ja "arviointi" toimivat ainoastaan asteikolla 0-1/5. "Tarkistus" kenttä tarkoittaa ylläpidon antamaa hyväksyntää, että työ saa näkyä julkisesti, kun taas "Lupa" kohta on varattu itse lähettäjälle, jos hän haluaa työnsä näkyvän julkisesti. Jos mo-

lemmat kentät ovat positiivisia, palvelun töiden katselusivu näyttää työn julkisesti. ”Arviointi” kenttä on varattu ylläpitoa varten, johon palvelu antaa ylläpitäjän tähtiasteikon kautta valitseman arvosanan 1-5 väliltä.

4.5.1 Youtube API:n käyttö palvelussa

Youtube API eli rajapinta on Googlen kehittäjien luoma keino tuoda YouTube järkevästi käyttäjän omille verkkosivulle, sovellukseen tai laitteeseen. Tämä rajapinta mahdollistaa maailman ylivoimaisesti suosituimman ja selkeimmän videontoistopalvelun käytön Satakirin palvelussa käyttäjien omien raakojen videotiedostojen sijaan. Paremmat videokontrollit sekä YouTube tilin käyttö myöskin vapauttaa tilaa webhotellilta, joka veloittaa enemmän suurista tilämääristä.

```
// Tarkista, että Authentication key on olemassa
$tokenSessionKey = 'token-' . $client->prepareScopes();
if (isset($_GET['code'])) {
    if (strval($_SESSION['state']) !== strval($_GET['state'])) {
        die('The session state did not match.');
```

```
    }
    $client->authenticate($_GET['code']);
    $_SESSION[$tokenSessionKey] = $client->getAccessToken();
    header('Location: ' . $redirect);
}
if (isset($_SESSION[$tokenSessionKey])) {
    $client->setAccessToken($_SESSION[$tokenSessionKey]);
}
// Tarkista, että Access Token on oikea
if ($client->getAccessToken()) {
    $htmlBody = '';
    try{
        // Vaihda tämän arvo, sillä minkä haluat ladattavan.
        $videoPath = $target_file;
        // Luo snippet titlen, kuvauksien, tagien ja kategorian kera
```

Kuva 17. Uuden pääsyavaimen haku YouTube latausta varten

YouTube API:n käyttö tulee palvelussa esiin, kun käyttäjä päättää lähettää videon, jolla osallistuu kilpailuun. Tällöin palvelu lähettää videon Satakirin YouTube kanavalle listaamattomana videona, eli ainoastaan linkin saanut voivat katsella sitä. Tämä tapahtuu siten että, jokainen kerta, kun käyttäjä lähettää videon, palvelu hakee uuden ”Access Tokenin”, eli koodin jolla palvelu voi la-

data videoita YouTubeen ilman kirjautumista. Pääsyn saatuaan palvelu lähettää videon palasina YouTubeen (chunk), joka kestää yleensä huomattavasti pitempään kuin muiden mediamuotojen tallentaminen. Palvelu saa myöskin samaan aikaan YouTubeelta osoitteen, jonka se tallentaa normaalin tiedostopolun kohtaan, mitä palvelu käyttää monissa toiminnoissaan myöhemmin.

4.5.2 Salaus ja yksityisyys

Julkaisualusta tallentaa sekä käyttäjätunnuksen, että salasanan tietokantaan omaan tauluunsa, joista jälkimmäinen salataan MD5 salauksella. Tämä mahdollistaa sen, että ylläpitäjät eivät näe toistensa salasanvoja, edes ylläpidon valikoissa vaan ainoastaan heidän käyttäjätunnuksensa. Salasanan vaihto onnistuu normaalisti myöskin ylläpidon puolella, jossa palvelu vaihtaa edellisen salasanan uuteen salattuun valintaan.

Ohjelma sisältää myöskin lokitiedot kaikista ylläpidon kirjautumista, joissa se kerää käyttäjän, IP-osoitteen sekä kellon ajan jokaisesta kirjautumisesta. Tämä oli Mikkelin kaupungin kirjaston vaatimus, koska jokaisella käyttäjällä on oikeus tietää, milloin heidän henkilötietojaan tai liitettyjä mediatiedostoja ollaan mahdollisesti katseltu.

```
$luokka = stripslashes($luokka);
$posti = stripslashes($posti);
$puh = stripslashes($puh);

$etunimi = preg_replace('/[^\w\d-9öäåöä\ ]/', '', $etunimi);
$sukunimi = preg_replace('/[^\w\d-9öäåöä\ ]/', '', $sukunimi);
$koulu = preg_replace('/[^\w\d-9öäåöä\ ]/', '', $koulu);
$luokka = preg_replace('/[^\w\d-9öäåöä\ ]/', '', $luokka);
```

Kuva 18. Erikoismerkkin poistaminen lomakkeelta kerätyistä tiedoista

Ohjelman käyttäjäpuolella on myöskin tekijöitä, jotka estävät mahdollisen SQL-injektion. Kaikenlaisten erikoismerkkin jälkikäteen poistaminen lomakkeen kirjoituskentistä, pitäisi estää perinteiset SQL-injektio yritykset kätevästi. Tämä toteutettiin komennoilla kuten ”stripslashes” ja ”preg_replace”.

4.6 Ylläpidon kirjautuminen

Ylläpidon kirjautuminen on toteutettu pääasiassa käyttämällä SQL-tietokantaa sekä istunto (session) toimintoja. Kirjautumissivu hakee tietokannasta käyttäjän tunnuksen sekä salatun salasanan, joita se vertaa käyttäjän syöttämiin tietoihin. Jos arvot täsmäävät, palvelu ohjaa käyttäjän ylläpidon etusivulle ja aloittaa samalla istunnon sekä kerää IP-osoitteen ja nimen palvelu lokitietotauluun.

```
//Tarkista käyttäjän nimi, salasana ja ylläpito oikeudet tietokannasta.
$sql="SELECT uid FROM Kayttajat WHERE username='$username' and password='$password' and oikeudet='1'";
$result=mysqli_query($db,$sql);
$row=mysqli_fetch_array($result,MYSQLI_ASSOC);

//Jos arvot täsmäävät, luo istunto.
//Muuten virhe.

if(mysqli_num_rows($result) == 1)
{
```

Kuva 19. Palvelu tarkistaa täsmäävätkö käyttäjän syöttämät tiedot tietokannan kanssa

```
//Koodi tarkistaa aina onko käyttäjä kirjautunut
session_start();
$user_check=$_SESSION['username'];

$ses_sql = mysqli_query($db,"SELECT username FROM Kayttajat WHERE username='$user_check' ");
$row=mysqli_fetch_array($ses_sql,MYSQLI_ASSOC);

$login_user=$row['username'];

if(!isset($user_check))
{
header("Location: kirjautuminen.php");
}
}
```

Kuva 20. Tarkistus koodi sivujen alussa, joka seuraa onko käyttäjä kirjautunut

Istuntoa ylläpidetään tarkista.php tiedoston avulla, joka on lisätty jokaisen ylläpidonsivun alkuun. Tämä mahdollistaa sen, ettei kukaan ulkopuolinen voi ohjata itseään ylläpidon sivulle simppelein URL-osoitteen muokkailun jälkeen. Jos käyttäjä yrittää navigoida itsensä kirjautumattomana esimerkiksi yllapito.php sivulle, palvelu ohjaa käyttäjän takaisin kirjautumaan.

4.7 Ylläpidon työkalut

Ylläpidolla on viisi päätyökalua. Tietokantanäkymä, Raati, Arkisto, Top -listat sekä Julkaisu sisältävät kaikki omat käyttötarkoituksensa, joista jotkin sisältävät myös omia lisätyökalujaan.

Ylläpito



Tietokanta

Näkymä kaikkien vastaanotettujen töiden tiedoista, sekä ylläpitäjien tunnuksista. Lisää, muokkaa, poista.



Raati

Arvostele vastaanotettuja töitä asteikolla 1-5. Voit myöskin kirjoittaa lyhyen kirjallisen arvion teoksesta.



Arkisto

Lista kaikista vastaanotetuista töistä lomaketietoineen. Sisältää myöskin arvostelemattomat teokset.



Top -listat

Lista parhaiten arvostetuista töistä lomaketietoineen. Simppeli työkalu mahdollisten voittajien löytämisessä.



Julkaisu

Työkalu vastaanotettujen töiden julkaisemiselle. Työt eivät näy julkisesti ennen kuin ne on hyväksytty täällä.



Kirjaudu ulos

Takaisin etusivulle

Kuva 21. Ylläpidon työkalut

Tietokanta-työkalu sisältää nimensä mukaisesti näkymän kaikista tietokannoista. Näkymä sisältää perinteiset CRUD-työkalut (lisää, katsele, muokkaa, poista) sekä yhden napin takaa myöskin lokitiedot, että ylläpitäjien tunnuksiset salattuna CRUD-toiminnollisuksiin. Kaikki CRUD-työkalut käyttävät toiminnossaan ”liite” kohdan sisältöä, eli esimerkiksi Matin palauttama työ ”Jokimaisema” näkyisi liitekohdassa esimerkiksi: ”img/teokset/Matti.Jokimaisema;25851/Kuva1.jpg”. Tämä on suora tiedostopolku Matin työhön, joka sisältää Nimen ja työn nimen lisäksi pienen arvotun numerosarjan välttääkseen virheitä mahdollisten duplikaattien tapahtuessa. Kyseistä tiedostopolkua käytetään, kun käyttäjä haluaa esimerkiksi poistaa teoksen, jolloin työkalu poistaa teoksen sekä tietokannasta, että kansioista. YouTube video tapauksessa liitekohdassa lukee ainoastaan YouTuben antama osoite, jota palvelu käyttää muuten samalla tavalla kuin normaalia kansiorakenteestakin napattua osoitetta. YouTube videot tallentuvat ainoastaan YouTuben palvelimelle, eivätkä ne luo kansiota ohjelman kansiorakenteeseen.

Tietokanta

Yksittäiset

Etunimi	Sukunimi	Koulu	Luokka	Sähköposti	Puhelinnumero	Lite	Julkaisu	Lupa	Arviointi	Kommentit	Toiminnot
Piija			7-9			Piija.teos.644458/Piija.bt	Ei	Kyllä	Ei arvioitu		Muokkaa Poista
Johanna			1-3			https://www.youtube.com/embed/vbLVR-iNcag	Ei	Kyllä	Ei arvioitu	Kiva tusti onnistui! Nyt tässä menossa enemmän leikkailua kuin aikaisemmin	Muokkaa Poista
Annamari			4-6			Annamari_Sininen ja valkoinen.325138/IMG_20170911_180239.jpg	Ei	Odottaa	Ei arvioitu		Muokkaa Poista
piija			7-9			piija.teos.964751/piija.bt	Ei	Kyllä	Ei arvioitu		Muokkaa Poista
Kiia			4-6			Kiia_Suomi 150650400615067819738051434152979.jpg	Ei	Odottaa	Ei arvioitu		Muokkaa Poista
Olli			4-6			Olli.teos.775354/Olli.bt	Ei	Odottaa	Ei arvioitu		Muokkaa Poista
Annikka			4-6			Annikka.teos.143146/Annikka.bt	Ei	Odottaa	Ei arvioitu		Muokkaa Poista
Jere			4-6			Jere.teos.60281/Jere.bt	Ei	Kyllä	Ei arvioitu		Muokkaa Poista
Aaro			4-6			Aaro.teos.757368/Aaro.bt	Ei	Odottaa	Ei arvioitu		Muokkaa Poista
Isko			4-6			Isko.teos.600849/Isko.bt	Ei	Odottaa	Ei arvioitu		Muokkaa Poista
Yasmin			4-6			Yasmin.teos.65535/Yasmin.bt	Ei	Odottaa	Ei arvioitu		Muokkaa Poista
Helmi			4-6			Helmi.teos.866361/Helmi.bt	Ei	Odottaa	Ei arvioitu		Muokkaa Poista



Kuva 22. Kuvakaappaus tietokantanäkymästä

Raati työkalu laittaa kaikki arvostelemattomat työt kategorisoituna kilpailuryhmien mukaan järjestykseen, josta ylläpitäjän on helppo antaa yhdestä viiteen arvostelu sekä mahdollisuus kirjoittaa lyhyt kommentti. Raati työkalu sisältää myös nopean ”Poista” painikkeen mahdollisten epäasiallisien töiden poistoa varten, ettei käyttäjän tarvitse vaivautua siirtymään poistamaan sitä tietokantanäkymän kautta. Kyseistä työkalua pystyy myöskin rajaamaan yläpalkin työkalun avulla, jos haluaa tarkastella esimerkiksi ainoastaan ylä-astelaisten palautettuja töitä.

Arkisto ja Top-listat työkalujen toimintaperiaate on sama. Arkisto näyttää kaikki työt lomaketietoineen listattuna kilpailuryhmien mukaan arvosteluista ja julkaisuluvista piittaamatta, kuten taas Top-listat työkalu näyttää ainoastaan parhaat työt jokaisesta kilpailusarjasta. Näistä jälkimmäinen työkalu luotiin ainoastaan helpottamaan mahdollisten voittajien löytämistä.

ARKISTO Kaikki Vahvista

Yksittäiset työt ylä-aste (7-9)

Etunimi	██	Sukunimi	████	Etunimi	██	Sukunimi	████
Koulu	██████	Luokka	██	Koulu	██████	Luokka	██
Sähköposti	██████	Puhelinnumero	██████	Sähköposti	██████	Puhelinnumero	██████
Arvio	██	Lupa näkyä	██	Arvio	██	Lupa näkyä	Kyllä
Kommentit				Kommentit			

Yksittäiset työt lukio/ammattikoulu (L/A)

Ei Arvosteltavia töitä

Ryhmätyöt ala-aste (1-3)

TAKAISIN

Kuva 23. Kuvakaappaus Arkisto-työkalusta

Viimeinen työkalu julkaisu, listaa kaikki arvostellut työt, joille on annettu lupa näkyä samaan listaan ylläpitäjän nähtäväksi. Työkalu luotiin, jotta ylläpitäjien ei tarvitsisi manuaalisesti etsiä kaikkia palautusluvan saaneita töitä kymmenien joukosta ja joutua täten julkaisemaan kaikki yksitellen. Ylläpitäjä pystyy työkalun avulla julkaisemaan kaikki työt painamalla ”Julkaise kaikki” painiketta, jolloin kaikki työt tulevat julkisesti esiin palvelun palautettujen töiden sivulle. Työkalu mahdollistaa myös yksittäisien töiden julkaisun tarvittaessa.

5 PÄÄTÄNTÖ

Tässä luvussa arvioidaan opinnäytetyön onnistumista sekä pohditaan erilaisia palvelun kehittämisideoita. Luvussa käsitellään myös jatkokehitysmahdollisuuksia, joita tuli esiin opinnäytetyö prosessin aikana.

Kokonaisuutena opinnäytetyö onnistui. Kaikki olennaiset osat tilauksien vastaanottamisesta, ylläpidon työkaluista, julkisien töiden selaussivuun saakka onnistuivat hyvin. Ohjelmisto koki oikeaa käyttöä vastaanottaessaan koululaisien töitä sekä toimi helpottavana työkaluna Mikkelin kaupunginkirjaston henkilökunnalle, kun he valitsivat eri kilpailuryhmien voittajia useiden kymmenien palautusten joukosta. Tietoturva piti pintansa, eikä mahdollisia tunkeutumisia

tullut vastaan yhtäkään. Myöskin kaupunginkirjastolta vastaanotettu palaute oli erittäin positiivista.

Projektin aikana esiintyi myös monia kehittämisideoita. Tietoturva-asioissa SQL-injektion estoon sekä ylläpidon kirjautumiseen olisi voinut panostaa enemmän. Erikoismerkkien poistaminen estää tavallisia ilkeyöntekijöitä tekemästä haittaa ohjelmalle, mutta kyseisen alan ammattilaiset varmasti löytäisivät tien tunkeutua ylläpidon työkaluihin ja tietokantoihin. Ylläpidon kirjautuminen toteutettiin perinteisellä kirjautumisruudulla, jonka salasana salaus toimi md5 salauksella sekä sessioita käyttämällä. Loppujen lopuksi kyseiset turvatoimet jäivät kuitenkin viimeiseen versioon voimaan, koska kirjastotyöntekijät, sekä työni ohjaaja pitivät niitä riittävinä eivätkä uskoneet, että kukaan rupeaisi asiakseen järeämmin ottein tunkeutumaan pieneen koululaisille suunnattuun kilpailuun.

Ulkoasun kehittämiseen olisi myöskin voinut panostaa hieman enemmän erityisesti ylläpidon puolella, koska luonnollisesti ylläpitäjätkin ovat samalla ohjelman käyttäjiä samalla tavalla kuten kilpailijatkin. Näitä parannuksia olisi voinut tehdä tietokantanäkymään sekä arkiston luettavuuteen. Myöskin koodin uudelleenkäytettävyyden ympärille suunnitteluun olisi pitänyt panostaa enemmän, koska työprosessin aikana tuli hetkiä, jolloin pienen asian tekeminen tarvitsi muutoksia useisiin eri paikkoihin.

Opinnäytetyön kirjallinen osuus onnistui myöskin hyvin. Tekstiä syntyi tasaiseen tahtiin neljästä viiteen sivua viikossa, jonka tueksi haettiin myöskin useita erilaisia kuvakaappauksia. Opinnäytetyön laajuutta olisi mahdollisesti voinut saada enemmän erilaisien asioiden yksityiskohtaisemmalla kertomisella, mutta prosessin aikana uskottiin, että kaikki olennainen tieto niin teoriasta, kuten käytännönsuudesta on saatu jo kerrottua.

Yhteenvetona opinnäytetyön tekeminen oli opettavainen projekti, joka opetti useita asioita ohjelmoinnista, sisältöjen asettelusta, rajapinnoista ja Bootstrapist aivan projektien suunnitteluun ja laajojen tietomäärien käsittelyyn asti. Opinnäytetyön tuoman kokemuksen ansioista tulevien projektien suunnittelu ja toteutus tulee olemaan askelleen helpompaa sekä erilaisten ohjelmistojen käyttö pitäisi onnistua varmemmin ottein. Opinnäytetyön kirjallisen osuuden

luomisen ansioista suurempien tietomäärien asettelu ja hallinta pitäisi onnistua huomattavasti helpommin kuin aiemmin.

LÄHTEET

Bootstrap Get Started. s.a. W3Schools. WWW-dokumentti. Saatavissa: https://www.w3schools.com/bootstrap/bootstrap_get_started.asp [viitattu 12.10.2017].

General Information. s.a. MySQL. WWW-dokumentti. Saatavissa: <https://dev.mysql.com/doc/workbench/en/wb-intro.html> [viitattu 17.10.2017].

Bootsrap. s.a. Get Bootstrap. WWW-dokumentti. Saatavissa: <http://get-bootstrap.com/> [viitattu 31.10.2017].

Creative. Start Bootstrap. s.a. WWW-dokumentti. Saatavissa: <https://start-bootstrap.com/template-overviews/creative/> [viitattu 2.11.2017].

cPanel. 2016. Tutorialspoint. E-kirja. Saatavissa: https://www.tutorialspoint.com/cpanel/cpanel_tutorial.pdf [viitattu 6.11.2017].

MySQL Workbench. 2017. Workbench Reference Manual. PDF-tiedosto. Saatavissa: <https://downloads.mysql.com/docs/workbench-en.pdf> [viitattu 7.11.2017].

Top Five Goals for Front End Developer. 2011. Red Frog Consulting. WWW-dokumentti. Saatavissa: <http://www.redfrogconsulting.com/top-five-goals-for-front-end-developers.php> [viitattu 15.11.2017].

History. Get Bootstrap. s.a. WWW-dokumentti. Saatavissa: <https://v4-alpha.getbootstrap.com/about/history/> [viitattu 16.11.2017].

What is cPanel. 2016. Why Should You use cPanel Web Hosting. Webhosting Media. WWW-dokumentti. Saatavissa: <https://webhostingmedia.net/what-is-cpanel/> [viitattu 16.11.2017].

About. s.a. fabFORCE. WWW-dokumentti. Saatavissa: <http://fabforce.eu/about.php> [viitattu 16.11.2017].

Sun Microsystems Introduces MySQL Workbench. 2008. Cision PrWeb. WWW-dokumentti. Saatavissa: <http://www.prweb.com/releases/mysql/workbench/prweb866414.htm> [viitattu 16.11.2017].

Python Scripting in Workbench. 2010. Mysql Workbench. WWW-dokumentti. Saatavissa: <http://mysqlworkbench.org/2009/03/python-scripting-in-workbench/> [viitattu 16.11.2017].

Codesido, I. 2009. What is front-end development. Blogi. Saatavissa: <https://www.theguardian.com/help/insideguardian/2009/sep/28/blogpost> [viitattu 16.11.2017].

In what programming language is based cpanel. 2011. Forum cPanel. WWW-sivustot. Saatavissa: <https://forums.cpanel.net/threads/in-what-programming-language-is-based-cpanel.248071/> [viitattu 16.11.2017].