

Marika Mykrä

ILMOITTAUTUMISJÄRJESTELMÄN TUOTANTOPROSESSI

Tietojenkäsittelyn koulutusohjelma

2019

ILMOITTAUTUMISJÄRJESTELMÄN TUOTANTOPROSESSI

Mykrä, Marika
Satakunnan ammattikorkeakoulu
Tietojenkäsittelyn koulutusohjelma
Joulukuu 2019
Sivumäärä: 64
Liitteitä: 1

Asiasanat: ohjelmistotuotanto, vaatimusmäärittely, Wordpress, prosessi

Aiheena tässä opinnäytetyössä oli tilaajan tarpeisiin räätälöidyn web-pohjaisen ilmoittautumis- ja kurssihallintasovelluksen toteutusprosessin tarkasteleminen sekä teorianäkökulmasta että käytännön toteutuksen näkökulmasta. Tässä työssä edellä mainitusta sovelluksesta käytetään termiä ilmoittautumisjärjestelmä.

Työn alussa tarkasteltiin ohjelmistokehitystä, sen vaiheita ja prosessimalleja yleisellä tasolla ja siitä näkökulmasta, miten niitä voidaan hyödyntää ilmoittautumisjärjestelmän suhteen.

Työn myöhemmissä osioissa perehdyttiin ilmoittautumisjärjestelmän tuotantoprosessin eri vaiheisiin: määrittelyyn, suunnitteluun ja toteutukseen. Pyrin työssä käsittelemään mahdollisimman kattavasti erilaisia tekijöitä, jotka vaikuttavat prosessin eri osaluokiin.

Projektin suunnittelu- ja toteutusvaiheessa minulla oli resursseja kehittää järjestelmästä vielä monipuolisempi, mitä määrittelyvaiheessa alussa määriteltiin. Lopputuloksena syntynyt tuote, valmis järjestelmä, oli varsin onnistunut ja vastasi sille asetettuja tavoitteita.

SOFTWARE PROCESS OF THE REGISTRATION SYSTEM

Mykrä, Marika

Satakunnan ammattikorkeakoulu, Satakunta University of Applied Sciences

Degree Programme in Business Information Systems

December 2019

Number of pages: 64

Appendices: 1

Keywords: software engineering, software requirements, Wordpress, process

The subject of this thesis was to examine the software process of a web-based registration and course management application tailored to the needs of the subscriber from both a theoretical and practical perspective. In this thesis, the term registration system is used for the above application.

At the beginning of the thesis, software development, its parts and process models were examined at a general level and how they can be utilized in the registration system.

Later parts of the thesis focused on the different parts of the registration system's software process: requirements, design and development. I tried to address as much as possible the various factors that affect different aspects of the process.

During the project design and development section, I had the resources to make the system even more versatile than what was initially defined in the requirements section. The final product, the finished system, was successful and met its goals.

SISÄLLYS

1	JOHDANTO.....	6
2	OHJELMISTOJEN TUOTANTOPROSESSIT YLEISESTI	8
2.1	Ohjelmistotuotannon vaiheet	9
2.1.1	Vaatimusmäärittely	9
2.1.2	Suunnittelu	10
2.1.3	Toteutus, testaus ja ylläpito	10
2.2	Ohjelmistotuotannon prosessimallit.....	11
2.2.1	Vesiputousmalli.....	11
2.2.2	Ketterät menetelmät.....	12
3	ILMOITTAUTUMISJÄRJESTELMÄN VAATIMUSMÄÄRITTELY	13
3.1	Yleiskuvaus.....	13
3.1.1	Vanhan järjestelmän toiminta ja sen haasteet.....	15
3.1.2	Tilaaajan kehitystoiveet uudelle järjestelmälle	17
3.1.3	Käyttötarkoitus	18
3.1.4	Käyttäjät ja toimintaympäristö	19
3.2	Toiminnalliset vaatimukset.....	19
3.2.1	Yleiskuvaus järjestelmän toiminnasta	19
3.2.2	Käyttötapauskaavio	22
3.2.3	Ei-toteutettavat toiminnot.....	22
3.3	Ei-toiminnalliset vaatimukset	23
3.3.1	Sisällönhallintajärjestelmän valitseminen	23
3.3.2	Käytettävyys	24
3.3.3	Tietoturva ja toimintavarmuus	24
3.3.4	Ylläpidettävyys ja huollettavuus	25
3.3.5	Laajennettavuus.....	25
3.4	Muut vaatimukset.....	26
3.4.1	Suorituskyky	26
3.4.2	Selaimet	26
3.4.3	Käyttöliittymät.....	27
4	SUUNNITTELU	28
4.1	Ilmoittautumisjärjestelmän arkkitehtuuri.....	28
4.2	Sivurakenne.....	28
4.3	Sisältötyypit	28
4.4	Toiminnallisuuksien suunnittelu	32
4.4.1	Kurssihallinta	32

4.4.2	Jäsen- ja viitenumerotietojen päivittäminen	33
4.4.3	Ilmoittautuminen	34
4.4.4	Ilmoittautumistietojen käsittely	35
4.5	Tietoturva	35
4.6	Ulkoasu ja käyttöliittymä	36
5	TOTEUTUS	39
5.1	Wordpress alustana	39
5.1.1	Teemat	39
5.1.2	Lisäosat	40
5.2	Sisältörakenteiden luominen	41
5.2.1	Sivut	41
5.2.2	Lohkot	41
5.2.3	Sisältötyypit	43
5.2.4	Kentät	44
5.3	Toiminnallisuuksien toteuttaminen.....	45
5.3.1	Hallintapuolen toiminnallisuudet	45
5.3.2	Kurssilistaus	47
5.3.3	Ilmoittautuminen	49
5.4	Tietokannat	52
5.5	Käyttöliittymät ja ulkoasu.....	53
5.5.1	Hallintapuolen näkymät	54
5.5.2	Julkisen puolen näkymät	54
5.5.3	Responsiivisuus	59
5.6	Testaus	60
5.7	Käyttöönotto	60
6	TULOSTEN ARVIOINTI.....	61
6.1	Loppukäyttäjien kommentit	61
6.2	Projektinhallinta	61
7	LOPPUSANAT	62
	LÄHTEET	63

1 JOHDANTO

Jokaisella suuremmalla projektilla tulisi aina olla suunnitelma, vaikka kaikki projektit eivät aina noudatakaan suunnitelmia kovin tarkasti. Systemaattisella tekemisellä ja suunnitelmallisuudella, tuotantoprosessilla, tähdätään ohjelmistotuotannossa laadukkaaseen lopputulokseen.

Tämä opinnäytetyö käsittelee yksilöllisiin tarpeisiin luodun ilmoittautumisen ja kurssihallinnan kattavan websovelluksen tuotantoprosessia, jonka lopullisena tuotteena syntyi valmis sovellus. Käytän tässä työssä edellä mainitusta sovelluksesta termiä ilmoittautumisjärjestelmä. Projektissa tarkastellaan järjestelmän tuotantoprosessia sen eri vaiheissa – määrittelystä alkaen aina käyttöönottoon ja loppukäyttäjien kommentteihin asti.

Työn tilaaja on porilainen koiraharrastusseura TVA Pori ry, jonka kurssi-ilmoittautumisiin ja -hallintaan tämän projektin myötä valmistunutta järjestelmää tullaan käyttämään. Heillä on ollut käytössä lähes kymmenen vuotta edellinen järjestelmä, mutta vuosien kuluessa, kurssimäärien ja ilmoittautumisten lisääntyessä sekä teknisen kehityksen mennessä eteenpäin oli tullut tarve uudistaa ilmoittautumisjärjestelmä vastaamaan paremmin harrastusseuran tarpeita. Vanhan järjestelmän päivittäminen ei ollut mahdollista ja sen vuoksi täytyi löytää uusi ratkaisu ilmoittautumis- ja kurssihallinnalle.

Vaikka nykyaikana valmiita ilmoittautumissovelluksia erilaisiin tarpeisiin on saatavilla runsas kirjo, ei TVA Pori ry:n hieman normaalista poikkeavaan ilmoittautumismalliin ja jäsenyystarkastuksiin tuntunut löytyvän sopivaa. Tämän myötä päädyttiin toteuttamaan oma, juuri TVA Pori ry:n tarpeisiin suunniteltu järjestelmä, mikä sopi loistavasti opinnäytetyöni aiheeksi.

Tämän opinnäytetyön ensimmäisessä osiossa perehdytään yleisellä tasolla ohjelmistojen tuotantoprosessiin ja sen malleihin. Huolellinen ohjelmistosuunnittelu varmistaa, että lopullinen sovellus on tehokas, käyttökelpoinen ja ylläpidettävä sekä pysyy tavoitteissaan aikataulun ja budjetin suhteen (Stephens 2015, xxiii). Ohjelmistojen

huolellinen määrittely ja suunnittelu ovat siis avainasemassa tavoiteltaessa mahdollisimman laadukasta lopputulosta.

Opinnäytetyön jälkimmäisissä osioissa käydään läpi ilmoittautumisjärjestelmän tuotantoprosessin eri vaiheita peilaten ensimmäisen osan teoriapohjaan ja esiteltyihin käytäntöihin ja malleihin. Määrittely-, suunnittelu- ja toteutusvaiheet on kuvattu omissa osioissaan, vaikka käytännön työssä suunnittelu ja toteutus ovatkin kulkeneet hieman rinnakkain.

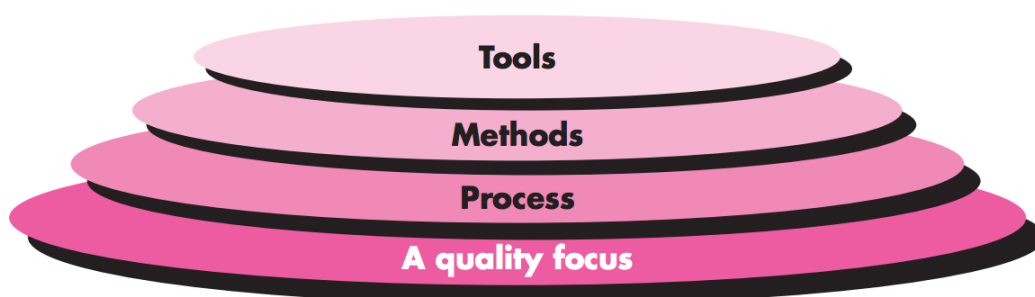
Järjestelmän toteutuksesta muodostui lopulta hieman laajempi ja monipuolisempi kuin alussa oli tarkoitus. Opinnäytetyön viimeisissä osioissa on tarkasteltu ilmoittautumisjärjestelmän tuotantoprosessin onnistumista ja lopputulosta sekä toteuttajan että loppukäyttäjän näkökulmasta ja pohdittu järjestelmän lisäkehitysmahdollisuuksia tulevaisuudessa.

2 OHJELMISTOJEN TUOTANTOPROSESSIT YLEISESTI

Pienempiä sovelluksia ja ohjelmia voi tehdä ilman sen suurempaa suunnittelua, mutta varsinkin suurempien ohjelmistojen tuotanto edellyttää systemaattisuutta prosessiin. Ohjelmistotuotanto tai –kehitys (software engineering) määritellään yleensä tarkoittamaan kaikkia ohjelmiston valmistuksen menetelmiä sekä prosessinhallintaa (Luukkainen 2011).

Ohjelmoijat eivät välttämättä ymmärrä, mitä käyttäjät haluavat tai tarvitsevat, joten ilman huolellista ohjelmistotuotantoprosessia on vaarana, että yksinkertainenkin projekti voi mennä pieleen. Ohjelmistotuotanto sisältää vaiheet ohjelmiston ideasta valmiiksi sovellukseksi ja varmistaa sen, että lopullinen sovellus on tehokas, käyttökelpoinen ja ylläpidettävä. Rod Stephens (2015, xxiii) kiteyttää ohjelmistotuotannon ohjelmoijan näkökulmasta erittäin hyvin: ”In short, software engineering lets you control what otherwise might seem like a random whirlwind of chaos.”

Ohjelmistokehityksen avulla voidaan rakentaa monimutkaisiakin järjestelmiä laadukkaasti. Se voidaan ajatella sisältävän kerroksittain tekniikoita. Kuvassa 1 on Pressmanin (2010, 14) mallintamana ohjelmistosuunnittelun kerrokset. Siinä kaikki pohjautuu laatuun ja ohjelmistoprosessi muodostaa perustan ohjelmistoprojektin hallinnalle.



Kuva 1. Ohjelmistosuunnittelun kerrokset (Pressman 2010, 14)

Ohjelmistosuunnittelumenetelmät tarjoavat teknisen ohjeen ohjelmistojen rakentamiseen. Se ei kuitenkaan ole jäykkä määräys siitä, miten ohjelmisto tehdään, vaan se on ennemminkin mukautuva lähestymistapa, jonka avulla toteuttajat voivat valita sopivat tavat ja tehtävät. (Pressman 2010, 14–15)

2.1 Ohjelmistotuotannon vaiheet

Tyypillisesti ohjelmistotuotannon prosessi jaetaan viiteen vaiheeseen: vaatimusmäärittely, suunnittelu, toteutus, testaus ja ylläpito (Luukkainen & Laine 2010, 1).

2.1.1 Vaatimusmäärittely

Tietojärjestelmien määrittelyvaiheessa on tärkeää selvittää tavoitteet ja käyttäjien toiveet, määrittellä, mitä ongelmaa ratkaistaan sekä määrittellä ja mallintaa tavoitetila mahdollisimman hyvin. Suunnittelijoiden ja asiakkaiden välinen kommunikaatio on myös suuressa merkityksessä, sillä sovelluksien sisällöt ovat suurelta osin asiakkaiden tiedon ja tietämyksen varassa. (Nykänen 2014, 23–27)

Vaatimusmäärittelyllä pyritään siis vastaamaan kysymykseen *mitä halutaan kehittää*, puuttumatta siihen, miten tekninen kehitys tapahtuu. Vaatimusmäärittelyssä pitää ottaa huomioon myös toimintaympäristön ja toteutusteknologian järjestelmälle asettamat rajoitteet (Luukkainen 2011).

Määrittelyssä voidaan kuvata toiminnallisia vaatimuksia, mitä järjestelmän tulisi toteuttaa, esimerkiksi “järjestelmään voi lisätä uusia kursseja” ja “järjestelmän tulee tarkastaa, onko kurssilla tilaa”. Määrittelyssä pitää ottaa huomioon myös rajoitteet, esimerkiksi “sovelluksen käyttö tapahtuu www-selaimella” ja “sisällönhallintajärjestelmänä on Wordpress”.

Toiminnallisia vaatimuksia on hyvä analysoida tarkemmin, jotta suunnittelu- ja toteutusvaihe on mahdollisimman selkeä. Tässä tarkemmassa analysoinnissa hyvä apu on käyttää erilaisia mallinnuksia, jotka auttavat hahmottamaan esimerkiksi asioiden keskinäisiä suhteita. Yleisimmin käytetty ohjelmistojen mallinnusmenetelmä on kansainvälisesti standardoitu UML (Unified modeling language), jota kehittää ja ylläpitää Object Management Group eli OMG (Lano 2009, 13). Määrittelyvaiheen mallinnuksessa käytännöllisiä ovat esimerkiksi käyttötapauskaaviot, joissa kuvataan käyttötapausten ja käyttäjien keskinäisiä suhteita.

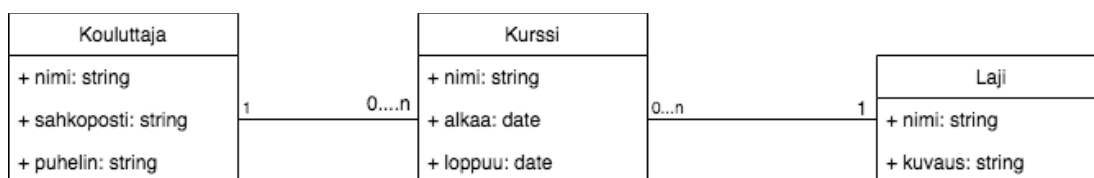
2.1.2 Suunnittelu

Ohjelmiston suunnitteluvaiheessa mietitään, miten vaatimusmäärittelyvaiheen määritykset saadaan toteutettua. Tyypillisesti suunnitteluvaihe jaetaan pienempiin osiin, jotka suunnitellaan erikseen.

Arkkitehtuurisuunnittelussa suunnitellaan, miten järjestelmä jakautuu korkean tason rakenneosiin. Esimerkiksi tietokanta, sovellustiedostot ja käyttäjän www-selain ovat rakenneosia. (Luukkainen & Laine 2010, 2)

Järjestelmän toiminnallisuudet voidaan tarvittaessa jakaa komponentteihin, jotka suunnitellaan erikseen. Tällöin komponenttien välille määritellään rajapinnat yhdistämään komponentit. Komponenttien suunnittelu erillisinä auttaa myös siinä, että yksittäinen komponentti voidaan antaa yksittäisen ohjelmoijan vastuulle.

UML-kaavioita käytetään apuna myös suunnitteluvaiheessa. Suunnitteluvaiheen mallinnuksessa käytetään tarkemman tason mallinnusmenetelmiä, esimerkiksi luokkakaavioita. Kuvassa 2 on esitetty yksinkertainen esimerkki luokkakaaviosta, josta selviää luokkien väliset suhteet sekä muuttujat.



Kuva 2. Esimerkki UML-kaaviosta suunnitteluvaiheessa

2.1.3 Toteutus, testaus ja ylläpito

Toteutusvaiheessa järjestelmä toteutetaan suunnitteluvaiheen suunnitelmien mukaisesti. Toteutus ja suunnittelu ovat kuitenkin paljon yhteydessä toisiinsa, sillä toteutusvaiheen huomioiden perusteella saatetaan joutua muuttamaan vielä suunnitelmia.

Testausta tehdään sekä toteutuksen aikana että toteutuksen jälkeen. Yksittäiset metodit testataan yksikkötestauksessa (unit testing) ja yhdistettäessä erilaiset alikomponentit

ja metodit testataan näiden yhteensopivuus integrointitestauksessa (integration testing). Yksikkötestaus ja integrointitestaus kulkevat toteutuksen rinnalla ja komponenttien ohjelmoijat suorittavat ne. Kun kaikki järjestelmän komponentit on saatu yhdistettyä, testataan järjestelmää kokonaisuutena järjestelmätestauksessa (system testing). (Luukkainen & Laine 2010, 3)

Ohjelman valmistumisen jälkeen alkaa ylläpitovaihe, jossa korjataan käyttövaiheessa ilmenneitä virheitä ja tarvittaessa täydennetään ohjelmaa tai muokataan sitä.

2.2 Ohjelmistotuotannon prosessimallit

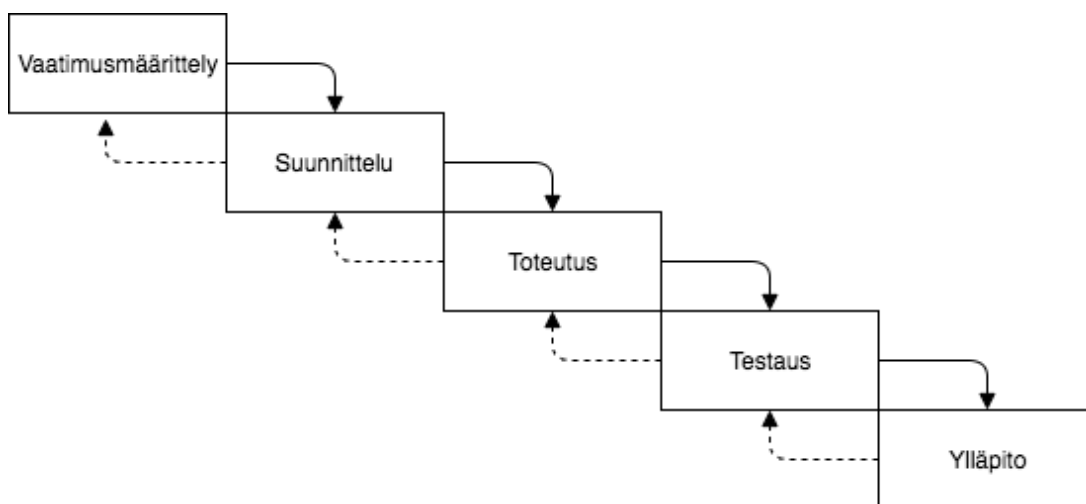
Ohjelmistotuotannon prosessimallit ovat eräänlaisia etenemissuunnitelmia projektille ja luovat hallintaa ja organisointia prosessiin. Seuraavissa kappaleissa käydään kaksi toisistaan poikkeavaa prosessimallia läpi. Näiden kahden prosessimallin yhdistelmää on sovellettu käytettäväksi tässä ilmoittautumisjärjestelmän tuotantoprojektissa.

2.2.1 Vesiputousmalli

Vesiputousmallissa (waterfall model) ohjelmistokehitys tapahtuu vaiheittain ja seuraava vaihe alkaa vasta edellisen loputtua. Tässä mallissa edellinen vaihe tehdään valmiiksi ja dokumentoidaan, jonka jälkeen vasta siirrytään suunnitteluvaiheeseen. (Luukkainen & Laine 2010, 3)

Vesiputousmalli on ollut hyvin perinteinen tapa tuottaa ohjelmistoja, mutta sen käyttöön liittyy myös paljon ongelmia: Mikäli projektissa on vaatimusten suhteen epävarmuustekijöitä, on vaikea soveltaa mallia. Muutosten tuominen myöhemmissä vaiheissa on hankalaa, samoin edellisten vaiheiden virheiden korjaaminen. (Thinking Portfolio 2016)

Vesiputousmalli soveltuu hyvin pienempiin projekteihin, joissa vaatimukset ovat selviä. Kuvassa 3 on esitetty vesiputousmallin periaate, jossa seuraava vaihe tehdään edellisen pohjalta.



Kuva 3. Vesiputousmalli ohjelmistokehityksessä (Thinking Portfolio 2016).

2.2.2 Ketterät menetelmät

Viimeisen vuosikymmenen aikana ohjelmistotuotannossa on yleistynyt ketterät mallit (agile), joissa edetään iteraatioissa eli pienissä paloissa. Ketteriä menetelmiä on useita, mutta tunnetuimmat lienevät XP eli Extreme Programming ja Scrum. Näissä yhteistä on se, että vaatimukset tarkentuvat projektin edetessä, projektit ovat joustavia ja kommunikaation ja yhteistyön merkitys on suuri.

Tässä mallissa alussa järjestelmä määritellään pääpiirteissään ja suunnitellaan korkealla tasolla (arkkitehtuurisuunnittelu). Tämän jälkeen kehitys jatkuu sykleissä, eli pienissä palasissa, jotka jokainen määritellään, suunnitellaan ja toteutetaan erikseen.

Vuonna 2001 julkaistiin ketterän ohjelmistokehityksen julistus, joka kiteyttää menetelmän ajatukset ja painottaa mm. yksilöitä ja kanssakäymistä, toimivaa ohjelmistoa, asiakasyhteistyötä ja vastaamista muutokseen enemmän kuin pitäytymistä suunnitelmassa (Manifesto for Agile Software Development 2001).

3 ILMOITTAUTUMISJÄRJESTELMÄN VAATIMUSMÄÄRITTELY

Ilmoittautumisjärjestelmän tuotantoprosessi lähti liikkeelle vaatimusmäärittelystä. Tiilaajan kanssa käytiin paljon keskusteluja niin vanhan ilmoittautumisjärjestelmän toiminnan hyvistä ja huonoista puolista kuin toiveista, rajoituksista ja kehitysideoista uuteen järjestelmään. Näitä analysoitiin tarkemmin ja koottiin järjestelmälle vaatimusmäärittely.

Heti projektin alussa ei ollut kaikki järjestelmän vaatimukset selvillä, vaan niitä muokattiin ja tarkennettiin projektin kuluessa. Projektissa käytettiin osittain vesiputousmallia, mutta osittain ketterää ohjelmistokehitystä, sillä osassa järjestelmän vaiheiden toteuttamista palattiin vaatimusmäärittelyyn ja johdettiin vaatimusmäärittelyyn muutoksia toteutusehdotuksen pohjalta.

3.1 Yleiskuvaus

TVA Pori ry toimii Porin Noormarkussa ja järjestää monipuolisesti eri koiraharrastuslajien kursseja ja koulutuksia läpi vuoden. Viikoittaisia kurssipaikkoja seuralla on keskimäärin 160 kpl. Kurseille on paljon kysyntää ja suosituimpien kurssien kurssipaikat menevät ilmoittautumisessa jopa ensimmäisen minuutin aikana. Tasapuolisen kurssipaikkajaon takaamiseksi on tärkeää, että ilmoittautumisjärjestelmä toimii moitteettomasti ja näin helpottaa myös yhdistyksen kurssivastaavien työmäärää.

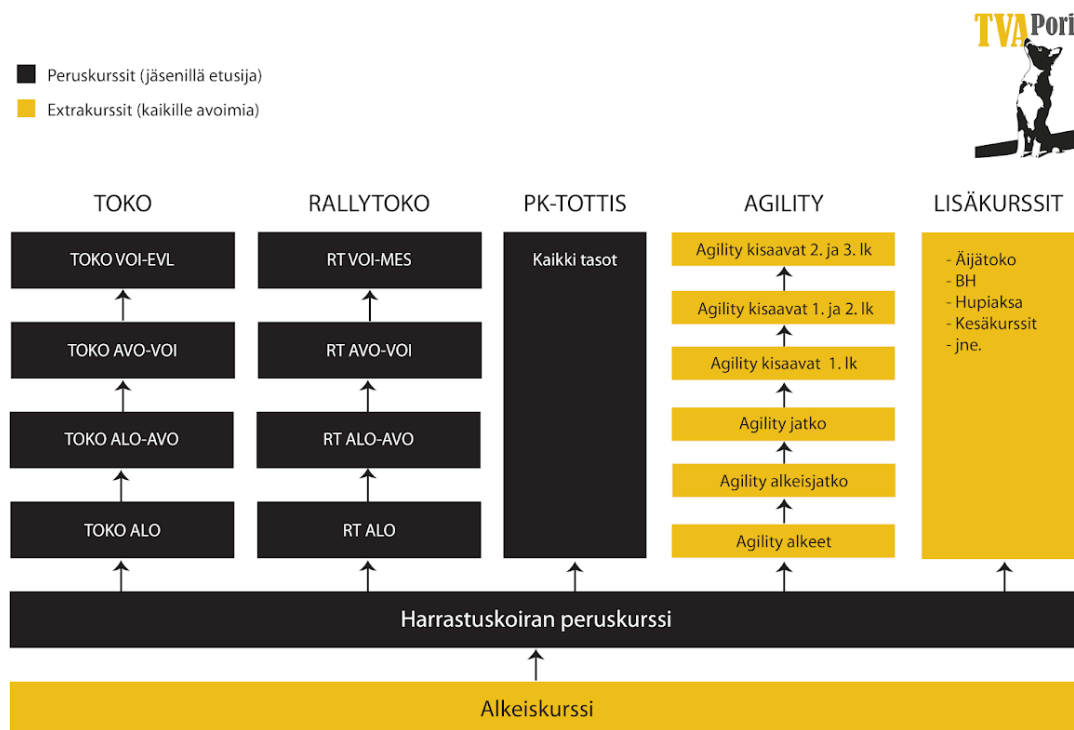
TVA Pori ry:llä on ollut käytössä ilmoittautumisjärjestelmä, joka on räätälöity aikanaan seuran tarpeisiin. Vuosien kuluessa ja tekniikan kehittyessä on tullut tarve uudistaa järjestelmä vastaamaan paremmin seuran tarpeita sekä päivittää järjestelmä käyttämään nykyaikaisempia tekniikoita.

Ilmoittautuminen yhdistyksen järjestämille kurseille tapahtuu jaksoittain viisi (5) kertaa vuodessa. Valittavia kursseja ilmoittautumisessa on kerralla noin 20–40, riippuen ilmoittautumisjaksosta. Kurssien ilmoittautumisajankohta on määritelty alkamaan tiettyyn päivään ja kellonaikaan, joten ilmoittautumishetkellä sivustolla on yleensä paljon

liikennettä ja ilmoittautumisen alkaessa tulee runsaasti kurssi-ilmoittautumisia samanaikaisesti.

TVA Pori ry tarjoaa osalle kursseista jäsenille etuoikeuden ilmoittautua. Käytännössä tämä tarkoittaa sitä, että tietyille kursseille on määrätty jäsenille oma ilmoittautumisaika, jonka jälkeen vasta ilmoittautuminen aukeaa kaikille avoimena. Osalle kursseista ei ole merkitystä sillä, onko ilmoittautuja yhdistyksen jäsen vai ei. Ilmoittautumisjärjestelmän yksi tärkeä tehtävä on tarkastaa ilmoittautujan jäsenyys niiden kursien osalta, missä se vaaditaan.

Kuvassa 4 on esitetty TVA Pori ry:n kurssirakenne. Kuvassa keltaisella pohjalla merkityille kursseille voi ilmoittautua kuka tahansa jäsenyydestä riippumatta. Mustalla pohjalla merkityille kursseille on ensimmäisenä määriteltynä ajankohtana mahdollista ilmoittautua vain harrastusseuran jäsenyyden omaavien henkilöiden ja myöhemmin, mikäli kurssipaikkoja on jäänyt, alkaa kaikille avoin ilmoittautuminen jäsenyydestä riippumatta.



Kuva 4. TVA Pori ry:n kurssirakenne (TVA Pori ry:n www-sivut 2019)

Kurssien osallistujamäärä on rajoitettu kurssikohtaisesti ja kurssille hyväksytään osallistujat ilmoittautumisjärjestyksessä. Suosituimmat kurssit täyttyvät jopa ensimmäisen minuutin aikana ilmoittautumisen alkamisesta. Ennen varsinaisen ilmoittautumisen alkamista on määrättyinä aikana kouluttajilla ja hallituksen jäsenillä mahdollisuus ilmoittautua etukäteen kursseille.

3.1.1 Vanhan järjestelmän toiminta ja sen haasteet

Vaikka toteutusajankohtanaan 2010-luvun alussa vanha ilmoittautumisjärjestelmä on vastannut tarpeita hyvin, on vuosien varrella tullut esiin kehitys- ja uudistamistarpeita.

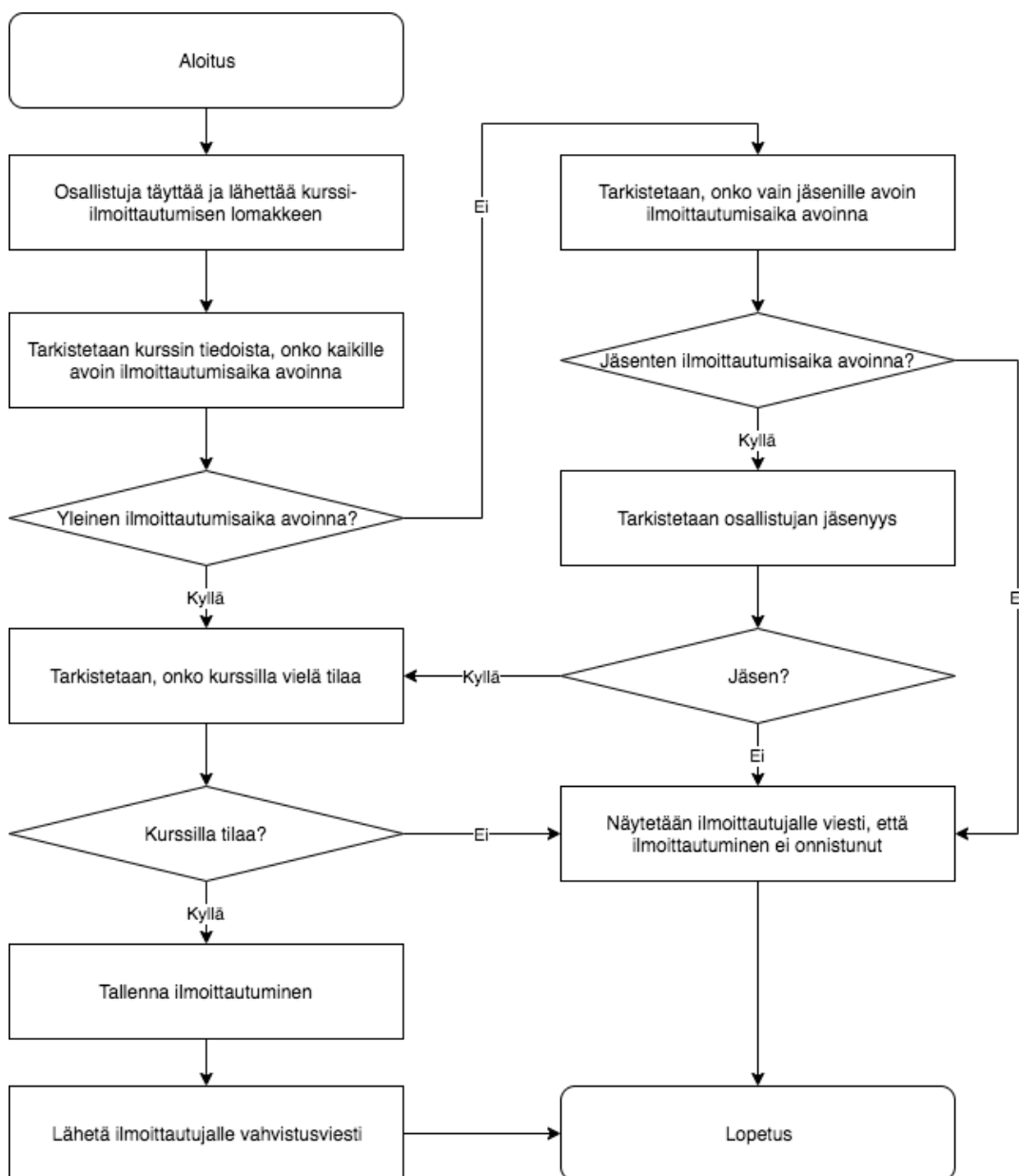
Vanhassa ilmoittautumisjärjestelmässä jäsenyyden tarkastamiseen on liittynyt haasteita. Jäsennumerot on siirretty manuaalisesti erillisestä jäsenkirjanpidon taulukosta ilmoittautumisjärjestelmän tietoihin rivi kerrallaan, ja samoin jäsenet, joiden jäsenyys on päättynyt, on poistettu järjestelmästä manuaalisesti. Yksitellen tehtävässä manuaalisessa jäsennumeroiden syötössä ja poistossa on suuri inhimillisen virheen riski. Tähän liittyviä virheitä on välillä sattunut ja niiden johdosta on päässyt käymään tilanteita, joissa ilmoittautumisjärjestelmä ei ole hyväksynyt ilmoittautuneen jäsennumeroa ja kurssi-ilmoittautuminen ei ole onnistunut, vaikka ilmoittautujan jäsenyys on ollut voimassa. Näissä tilanteissa osallistuja on saattanut jäädä vallaan ilman kurssipaikkaa.

Vanhassa järjestelmässä ilmoittautumisen kurssivalintalistassa ei erotu mitenkään täynnä olevat kurssit vapaita paikkoja sisältävistä kursseista. Toisin sanoen vanha järjestelmä sallii ilmoittautumisen yrittämisen jo täynnä olevalle kurssille, mikä tuo turhaa työtä ilmoittautujalle ja saattaa sen vuoksi jäädä toisen kurssin kurssipaikka saamatta. Ilmoittautumisia valvova henkilö on joutunut tiedottamaan kurssien täyttymisen tilanteesta sosiaalisen median kautta, jotta osallistujat eivät turhaan laittaisi ilmoittautumisia jo täynnä oleville kursseille.

Vanhan järjestelmän toiminnassa on huomattu myös virhetilanteita, joiden vuoksi on ilmoittautumisen yhteydessä kurssille rekisteröity väärä osallistuja. Tämä virhe on paikallistettu vanhojen kurssien tietoja poistavaan osioon, jossa uudet kurssit saattavat olla tietokannassa samoilla id-numeroilla kuin vanhat jo poistetut kurssit, jolloin

tietokannassa on lisätyille kurssille valmiina olemassa vääriä osallistujia. Tämä on aiheuttanut harmillisia tilanteita ja vaatinut ilmoittautumistietokantataulun manuaalista tyhjentämistä.

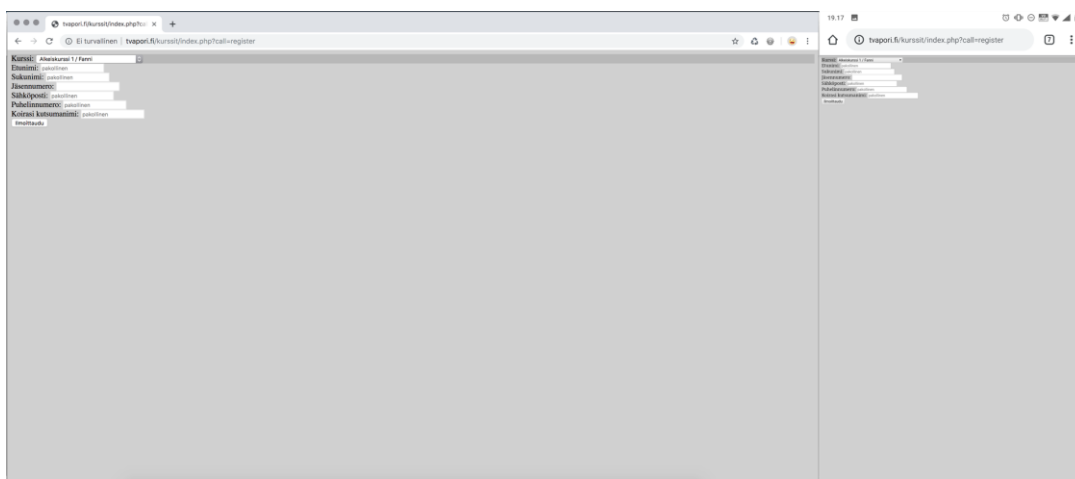
Kuviossa 5 on esitetty vuokaaviona vanhan järjestelmän ilmoittautumisprosessin eri vaiheet. Ilmoittautumisen tarkastamisessa ilmoittautumistilanteen perustoimintoja ovat ilmoittautumisajan, jäsenmäärän ja vapaiden kurssipaikkojen tarkastus.



Kuva 5. Vanhan ilmoittautumisjärjestelmän toimintamalli

Vanha järjestelmä on toteutusajankohtanaan vastannut sen hetken käyttöliittymätottumuksia, mutta tekniikan ja mobiililaitteiden aikakauden kehittyessä se ei enää ole käyttöliittymältään nykyajan tottumuksia ja käytettävyyttä vastaava, sillä se ei skaalaudu eri käyttölaitteille. Näin ollen mobiililaitteilla on ollut hankala ilmoittautua kursseille.

Kuvassa 6 on esitetty vanhan ilmoittautumisjärjestelmän näkymät kannettavan tietokoneen 1675 pikseliä leveässä selainikkunassa ja Nokia 8 -matkapuhelimen selainikkunassa. Molemmissa laitteissa näkymä on kuvattu käyttäen Chrome-selainta. Lomake jää todella pieneksi vasempaan yläkulmaan molemmilla laitteilla. Tämä havainnollistaa sitä, miten vuosikymmenen aikana käytettävien laitteiden näyttöjen koot ovat suurentuneet ja tarkentuneet. Vanhan ilmoittautumisjärjestelmän toteutusajankohtana pöytäkoneiden resoluutiot olivat paljon pienempiä ja mobiililaitteiden käyttö ilmoittautumisissa vähäisempää.



Kuva 6. Vanhan ilmoittautumisjärjestelmän käyttöliittymän skaalautuvuushaaste

3.1.2 Tilaajan kehitystoiveet uudelle järjestelmälle

Uudelle järjestelmälle toivottiin parannusta siihen, että ilmoittautuja näkee jo kurssia valitessaan, mitkä kurssit ovat jo täynnä, jotta ei turhaan laita ilmoittautumista täynnä olevalle kurssille ja näin pääsee nopeammin kilpailemaan muiden kurssien kurssipaikoista. Vanhassa järjestelmässä kurssin tullessa täyteen kurssi oli edelleen valittavana ilmoittautumislistassa ja sille pystyi laittamaan ilmoittautumisen. Niille ilmoittautujille, jotka eivät mahtuneet kurssille, tuli ilmoittautumisen jälkeen viesti, että kurssi on täynnä ja ilmoittautumista ei ole hyväksytty.

Yksi tilaajan toiveista oli löytää käytännöllisempi ratkaisu jäsenyyden tarkastamiseen ja jäsennumeroiden siirtämiseksi järjestelmään, jotta inhimillisen virheen mahdollisuus pienenee. Jäsennumeroiden hallintaan toivottiin kerralla päivitettävää listausta, jossa koko listauksen voi viedä kerralla eikä tarvitse listaa tarkistaa manuaalisesti riveittäin.

Uudelta järjestelmältä toivottiin selkeää ja käytännöllistä käyttöliittymää, joka sisältää myös ohjeistusta kurssi-ilmoittautumiseen. Kurssit on hyvä myös kategorisoida harrastuslajeittain. Ilmoittautumisjärjestelmän tulee mukautua käytännöllisesti eri mobiililaitteille.

Järjestelmän tietoturva haluttiin myös parantaa ja varmistaa, että jäsenten tiedot ovat varmasti turvassa. Järjestelmästä täytyisi olla myös helppo poistaa vanhoja kurssitietoja ja järjestelmälle täytyisi määritellä ajantasainen tietosuojaseloste.

Uutena ominaisuutena helpottamaan kurssivastaavien työtä oli toive saada laskutuksen viitenumerot valmiiksi järjestelmästä haettavaan ilmoittautumislistauksiin. Näiden lisääminen käsin viitenumero kerrallaan on vanhassa järjestelmässä ollut aikaa vievä ja manuaalinen vaihe.

3.1.3 Käyttötarkoitus

Tiivistetysti ilmoittautumisjärjestelmän käyttötarkoitus on ottaa ilmoittautumisia vastaan harrastusseuran järjestämille kurseille ja näin pyrkiä jakamaan käytössä olevat kurssipaikat tasapuolisesti. Tarkoitus järjestelmän käyttämisellä on helpottaa mahdollisimman paljon kurssivastaavien työtä.

Ilmoittautumisjärjestelmää on uudistuksessa tarkoitus myös laajentaa kattamaan kursien tiedot, kurssiselosteet sekä kouluttajien tiedot. Nämä eivät ole järjestelmän pääasiallinen tehtävä, mutta järjestelmä pyritään suunnittelemaan ja toteuttamaan niin, että ilmoittautujan on helppo tarkastaa kurssitarjonta myös ilmoittautumisjärjestelmän yhteydestä.

Ilmoittautumisessa järjestelmän pitää mahdollistaa tarvittaessa ilmoittautujan jäsenyyden tarkistaminen. Hallintapuolen näkökulmasta kurssien lisääminen, asetusten muokaus sekä osallistujalistojen tulostaminen pitää olla helposti käytettäviä. Järjestelmän tulee myös huolehtia viitenumeroiden lisäämisestä sellaisiin ilmoittautumisiin, joihin ne on tarve lisätä.

3.1.4 Käyttäjät ja toimintaympäristö

Ilmoittautumisjärjestelmän uudistamisessa tavoitteena on luoda uusi, vanhasta järjestelmästä erillinen web-selaimella toimiva sivusto, jota pystyy käyttämään www-selaimen avulla laitteesta riippumatta. Myös hallintapuolen tulee skaalautua käytettävästi mobiililaitteille, vaikka niiden pääasiallinen käyttö tapahtuu luultavimmin suurempien näyttöjen kautta.

Järjestelmän käyttäjäryhmät ovat ilmoittautijat, jotka ilmoittautuvat kursseille järjestelmän avulla sekä kurssivastaavat, jotka lisäävät kursseja hallinnan kautta ja hyödyntävät järjestelmän koostamia ilmoittautumistietoja. Kurssivastaavat voivat olla myös itse ilmoittautujia.

3.2 Toiminnalliset vaatimukset

3.2.1 Yleiskuvaus järjestelmän toiminnasta

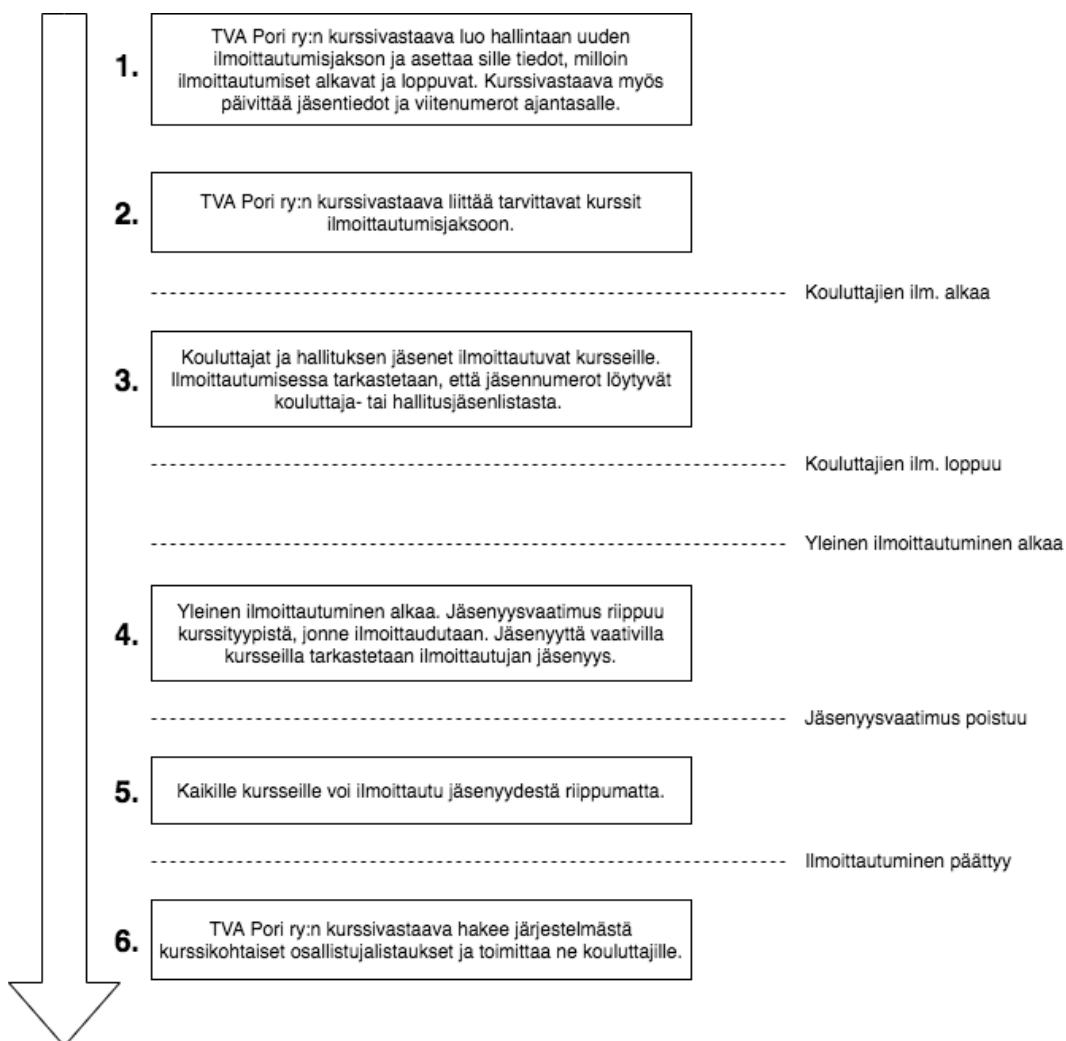
Ilmoittautumisjärjestelmän toimintamallin suunnittelussa lähdettiin liikkeelle pohtimalla, mitkä ovat järjestelmän toimintoja ja tehtäviä, jonka jälkeen mietittiin, missä järjestyksessä toiminnot tapahtuvat ja mitkä toiminnot ovat riippuvaisia toisistaan.

Harrastusseuralla on tapana antaa seuran kouluttajille sekä hallituksen jäsenille etukäteismahdollisuus ilmoittautua kursseille ennen kaikille jäsenille avointa ilmoittautumisajankohtaa. Uudessa järjestelmässä oli toiveena, että ilmoittautuminen voidaan avata tiettyä ajankohtana vain kouluttajille ja hallituksen jäsenille. Käytännössä tämä

tarkoittaa sitä, että tällöin vain kouluttajien ja hallituksen jäsenten jäsennumeroilla pääsee ilmoittautumaan kursseille.

Kurssit, joiden ilmoittautuminen on ollut vain jäsenille auki, on tarve muuttaa tietyn ajankohdan jälkeen avoimeksi kaikille ilmoittautua jäsenyydestä riippumatta. Tällöin kurssille täytyy määritellä ajankohta, jonka jälkeen kurssille ilmoittautumiseen ei vaadita enää jäsennumeroa.

Kuvassa 7 on esitetty suunniteltu tapahtumien järjestys ilmoittautumisjaksolla. Kuvasta selviää, missä järjestyksessä asetukset ilmoittautumisajoista vaikuttavat. Ilmoittautumisjärjestelmän tapahtumat voidaan pilkkoa kuvion mukaisiin osiin ja määritellä ne erikseen.



Kuva 7. Ilmoittautumisjakson tapahtumien järjestys

Kuvan 7 kaavion ensimmäisessä kohdassa kurssivastaava luo ilmoittautumisjakson ja sen asetukset. Asetuksista selviää ilmoittautumisvaiheiden alkamis- ja loppumisajat. Tässä vaiheessa kurssivastaavan täytyy myös päivittää jäsentiedot sekä kouluttaja- ja hallitusjäsenlistaukset. Riippuen viitenumeroiden suunnitellusta toteutustavasta, tulee kurssivastaavan myös päivittää viitenumerolistat niin, että viitenumeroita riittää liitettäväksi ilmoittautumisiin.

Toisessa vaiheessa kurssivastaava luo tai liittää suunnitellusta toteutustavasta riippuen kurssit ilmoittautumisjaksoon. Kurseista pitää selvittää vähintään niiden tyyppi ja suurin sallittu osallistujamäärä. Kurssien tyyppillä tarkoitetaan tässä perus- ja ekstrakurssi-valintaa, eli onko kurssit tarkoitettu ensisijaisesti jäsenille.

Ensimmäisenä ilmoittautumisaikana, kuvan 7 kolmannessa vaiheessa, voi kursseille ilmoittautua vain kouluttajien ja hallituksen jäsenten jäsennumeroilla. Kouluttajien ilmoittautumisiin ei liitetä viitenumeroa toisin kuin hallitusjäsenten ilmoittautumisiin. Järjestelmän täytyy pitää huoli siitä, että kurssien osallistujamäärä pysyy asetetun rajan sisällä.

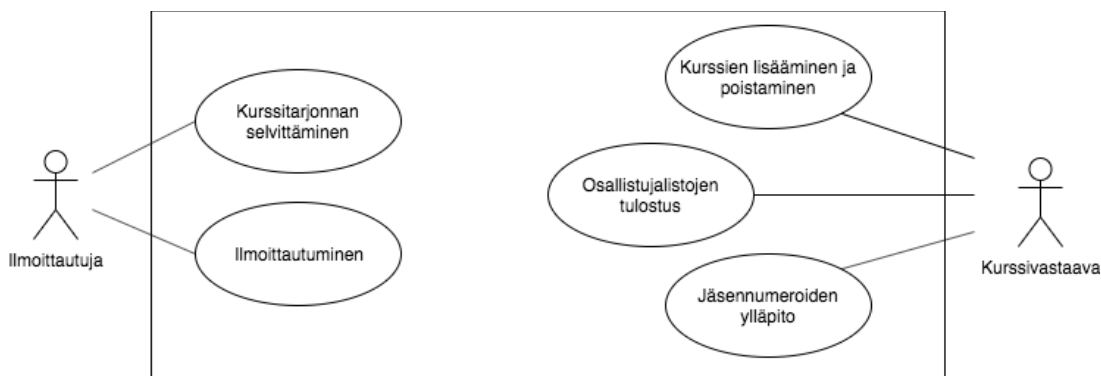
Varsinaisella ilmoittautumisajalla, kuvan neljännessä vaiheessa aukeaa varsinainen ilmoittautumisaika. Tällöin ekstrakursseille voi ilmoittautua kuka tahansa jäsenyydestä riippumatta, mutta peruskursseille vaaditaan seuran jäsenyys. Järjestelmän pitää tarkastaa jäsenyys peruskurssien osalta.

Viidennessä vaiheessa jäsenten etuoikeus peruskurssien ilmoittautumiseen poistuu ja vapaille jäljelle jääneille kurssipaikoille voi ilmoittautua kuka tahansa. Viimeisessä vaiheessa ilmoittautumisen loputtua ilmoittautumistiedot haetaan järjestelmästä ja toimitetaan kouluttajille.

Kaikissa ilmoittautumisen vaiheissa järjestelmän on pidettävä huolta siitä, ettei kurssin maksimiosallistujamäärä ylity ja täynnä olevat kurssit merkitään täysiksi. Ilmoittautumisen yhteydessä järjestelmä lisää ilmoittautumistietoon viitenumeron, mikäli kyseessä ei ole kouluttajan ilmoittautuminen.

3.2.2 Käyttötapauskaavio

Kuvassa 8 on mallinnettu UML-standardin mukaisesti ilmoittautumisjärjestelmän korkean tason käyttötapauskaavio. Kaaviosta selviää järjestelmän käyttäjien roolit ja heidän tarpeensa käyttää järjestelmää.



Kuva 8. Ilmoittautumisjärjestelmän käyttötapauskaavio

Ilmoittautujan roolissa ilmoittautumisjärjestelmän käyttö on kurssitarjonnan selaus ja kurssille ilmoittautuminen. Ilmoittautuja ei voi järjestelmän kautta perua ilmoittautumistaan. Ilmoittautujan ei myöskään tarvitse kirjautua järjestelmään sisälle, sillä ilmoittautuminen tapahtuu täyttämällä ja lähettämällä ilmoittautumista varten luotu lomake.

Kurssivastaavalla on ilmoittautujaa enemmän käyttötapauksia. Kurssivastaavan pitää pystyä lisäämään ja poistamaan ilmoittautumisjakso ja sen tiedot ja lisäämään sekä poistamaan kurseja. Kurssivastaavan toinen käyttötapaus on tulostaa tai tallentaa järjestelmästä kurssien ilmoittautumistiedot. Kolmas käyttötapaus on jäsentietojen ylläpito, jotta järjestelmä voi tarkistaa osallistujien jäsenyyden.

3.2.3 Ei-toteutettavat toiminnot

Tilaaajan kanssa on sovittu, että projektin tässä laajuudessa jäsenyyden tarkistamista ei toteuteta kyselynä rajapinnan yli ulkopuoliseen jäsentietojärjestelmään, vaan jäsenyys tarkastetaan ilmoittautumisjärjestelmän sisällä sinne tuoduista jäsentiedoista.

3.3 Ei-toiminnalliset vaatimukset

Tilaajan vaatimukset järjestelmälle ovat suurimmaksi osaksi toiminnallisuuteen liittyviä, joten minulla toteuttajana on hyvin vapaat kädet määritellä ei-toiminnallisia vaatimuksia, kuten esimerkiksi käytettävä sisällönhallintajärjestelmä ja sen myötä ohjelmointikielet, -kirjastot ja tietokannat.

3.3.1 Sisällönhallintajärjestelmän valitseminen

Sisällönhallintajärjestelmä (käytetään myös termiä julkaisujärjestelmä, CMS, Content Management system) on tietojärjestelmä, jonka avulla verkkosivuston sisältöä on helppo hallita ja muokata.

Ilmoittautumisjärjestelmän sisällönhallintajärjestelmäksi valikoitui Wordpress sen monipuolisuuden ja muokattavuuden vuoksi. Lisäksi valintaan vaikutti se, että tilaajalla ei ollut merkitystä sisällönhallintajärjestelmän valinnalle ja Wordpress oli minulle tutuin alusta.

Wordpress on avoimen lähdekoodin sisällönhallintajärjestelmä, jonka etuina ovat muun muassa hakukoneystävällinen rakenne, käyttäjähallinta, muokattavuus ja joustavuus sekä monipuolinen rajapinta lukuisille lisäosille (Wordpress 2019a). Wordpress on maailman suosituin sisällönhallintajärjestelmä (Web Technology Surveys 2019). Sen suosion ja monipuolisen rajapinnan ansiosta lukuisat toimijat Suomessa sekä maailmalla ovat kehittäneet Wordpressille yhteensopivia integraatioita. Wordpress on toteutettu php-koodikielellä ja sen tietokannat ovat MySQL-relaatiotietokantatyyppejä. (Wordpress 2019b)

Wordpressin laajennettavuuden ansiosta on tulevaisuudessa mahdollista laajentaa ilmoittautumisjärjestelmän ominaisuuksia esimerkiksi kommunikoimaan suoraan jäsentietohallintasovelluksen kanssa, mikäli käytettävällä jäsenhallintasovelluksella on olemassa Wordpress-integraatio.

Wordpressin käyttäminen ilmoittautumisjärjestelmän alustana mahdollistaa myös tulevaisuudessa harrastusseuran websivuston liittämisen ilmoittautumisjärjestelmän yhteyteen.

3.3.2 Käytettävyys

Ilmoittautumisjärjestelmän täytyy olla responsiivinen eli skaalautua erilaisille päätelaitteille, jotta sen käyttö onnistuu www-selaimella laitteesta riippumatta.

Ilmoittautumisjärjestelmän ylläpitopuolen tulee olla selkeä käyttää, jotta olisi mahdollisimman helppoa lisätä kursseja järjestelmään, päivittää jäsentietoja sekä listata ilmoittautumistietoja järjestelmästä ulos.

Koska kurssivastaavat saattavat vaihtua vuosittain, on järjestelmän käytön oppimisen oltava helppoa ja selkeää, jotta sen opettelu ei vie liikaa aikaa.

3.3.3 Tietoturva ja toimintavarmuus

Henkilötietoja käsiteltäessä on aina noudatettava tietosuojaperiaatteita. Tietoturva on yksi tietosuojan toteuttamisen keino, joka suojaa tietoaineistoa ja järjestelmää. Tietoturvan avulla varmistetaan, että rekisteröidyn oikeudet toteutuvat ja tiedot pysyvät luottamuksellisina. (Tietosuojavaltuutetun toimisto 2019a)

Ilmoittautumisjärjestelmässä käsitellään ja tallennetaan ilmoittautujien henkilötietoja, joita kurssille ilmoittautuminen edellyttää. Ilmoittautumisjärjestelmältä edellytetään tietoturvallista henkilötietojen käsittelyä ja tallennusta. Ilmoittautumisjärjestelmälle laaditaan tietosuojaseloste, joka koostuu henkilötietolain 24 §:ssä säädetyn tietovielvoitteen edellyttämistä tiedoista sekä henkilötietolain 10 §:n mukaisen rekisteriselosteen edellyttämistä tiedoista (Tietosuojavaltuutetun toimisto 2019b).

Järjestelmän tulee olla mahdollisimman toimintavarma, jotta kurssipaikat saadaan jaettua tasapuolisesti eikä harrastusseuran kurssivastaavien tarvitse valvoa ilmoittautumisia liian tiheästi. Koska ilmoittautumisjärjestelmässä kursseille ilmoittautuminen

alkaa tiettyyn ajankohtaan ja kurssipaikoista on kilpailua, täytyy ilmoittautumisjärjestelmän kestää kävijäpiikkejä ja useita ilmoittautumisia samalla sekunnilla.

3.3.4 Ylläpidettävyys ja huollettavuus

Sivuston ylläpito tarkoittaa sivuston hallintaa ja päivittämistä sen julkaisun jälkeen. Ilmoittautumisjärjestelmän tapauksessa se tarkoittaa esimerkiksi sisällönhallintajärjestelmän pitämistä ajan tasalla. Tärkeimpiä ylläpidollisia toimia ovat järjestelmän säännöllinen varmuuskopiointi ja sisällönhallintajärjestelmän ja lisäosien päivittäminen,

Suurimmaksi osaksi ilmoittautumisjärjestelmän ylläpidosta vastaa palvelintilan palveluntarjoaja, jonka palveluihin sisältyy muun muassa testatut päivitykset ja varmuuskopiointi (WP-Palvelu 2019).

Kurssivastaavilla on järjestelmään pääkäyttäjaoikeudet ja he voivat lisätä tarvittaessa uusia käyttäjiä ylläpitoon. Olosuhteiden muuttuessa voidaan tarpeellisista muutoksista järjestelmän toimintaan sopia erikseen.

3.3.5 Laajennettavuus

Järjestelmään tulee mahdollistaa myös tulevaisuudessa harrastusseuran www-sivuston liittäminen ilmoittautumisjärjestelmän yhteyteen, jolloin www-sivusto ja ilmoittautumisjärjestelmä voisivat käyttää samoja kurssi- ja kouluttajatietoja ja näin helpottaa sisällönsyöttäjien työtä. Tässä projektissa ei määritellä, suunnitella eikä toteuteta muuta www-sivustoa, mutta toteutuksessa tämän liittäminen myöhemmin tulee ottaa huomioon.

3.4 Muut vaatimukset

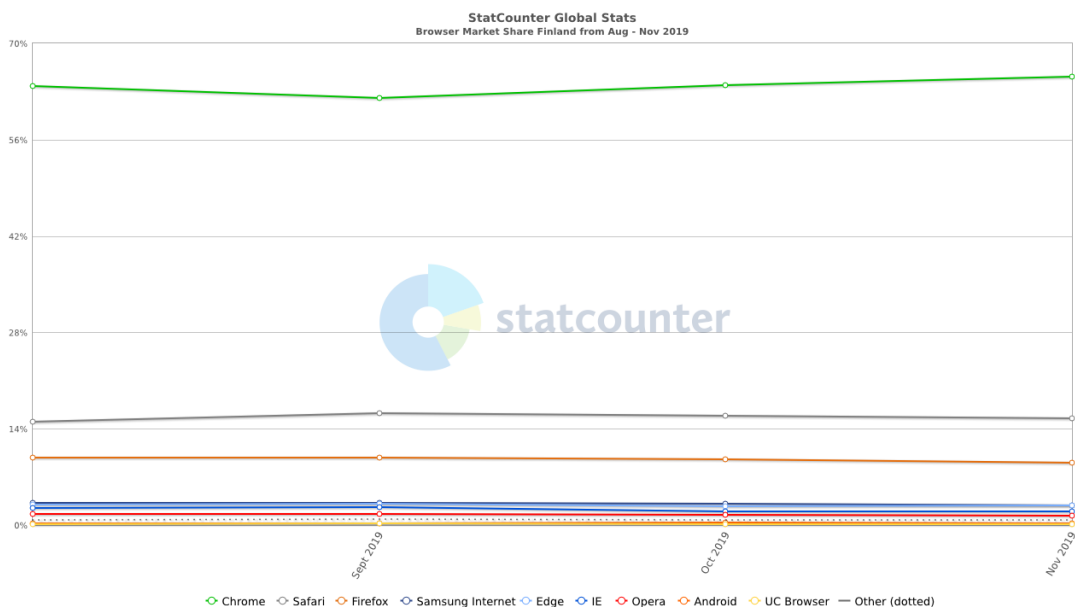
3.4.1 Suorituskyky

Verkkosivuston suorituskyky määritellään usein latausaikojen perusteella. Sivuston latausnopeutta hidastavat mm. suuret tiedostot (esim. suuret kuvat), suuri ladattavien elementtien määrä ja palvelimen hitaus. Latausaika taas vaikuttaa hakukoneoptimointiin ja käyttökokemukseen.

Ilmoittautumissivustolla olisi hyvä välttää suurien kuvien ja muiden hitaasti latautuvien osien käyttöä, jotta sivut latautuvat kohtuullisessa ajassa.

3.4.2 Selaimet

Ilmoittautumisjärjestelmän tulee toimia kaikilla yleisimmillä www-selaimilla. Kuvassa 9 on StatCounterin tilasto pöytäkoneilla käytettyjen selainten jakaumasta Suomessa aikavälillä elokuu–marraskuu 2019. Ylivoimaisesti suosituin selain on Chrome. Tämän päätelmän pohjalta voimme todeta, että järjestelmän tulee toimia moitteettomasti vähintään Chromella, Safarilla, Firefoxilla ja Edgellä.

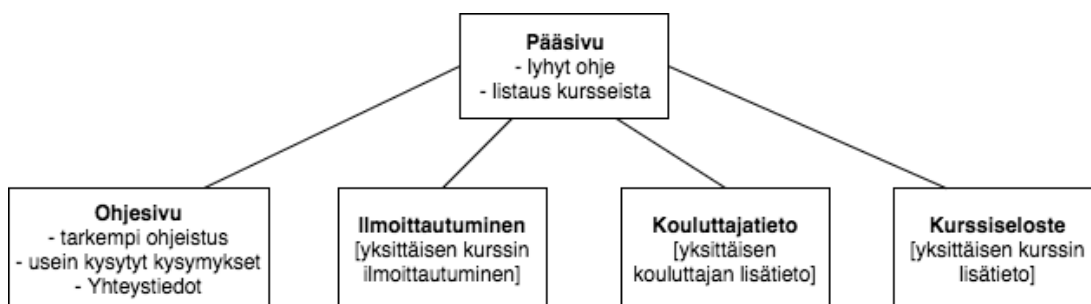


Kuva 9. Selainten käyttäjajakauma Suomessa elokuu–marraskuu 2019 (StatCounter 2019)

3.4.3 Käyttöliittymät

Ilmoittautumisjärjestelmän käyttöliittymä ilmoittautujan näkökulmasta katsottuna sisältää kurssitarjonnan, niille ilmoittautumisen sekä kurssien ja niiden kouluttajien lisätiedot. Pääsivu koostuu lyhyestä ohjeistuksesta ilmoittautumiseen ja kurssien etsimiseen sekä kurssilistauksesta. Koska kursseja on paljon, kurssinäkömään on tarpeellista luoda myös lajittelutyökalu. Kurssien lisätiedot ja kurssien kouluttajien esittelyt aukeavat jokainen omaan näkymäänsä, joka voi olla sivuston sisäinen aukeava näkymä.

Tarkemmat ohjeistukset, usein kysytyt kysymykset ja yhteystiedot kootaan yhdelle ohjesivulle, josta ilmoittautuja löytää helposti tarvitsemansa tiedon. Ilmoittautuminen valitulle kurssille tapahtuu ilmoittautumissivulla. Kuvassa 10 on mallinnettuna ilmoittautumisjärjestelmän käyttöliittymäkaavio.



Kuva 10. Ilmoittautumisjärjestelmän käyttöliittymäkaavio

4 SUUNNITTELU

4.1 Ilmoittautumisjärjestelmän arkkitehtuuri

Korkealla tasolla ilmoittautumisjärjestelmän arkkitehtuuri jakaantuu palvelimeen, tietokantaan ja käyttäjän selaimen. Arkkitehtuurin yhtenä osa-alueena voi olla ulkopuolinen järjestelmä, mikäli jäsennumerolistauksen suunnittelussa ja toteutuksessa päädytään hakemaan nämä tiedot ulkopuolisesta lähteestä.

4.2 Sivurakenne

Ilmoittautumisjärjestelmän sivurakenteen ei tarvitse olla kovin laaja, sillä kurssilistaus esitetään lajittelutyökalun avulla jo pääsivulla. Tämän vuoksi kursseille ei tarvita omia sivuja. Erillisiä sivuja tarvitaan kuitenkin ohjesivulle, tietosuojaselosteelle ja ilmoittautumislomakkeelle. Myös kurssien peruutusehdot on hyvä olla erillisellä sivulla, jotta niihin saadaan linkitettyä helposti. Taulukkoon 1 on koottu tarvittavat sivut, niiden sivumallit sekä keskeiset sisällöt.

Sivu	Sisältö
Pääsivu	Lyhyt ohjeistus, kurssilistaus
Ilmoittautuminen	Yksittäisen kurssin tiedot ja ilmoittautumislomake
Ohjeet	Ilmoittautumisen ohjeet ja usein kysytyt kysymykset
Peruutusehdot	Kurssien peruutusehdot
Tietosuojaseloste	Rekisteri- ja tietosuojaselosteen sisältö

Taulukko 1. Ilmoittautumisjärjestelmän sivurakenne

4.3 Sisältötyypit

Sisältötyypit ovat sisällönhallintajärjestelmän tapa kategorisoida sisältöä. WordPressissä valmiiksi asennettuja sisältötyyppejä ovat esimerkiksi artikkeli, sivu ja liite. WordPressiin on mahdollista luoda omia sisältötyyppejä (custom post types) tarpeen mukaan. (Wordpress 2019c)

Ilmoittautumisjärjestelmä on helpoin toteuttaa käyttämään mukautettuja, juuri tätä järjestelmää varten muokattuja sisältötyyppejä (custom post types), jolloin varsinkin hallintapuolen käyttöliittymästä tulee selkeämpi.

Koska kursseille yhteinen tekijä on ilmoittautumisjakson tiedot, päädyttiin kurssien osalta sisältötyypeihin ilmoittautumisjakso ja kurssi. Tällöin ilmoittautumisjaksoon asetetaan tiedot sille, milloin ilmoittautumisjaksossa alkaa ilmoittautumisten eri vaiheet.

Yksi mietittävä asia kurssi- ja ilmoittautumisjaksosisältötyyppien suunnittelussa on se, halutaanko kursseista toistuvia, jolloin yksi kurssi voi kuulua moneen ilmoittautumisjaksoon. Tämän etuna on se, että joka jaksoon ei tarvitse syöttää kaikkia kurssien tietoja ja kuvauksia aina uudelleen. Tällöin ilmoittautumiset pitää kohdistaa tietyn kurssin sijasta ilmoittautumisjakso–kurssi -yhdistelmään, jolloin se linkittyy kurssin lisäksi myös siihen, mikä on ilmoittautumisjakso. Varsinaisesti ilmoittautuminen linkittyy tietyn ilmoittautumisjakson tiettyyn kurssin variaatioon. Jaksoissa voi olla useita rinnakkaiskursseja. Tämän vuoksi on järkevää liittää jaksoon liitettäviin kursseihin viittaus siihen, mihin varsinaiseen kurssityyppiin jakson kurssi pohjautuu. Vaikka tämä vaihtoehto vaatii enemmän riippuvuuksia sisältötyyppien välillä, koettiin se sopivammaksi vaihtoehdoksi ilmoittautumisjärjestelmälle kuin vaihtoehto, jossa jokaisen jakson kurssit syötetään aina uudelleen.

Kurssisisältötyyppien suunnittelussa otettiin huomioon myös se, että yhdistyksen www-sivusto voitaisiin tulevaisuudessa liittää järjestelmän yhteyteen, jolloin sivustolle lisättävät kurssit käyttäisivät samaa sisältötyyppiä ja tietoja ei tarvitsisi lisätä useampaan paikkaan. Tämän vuoksi kurssisisältötyyppi päädyttiin pitämään yleisellä tasolla tietynlaisena tyyppityksenä. Kurssisisältötyypin kentät ovat kurssin nimi, laji, tyyppi (perus- vai ekstrakurssi) ja kurssikuvaus.

Koska osalle kursseista tarvitaan jäsenyys, tulee jäsenyystiedon olla järjestelmässä saatavilla. Jäsentiedoille tarvitaan oma sisältötyypinsä, jonka tietosisältöä voidaan päivittää ajan tasalle määrätyllä syklillä. Jäsenyysisältötyypistä voidaan

ilmoittautumisen yhteydessä tarkastaa ilmoittautujan jäsenyys. Jäsenyyden sisältökentät ovat jäsennumero ja nimi.

Ilmoittautumisiin yhdistetään myös viitenumero, mikäli ilmoittautuminen ei ole tyypiltään kouluttajailmoittautuminen. Kuten jäsennumerot, viitenumerotkin tulee päivittää listauksena ajan tasalle ennen ilmoittautumisjakson alkua. Viitenumeroille luodaan oma sisältötyyppi, josta voidaan poimia ilmoittautumiseen yksilöllinen viitenumero.

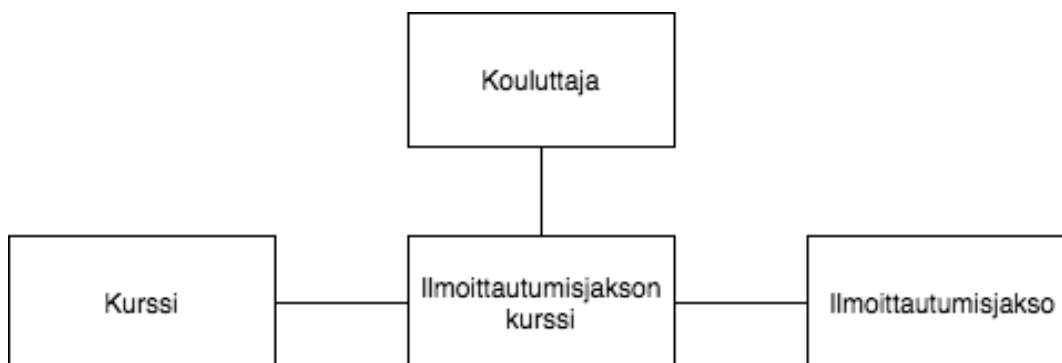
Kouluttajatietoa ei voida liittää kurssin sisältökenttiin, sillä kurseja kouluttavat kouluttajat saattavat vaihdella ja samassa jaksossa samaa kurssityyppiä voi kouluttaa useampi kouluttaja. Tämän vuoksi kouluttajat ovat oma sisältötyypinsä. Kouluttajasisältötyyppiä voidaan myöskin hyödyntää mahdollisesti tulevaisuudessa www-sivustolla, jossa olisi hyvä olla kouluttajien yhteystietolistauksia ja esittelyjä. Kouluttajien tietojen yhteyteen lisätään myös jäsennumero, jonka avulla voidaan kouluttajailmoittautumisessa varmistaa kouluttajarooli. Kouluttajasisältötyypin sisältökentät ovat nimi, sähköpostiosoite, puhelinnumero, jäsennumero, koulutettavat lajit, aktiivisuustilanne, pätevyudet ja esittelyteksti.

Kouluttajien lisäksi seuran hallituksen jäsenillä on etuoikeus ilmoittautua kursseille. Kuten kouluttajatiedoissakin, hallitustiedot kannattaa lisätä omaan sisältötyyppiin, jotta myöhemmin tulevaisuudessa sitä voidaan hyödyntää www-sivustolla. Hallitusrooli voidaan kouluttajaroolin tavoin varmistaa etukäteisilmoittautumisessa hallitusjäsenen asetetun jäsennumeron avulla. Hallitusjäsenen sisältökentät ovat nimi, puhelinnumero, sähköpostiosoite, rooli hallituksessa ja jäsennumero.

Kurssi- ja kouluttajasisältötyyppien suunnittelussa tuli jo esille se, että yksi niitä yhdistävä tekijä on laji (koiraharrastuslaji). Ilmoittautumisjakson kursseilla pitää myös näkyä, mihin lajiin tämä kuuluu. Laji on siis yksi oma sisältötyypinsä, jonka kenttiä ovat lajin nimi ja lajin kuvaus. Lajisisältötyyppi linkitetään kurssisisältötyyppiin.

Tietyn ilmoittautumisjakson ilmoittautumiset kohdistetaan ilmoittautumisjakson, kurssin ja kouluttajan yhdistelmään kuvan 11 mukaisesti. Tämä voidaan mallintaa omaksi sisältötyypikseen. Koska hallintapuolen käytännöllisyys ja helppokäyttöisyys ovat avainasemassa, kannattaa tämä toteutuksessa muodostaa Wordpressissä

ilmoittautumisjaksosisältötyypin alle toistuviksi riveiksi. Tällöin kurssit saa lisättyä suoraan halutun ilmoittautumisjakson yhteyteen.



Kuva 11. Ilmoittautumisjakson kurssiin vaikuttavat tekijät

Ilmoittautumisen sisältötyyppien suunnittelussa täytyy pohtia se, miten tarkasti osallistujat halutaan eritellä ilmoittautumisissa. Koska kurssit ovat koiraharrastusseuran kursseja, liittyy ilmoittautumisiin olennaisesti myös koiran tiedot. Harrastusseuralla on kuitenkin olemassa myös kursseja, jotka ovat ainoastaan koiranohjaajille, jolloin koiran tietoja ei tule ilmoittautumiseen. Yhdelle kurssille voi yksittäinen koiranohjaaja ilmoittautua myös useamman koiran kanssa. Koiranohjaaja voi myös saman koiran kanssa ilmoittautua useammalle eri kurssille. Ei voida myöskään olettaa, että tietyt koirat linkittyvät vain tiettyihin ohjaajiin, vaan yksittäinen koira voi osallistua eri ohjaajien kanssa eri kursseille. Näiden seikkojen pohjalta voidaan ajatella, että ilmoittautuminen koostuu ilmoittautumisjakso-kurssi -yhdistelmästä sekä ohjaajasta ja koirasta.

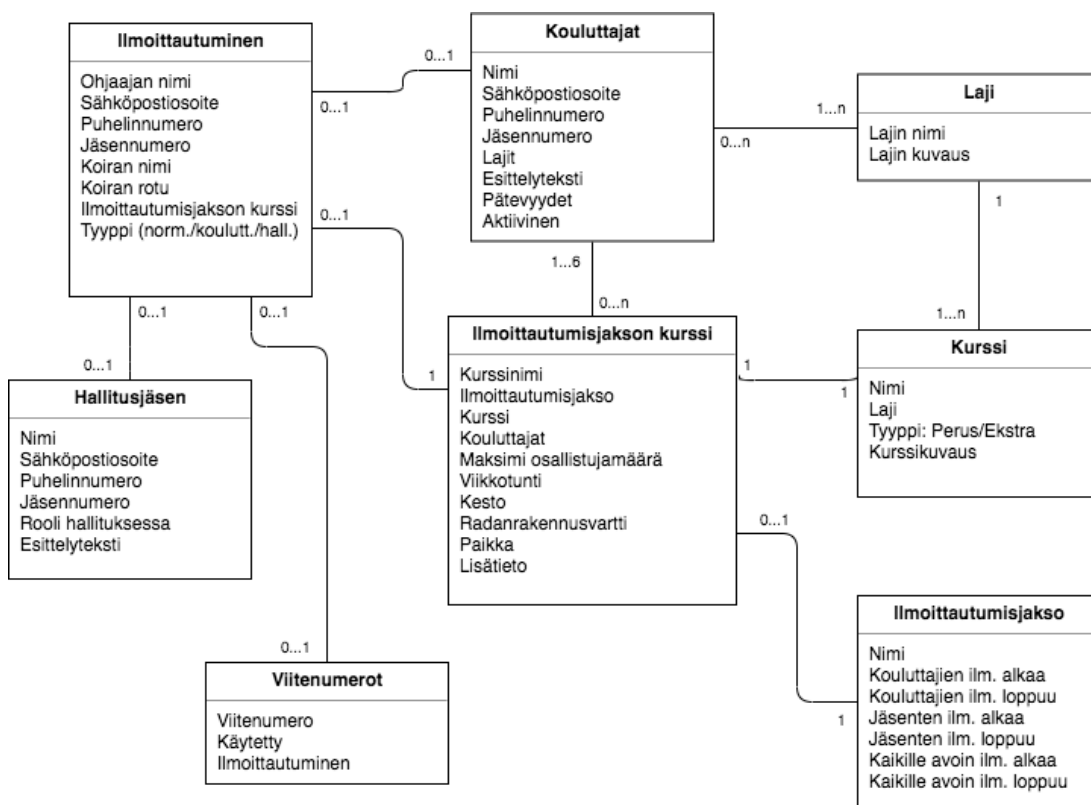
Järjestelmässä ei ole tarpeellista ylläpitää koirien tietoja erillisinä, sillä koiran tiedoista ilmoittautumisen yhteydessä ei kerätä mitään sellaista tietoa, jonka avulla voitaisiin identifioida koira. Voi siis olla useampi Musti-niminen sekarotuinen koira. Tämän vuoksi koiran tiedot kannattaa sisällyttää ilmoittautumisen tietoihin omiksi tekstimuotoisiksi kentikseen sen sijaan, että sillä olisi riippuvuus koirasisältötyyppiin.

Toisin kuin koirat, koiranohjaajat voidaan identifioida esimerkiksi sähköpostiosoitteen avulla. Oman sisältötyypin luomisessa koiranohjaajille täytyy pohtia se, tarvitaanko koiranohjaajan tietoja muualla, kuin ilmoittautumisen tietoina. Tällainen tilanne voisi olla esimerkiksi koiranohjaajien kirjautumisen mahdollistaminen järjestelmään. Koska tässä ilmoittautumisjärjestelmässä ei tilaajan kanssa pohdittaessa ole nähty

tarpeellista hyötyä työmäärään nähden luoda koiranohjaajille omia profiileita, ei ole myöskään tarpeellista luoda koiranohjaajista omaa sisältötyyppiä.

Ilmoittautuminen on siis tämän järjestelmän yksi sisältötyyppi, jonka tietokenttinä on se, mihin ilmoittautumisjakson kurssiin ilmoittautuminen kuuluu sekä ohjaajan ja koiran tiedot.

Kuvassa 12 on esitetty luokkakaaviona ilmoittautumisjärjestelmän sisältötyypit, niiden sisältämät tietokentät sekä sisältötyyppien väliset suhteet.



Kuva 12. Luokkakaaviomallinnus ilmoittautumisjärjestelmän sisältötyypeistä

4.4 Toiminnallisuuksien suunnittelu

4.4.1 Kurssihallinta

Kursseille yhteinen tekijä on ilmoittautumisjakso. Ilmoittautumisjaksoon asetetaan tiedot sille, milloin jaksossa eri ilmoittautumisen vaiheet alkavat ja loppuvat.

Ilmoittautumisjakson asetusten syöttämisessä järjestelmän tulee tarkastaa, että päivämäärät ovat suhteessa toisiinsa kelvolliset. Tämä tarkoittaa sitä, että ilmoittautuminen ei voi esimerkiksi loppua ennen kuin se alkaa tai ilmoittautumisen toinen vaihe ei voi alkaa ennen ensimmäisen vaiheen loppumista.

Määrittelyvaiheessa päädyttiin siihen, että kurssit liitetään ilmoittautumisjaksosisältötyypin yhteyteen jakson alaisiksi kurseiksi. Jotta kurseja voidaan näkymässä lajitella ja järjestää, pitää kurssien ajankohdasta ja tyypistä lisätä myös tiedot kurssitietojen yhteyteen. Osalla kurseista on säännöllinen viikkotunti ja nämä voisi järjestää esimerkiksi viikkotunnin mukaan. Tässä tapauksessa kurssien ajankohdalle tarvitaan syötettäväksi erikseen viikonpäivä sekä kellonaika.

Vaikka mikään ilmoittautumisen vaihe jaksossa ei olisi käynnissä, on kurssien silti tarkoitus olla näkyvillä, jotta ilmoittautujat voivat jo etukäteen tutustua kurssivalikoimaan. Vanhassa järjestelmässä käyttäjät ovat tottuneet siihen, että ilmoittautumislomakkeen voi täyttää jo valmiiksi ja lähettää vasta ilmoittautumisen alkaessa. Tässä uudessa järjestelmässä päädyttiin siihen, että kurssien ilmoittautumislomakkeet aukeavat täytettäväksi 10 minuuttia ennen ilmoittautumisen alkamista. Kurssitietojen osalta tämä tarkoittaa sitä, että ilmoittautumislomakkeen linkki on näkyvillä vain silloin, kun lomake on avoinna.

4.4.2 Jäsen- ja viitenumerotietojen päivittäminen

Jäsentietojen tarkastus ilmoittautumisen yhteydessä, tai oikeastaan jäsentietojen päivittävyys järjestelmän sisältötyypin tietosisältöön, on ollut yksi suunnittelun pohdituimpia asioita. Määrittelyvaiheessa rajattiin tilaajan kanssa aikataulu- ja budjettisyistä pois vaihtoehto siitä, että tietoja kyseltäisiin ilmoittautumisen yhteydessä ulkopuolisesta jäsentietojärjestelmästä rajapinnan välityksellä.

Helpoin vaihtoehto varmasti olisi ylläpitää manuaalisesti tietosisältöä, mutta tässä on suuri inhimillisen virheen riski, työmäärä on suuri ja tästä vanhan järjestelmän tavasta juuri haluttiin päästä eroon. Lista siis halutaan päivittää yhtenäisenä. Tämä voidaan toteuttaa WordPressin lisäosan avulla, jossa ladattava tiedosto voidaan viedä

esimerkiksi csv- tai xml-muodossa järjestelmään ja päivittää siten sisältötyypin tietosisältö. Jos listaus sijaitsee ulkopuolisessa lähteessä, on mahdollista myös ajastaa tietojen tuonti järjestelmään käynnistymään valituin väliajoin.

Viitenumerolistauksen päivittäminen järjestelmään on järkevää toteuttaa samalla tavalla kuin jäsentietojen päivittäminen: yhtenä listauksena tietojen tuonti sisältötyyppiin. Eroavaisuus näiden sisältötyyppien päivittämisessä on se, että viitenumeroita vain lisätään, niistä ei poisteta listasta puuttuvia. Tämä siksi, että ilmoittautumisjakson lopuksi järjestelmään jää listaus rahastonhoitajalle, mikä ilmoittautuminen on liitetty mihinkin viitenumeroon.

4.4.3 Ilmoittautuminen

Ilmoittautumisjakso jakaantuu ajallisesti kolmeen vaiheeseen: kouluttajien sekä hallituksen jäsenten etukäteisilmoittautumiseen, varsinaiseen ilmoittautumiseen ja kaikille avoimeen ilmoittautumiseen, jossa jäsenyysvaatimuksia ei ole millekään kurssille. Järjestelmän näkökulmasta nämä käyttävät kaikki lähes samaa toimintamallia.

Suunnittelussa päädyttiin siihen, että kurssien ilmoittautumislomakkeet aukeavat täytettäväksi 10 minuuttia ennen ilmoittautumisen alkamista, mutta kuitenkin niin, että lomaketta ei voi lähettää etukäteen. Käytännössä tämä tarkoittaa sitä, että tänä aikana lomakkeen yllä on hyvä olla huomautusteksti siitä, milloin ilmoittautuminen alkaa ja lähetuspainike ajastetaan näkymään vasta kun aika täyttyy.

Liitteessä 1 on esitetty vuokaaviomalli ilmoittautumisen suunnitellusta etenemisestä. Ilmoittautumisen käyttötapauksessa ensimmäisenä ilmoittautuja valitsee kurssin, täyttää ilmoittautumislomakkeen ja lähettää sen. Ennen kuin ilmoittautuminen voidaan tallentaa onnistuneesti, on järjestelmällä monta tarkistustehtävää.

Ensimmäinen tarkastus lähetetyssä ilmoittautumisessa tehdään kurssin vapaiden paikkojen suhteen, ettei kurssi ole ehtinyt täyttyä sillä välin, kun ilmoittautuja täytti lomaketta. Mikäli kurssilla on tilaa, jatketaan tarkastelemalla ilmoittautumisen aika-asetuksia ilmoittautumisjakson tiedoista. Jos on voimassa kouluttajien ja hallituksen jäsenten

ilmoittautuminen, tarkastetaan kouluttajan tai hallituksen jäsenen jäsennumero. Jos taas varsinainen ilmoittautuminen, tarkastetaan peruskurssien osalta jäsenyys.

Ilmoittautumisen onnistuessa lähetetään ilmoittautujalle vahvistusviesti sähköpostitse. Vastaavasti ilmoittautumisen epäonnistumisen tilanteessa ilmoittautujalle näytetään ilmoitus epäonnistuneesta ilmoittautumisesta ja syy tilanteelle.

Mikäli kyseessä ei ole ollut kouluttajan ilmoittautuminen, lisätään ilmoittautumisen tietoihin viitenumerosisältötyypistä vapaa viitenumero.

4.4.4 Ilmoittautumistietojen käsittely

Ilmoittautumisjakson lopussa kurssivastaavan tehtäviin kuuluu osallistujatietojen hakeminen järjestelmästä sekä kouluttajia varten että harrastusseuran rahastonhoitajaa varten. Nämä listat on hyvä saada järjestelmästä taulukkomuodossa, jotta kurssivastaavien on niitä helppo käsitellä ja jakaa kouluttajille.

Ilmoittautumistietojen vieminen järjestelmästä pois täytyisi tapahtua mahdollisimman helposti. Tämän vuoksi suunnitelmiin tuli automaatiotyökalun käyttö, jonka tarkoitus on viedä määrätty vientitiedosto ulkopuoliseen palveluun, esimerkiksi Google Driveen.

Yleiseen kurssi- ja ilmoittautumistietojen hallintaan järjestelmässä käytetään Wordpressin omaa hallintapuolen käyttöliittymää ja sen toimintoja.

4.5 Tietoturva

Wordpress alustana on melko tietoturvallinen. Sen haasteet tietoturvan suhteen liittyy suurimmaksi osaksi käyttäjien huonoon salasanahygieniaan ja siihen, että sivustojen päivityksistä ei huolehdi. On siis tärkeää huolehtia päivityksistä sivustolla. Päivitysten tulee kattaa sekä Wordpressin ydinosat että lisäosat. (Seravo 2019)

Ilmoittautumisjärjestelmässä valmiin järjestelmän palvelintilan palveluntarjoajaksi on valittu WP-palvelu, jonka peruspalveluihin kuuluu testatut päivitykset, ympärivuorokautinen valvonta ja tietoturvatakuu (WP-Palvelu 2019). Tietoturvatakuusta huolimatta asiakkaan vastuulla on salasanojen ja sivuston huolellinen käyttö. Hallintapuolelle kirjautumisessa käytetään WordPressin omaa kirjautumista ja sinne asetetaan vahvan salasanan vaatimus, jotta riski ylläpitäjän salasanan murtamiseen pienenee.

Palveluntarjoajan toimesta sivuston yhteydet ovat salattuja eli käytössä on SSL-sertifikaatti. SSL on lyhenne termistä Secure Sockets Layer ja se on salausprotokolla, jolla voidaan suojata arkaluontoista tietoa kahden järjestelmän välillä (Digicert 2019).

Varmuuskopio sivustosta valitussa ympäristössä tallentuu palvelimelle kerran vuorokaudessa automaattisesti palveluntarjoajan toimesta. Tämä helpottaa sivuston ylläpidossa.

4.6 Ulkoasu ja käyttöliittymä

Tavoitteena on, että uusi ilmoittautumisjärjestelmä on käyttäjille käyttöliittymältään helppokäyttöinen ja käytännöllinen, skaalautuu hyvin sekä suuremmille näytöille että eri mobiililaitteille.

Kuvassa 14 on rautalankamalli ilmoittautumissivuston pääsivusta. Sivun yläosassa tärkeimmät ohjeet ilmoittautumiseen sekä ohjaus ohjesivulle, jonne kootaan tarkempi ohjeistus ja usein kysytyt kysymykset –osio. Koska kursseja on yhdessä ilmoittautumiskaksossa keskimäärin yli 20 kappaletta, on lajitteluvalinta kurssilistausnäkyään tarpeellinen.

LOGO

Ilmoittautuminen

Tässä on ohjetekstiä. Lorem ipsum dolor sit amet.

Ohjeet

Näytä kurssit

Perusteet ▼ Oletus Viikkotunnin mukaan

Alkeiskurssi 1	ma 18.15	Ilmoittaudu
Harrastuskoiran peruskurssi 1	ma 19.15	Ilmoittaudu
Harrastuskoiran peruskurssi 2	ke 18.15	Ilmoittaudu
Alkeiskurssi 2	ma 18.15	Ilmoittaudu

Kuva 14. Ilmoittautumissivun rautalankamalli

Kurssien lajitteluun mahdollisia määreitä ovat harrastuslaji, kurssityyppi ja jäsenyysvaatimus sekä kouluttaja. Edellä mainituista kouluttajatieto voidaan jättää pois, sillä se ei ole niin olennainen osa käyttäjälle kurssia valitessa. Tärkein tekijä käyttäjälle on varmasti harrastuslaji, joten se on järjestelmän tärkein lajittelutekijä. Tämän lisäksi on hyvä erotella kurssit, joille ei tarvita jäsenyyttä.

Kurssien järjestämiseen vaikuttavia tekijöitä voi olla kurssin viikkotunti, nimi tai se, onko kurssi täynnä. Ilmoittautumisjärjestelmässä päädyttiin oletuksena listaamaan kurssit niin, että ne järjestyvät kurssin nimen mukaisesti aakkosjärjestykseen, mutta kuitenkin niin, että täynnä olevat kurssit näkyvät viimeisenä. Järjestysvalinnaksi voi valita myös viikkotunnin.

Viikkotunnin mukaan järjestämisessä on se haaste, että kaikilla kursseilla ei välttämättä ole säännöllistä viikkotuntia, ja tällöin kurssit, joilla ei ole viikkotuntia, listataan aakkosjärjestykseen muiden jälkeen.

Ilmoittautumislomakenäkymä on suunniteltu kahdelle palstalle, jossa vasemman puoliselle palstalle listautuu näkyviin kurssin tiedot sekä kurssiseloste ja oikealle puolelle lomake. Tämä auttaa varmistamaan sen, että ilmoittautuja ilmoittautuu varmasti oikealle kurssi.

Ulkoasun tyyleihin liittyen TVA Porilla ei ole varsinaista graafista ohjeistoa, mutta logo ja sen myötä päävärit on määritelty logossa käytettyyn keltaiseen, harmaaseen ja mustaan. Nämä huomioon ottaen minulla on toteuttajana hyvin vapaat kädet myös ilmoittautumisjärjestelmän ulkoasun suunnittelulle.

5 TOTEUTUS

Tässä ilmoittautumisjärjestelmän tuotantoprojektissa toteutusta tehtiin osittain samaan aikaan suunnitteluvaiheen kanssa, osittain saatettiin palata jopa määrittelyvaiheeseen asti yksittäisten osa-alueiden suhteen.

5.1 Wordpress alustana

Wordpress on yksi suosituimmista sisällönhallintajärjestelmistä ja sillä on maailmanlaajuisesti vilkkaita käyttäjä-, kehittäjä- ja tukiyhteisöjä. Wordpress erottuu muista sisällönhallintajärjestelmistä varsinkin laajalla kirjolla lisäosia eli toiminnallisia laajennuksia.

Käytännössä Wordpress koostuu kolmesta osasta: ytimestä, teemasta ja lisäosista. Ydin (core) on ikään kuin Wordpressin perusta, kivijalka. Ydin sisältää tiedostot, jotka muodostavat sisällönhallinnan toiminnallisuuden, käyttöliittymän ja ulkoasun. Tämän lisäksi ytimen tiedostot määrittelevät asetukset monelle asialle, esimerkiksi miten lisäosat ladataan ja miten sisältösivut luodaan. Itse sivuston muokattuihin toiminnallisuuksiin ja ulkoasuun liittyvät tiedostot tulevat teeman kautta. Lisäosien avulla voidaan teeman ulkopuolelta lisätä sivustolle toiminnallisuuksia. (Williams, Damstra & Sabin-Wilson 2014, 61–62)

Tässä projektissa on hyödynnetty Wordpressin laajennettavuutta ja luotu mukautettu teema, jonne on luotu toiminnallisuutta ulkoasu- ja käyttöliittymämäärittysten lisäksi. Osa toiminnallisuuksista on tuotu lisäosien avulla. Näiden osa-alueiden muokkaamisesta kurssijärjestelmän tarpeisiin on käsitelty tarkemmin seuraavissa kappaleissa.

5.1.1 Teemat

Teema ovat Wordpressin ytimen liitännäinen, joka ohjaa sitä, miten sivusto ja sen sisältö esitetään www-selaimessa. Teema määrittää sen, missä ja miten sisältö esitetään. Teema yhdistelee erilaisia tiedostoja ja tiedostotyyppejä: PHP-, CSS- ja JavaScript-

tiedostoja. Näiden avulla teema muodostaa sivumalleja, tyylejä ja toiminnallisuuksia. Sivustolla voi olla asennettuna useampi teema, mutta vain yksi voi olla aktiivinen. (Williams ym. 2014, 231–234)

Teemoja on kahta päätyyppiä: projektiteemat ja lapsiteemat. Projektiteema voi olla kertaluonteinen tyhjältä rakennettu, jota muokataan tiettyä projektia varten. Usein projektiteemat ovat kuitenkin aloitusteemoja, joilla on tarkoituksellisesti minimaalinen muotoilu. Tällöin varsinaiseksi teemaksi asetetaan lapsiteema, joka perii isäntäteemaltaan tyylit ja toiminnallisuudet. Lapsiteemassa voidaan yliajaa halutut tiedostot ja siten muokata teema sopimaan omiin tarpeisiin. (Williams ym. 2014, 236)

Tässä projektissa aloitusteemaksi valittiin Bootstrap-pohjainen perusteema, jolle luotiin lapsiteema. Aloitusteema on hyvin minimaalisesti muotoiltu, mutta sisältää valmiina Bootstrap-kirjaston ja selkeän rakennepohjan, jonka pohjalle on helppo rakentaa projektin tarpeisiin sopiva lapsiteema.

Bootstrap on maailman suosituin käyttöliittymäpuolen komponenttikirjasto. Se perustuu avoimeen lähdekoodiin ja sisältää työkaluja HTML-, CSS- ja JavaScript-kehittämiseen. (Bootstrap 2019a)

5.1.2 Lisäosat

Wordpressin lisäosat (plugins) ovat PHP-scriptejä, joiden avulla voidaan laajentaa tai muuttaa Wordpressin toimintaa. Lisäosat ovat vuorovaikutuksessa Wordpressin ytimen kanssa kahden erityyppisten toimintokutsujen (hooks) kautta: toimintojen (actions) ja suodattimien (filters). Toiminto voidaan laukaista suoritettavaksi tietyssä kohdassa määrättyä tapahtumaketjua. Suodattimien avulla taas voidaan muokata sisältöä määritellyssä vaiheessa, esimerkiksi tietokannasta hakemisen jälkeen. (Williams, Richard & Tadlock 2011, 1–4)

Wordpressin suosio perustuu osittain sen helppoon laajennettavuuteen lisäosien avulla ja niitä on saatavilla hyvin laaja kirjo. Wordpress.orgin lisäosahakemistossa on saatavilla yli 54 000 avoimen lähdekoodin lisäosaa (Wordpress 2019d). Suuri määrä

lisäosia aiheuttaa sen, että omiin tarpeisiinsa sopivaa lisäosaa saattaa olla vaikea valita monien joukosta.

Tässä projektissa ilmoittautumisjärjestelmän Wordpressiin asennettiin muutamia lisäosia, joiden avulla saatiin laajennettua toimintoja ja helpotettua esimerkiksi sisältörakenteiden luomista. Asennettuja valmiita lisäosia ovat muun muassa Advanced Custom Fields, Custom Post Type UI, Ninja Forms ja Admin Columns.

5.2 Sisältörakenteiden luominen

5.2.1 Sivut

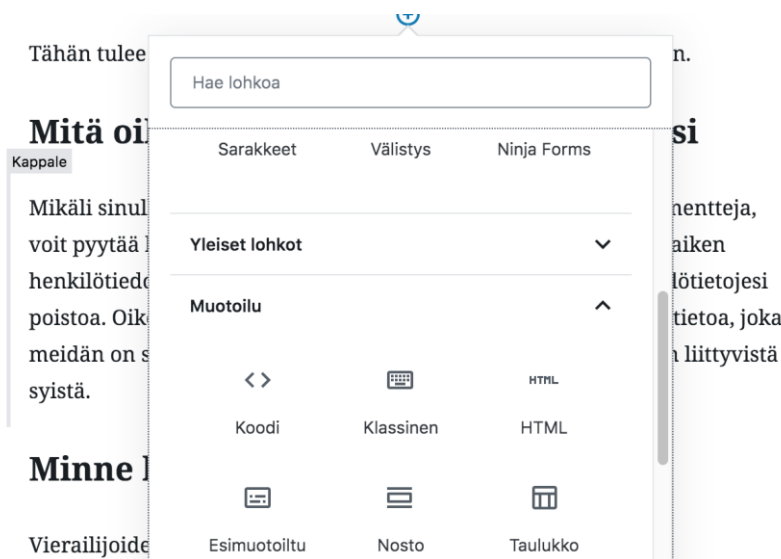
Wordpressin sivurakenteeseen luotiin suunnitteluvaiheessa suunnitellut sivut: etusivu, ilmoittautuminen, ohjeet, ehdot ja tietosuojaseloste. Näille ei tehty erillisiä sivumalleja, sillä tarvittavat mukautetut elementit, kuten kurssilistaus ja ilmoittautumislomake saadaan tuotua sivun pääsisältöön niitä varten tehtyjen lohkojen avulla. Tarkempi kuvaus lohkoista on tämän työn kohdassa 5.2.2.

Lohkorakenteeseen päädyin sen vuoksi, että tällöin voidaan valita tietyille sivulle näkymään haluttu ilmoittautumisjakso ja näin kurssivastaavat voivat luoda suunnittelusivun, jonka avulla voidaan hahmotella jo seuraavan ilmoittautumisjakson kurssivalikoimaa, vaikka edellinen ilmoittautumisjakso olisi vielä käynnissä varsinaisella ilmoittautumissivulla. Tämä ratkaisu onnistuisi kyllä sivumallienkin avulla, mutta lohkorakenne tuo enemmän muokattavuutta ja joustavuutta sivuille.

5.2.2 Lohkot

Lohkoja (blocks) voisi kuvata sisällön eräänlaisiksi rakennuspalikoiksi, joiden avulla sivustoa rakennetaan. Lohkot liittyvät Wordpressin joulukuussa 2018 julkaisemaan Gutenberg-editoriin, joka tuli version 5.0 myötä (Wordpress 2019e).

Valmiina Gutenberg-editorissa on lukuisia määriä eri tyyppisiä lohkoja, jotka on ryhmitelty käyttötarkoituksen mukaan, esimerkiksi aseteluun tai muotoiluun liittyvät lohkot (Wordpress 2019f). Kuvassa 15 on näkymä Gutenberg-editorin valittavista lohkoista. Gutenbergin lohkovalikoimaan on mahdollista tehdä myös omia muokattuja lohkoja, jolloin tarvittavat muokatut sisältöosiot saa tuotua lohkona haluttuun kohtaan sivua.



Kuva 15. Gutenberg-editorin lohkovalintatyökalu

Ilmoittautumisjärjestelmässä käytettiin muokattuja lohkoja esittämään sisältöosia sivuilla, esimerkiksi usein kysytyt kysymykset, kurssilistaus ja ilmoittautumislomake. Näiden muodostamisessa apuna käytettiin ACF-lisäosan (Advanced Custom Fields) lohkokoreksteröintiä. Kuvassa 16 on esimerkki ilmoittautumisjärjestelmän Usein kysytyt kysymykset -lohkon luomisesta functions.php-tiedostossa. (Advanced Custom Fields 2019)

```

acf_register_block(array(
    'name'           => 'faq',
    'title'          => __('Usein kysytyt kysymykset'),
    'description'    => __('Add FAQ element.'),
    'render_template' => 'template-parts/blocks/faq.php',
    'category'       => 'common',
    'icon'           => 'welcome-add-page',
    'keywords'       => array( 'faq' ),
    'mode'           => 'auto',
    'align'          => 'full',
));

```

Kuva 16. Usein kysytyt kysymykset -lohkon luominen

Lohkojen luomisessa määritellään tiedosto, jonne lohkon muodostava koodi kirjoitetaan. Lohko voi lukea tarvittavaa sisältöä muista elementeistä tai sisältötyypeistä tai sivulle voidaan määritellä muodostettavaa lohkoa varten omia sisältö- tai asetuskenttiä. Kurssilistauslohkossa sivun asetuksissa määritellään, minkä ilmoittautumisjakson kurssit lohkossa näytetään ja lohko hakee kurssit sen mukaisesti.

5.2.3 Sisältötyypit

Wordpressiin voidaan lisätä sivuston tarpeisiin sopivia muokattuja sisältötyyppejä (custom post types). Sisältötyypeille voidaan luoda myös luokitteluun tarkoitettuja taksonomioita ja ne voivat olla hierarkisia. Sisältötyyppejä voidaan luoda register_post_type-funktion avulla teeman functions.php-tiedostossa tai erillisessä lisäosassa. (Williams ym. 2014, 128)

Tässä projektissa mukautetut sisältötyypit luotiin käyttäen Custom Post Type UI -lisäosaa, sillä sen avulla sisältötyyppien luominen ja asetusten hallinta sekä muokkaus saadaan yhdistettyä Wordpressin hallintapaneeliin. Taulukkoon 2 on koottu ilmoittautumisjärjestelmään lisätyt sisältötyypit.

Sisältötyyppi	Slug	Hierarkisuus
Kurssiselosteet	kurssi	ei
Kouluttajat	kouluttaja	ei
Ilmoittautumisjaksot	jakso	ei

Ilmoittautumiset	ilmoittautumiset	ei
Harrastuslajit	laji	ei
Jäsennumerolistat	jasennumerot	ei
Hallituksen jäsenet	hallitusjasenet	ei
Viitenumerot	viitenumerot	ei
Usein kysytyt kysymykset	ukk	ei

Taulukko 2. Mukautetut sisältötyypit ilmoittautumisjärjestelmässä

5.2.4 Kentät

Metatiedot (metadata) ovat sisältötyyppeihin liitettyjä lisätietoja, joita Wordpressissä kutsutaan tavallisesti mukautetuiksi sisältökentiksi (custom fields). Näiden avulla voidaan sisältötyyppeihin liittää tietoja ja määritellä, missä muodossa ne liitetään. Tietokannassa nämä tiedot lisätään tietokannan tauluun wp_postmeta. Mukautettuja sisältökenttiä luodaan Wordpressissä funktion add_post_meta() avulla. (Williams ym. 2014, 147–148)

Tässä projektissa on käytetty mukautettujen sisältökenttien luomiseen erillistä lisäosaa Advanced Custom Fields, sillä sen avulla on helpompi sisällönhallinnan käyttöliittymän kautta hallita ja muokata luotuja sisältökenttiä.

Ilmoittautumisjärjestelmässä esimerkiksi ilmoittautumisjakso tarvitsee lisätietokenttiä tiedoille ilmoittautumisjakson alkamisesta ja päättymisestä. Taulukossa 3 on esitetty sisältötyypille kouluttaja määritellyt mukautetut sisältökentät. Taulukossa ensimmäisessä sarakkeessa on sisältökentän nimiö, joka näkyy hallinnan käyttöliittymässä kentän otsakkeena. Taulukon toisessa sarakkeessa on kentän nimi, jonka avulla teeman koodissa voidaan hakea kentän sisältö. Taulukon kolmas sarake kertoo, minkä tyyppinen kentän sisältö on.

Nimiö	Nimi	Tyyppi
Etunimi	etunimi	Teksti
Sukunimi	sukunimi	Teksti

Aktiivinen	aktiivinen	”Tosi / Epätosi” -valinta
Jäsennumero	jasennumero	Teksti
Sähköpostiosoite	sahkopostiosoite	Teksti
Puhelinnumero	puhelinnumero	Teksti
Lajit	lajit	Artikkeliolio
Pätevyudet	patevyydet	Teksti

Taulukko 3. Sisältötyypin kouluttaja mukautetut sisältökentät

5.3 Toiminnallisuuksien toteuttaminen

Wordpressissä toiminnallisuuksia voi lisätä joko lisäosien kautta liittämällä tai teemaan lisäämällä. Konfiguroitavissa olevat ja sellaiset toiminnallisuudet, jotka voidaan myös ottaa tarvittaessa pois käytöstä, kannattaa liittää lisäosien avulla.

Tiedosto `functions.php` on hyvin tärkeä tiedosto teemassa, sillä se toimii eräänlaisena kirjastona koodeille, joiden kautta laajennetaan Wordpressin toiminnallisuuksia. Wordpress sisällyttää tämän suoritukseen aina, joten sen sisältöä voi käyttää teeman missä osassa tahansa. `Functions.php`-tiedostoon kannattaa sisällyttää sellaiset toiminnallisuudet, joita teemassa tarvitaan usein tai jotka ovat välttämättömiä teeman toiminnalle, kuten esimerkiksi valikot. (Williams ym. 2014, 261–263)

Ilmoittautumisjärjestelmässä `functions.php`-tiedostoon on teeman perustoimintojen ja ulkoasuun vaikuttavien koodien lisäksi lisätty apufunktioita, joita käytetään muodostettaessa ilmoittautumisjärjestelmän toiminnallisuuksia. Esimerkki tällaisesta apufunktiosta on `functions.php`-tiedostoon lisätty funktio `get_current_registrationperiod()`, joka palauttaa sen hetken ilmoittautumisjakson vaiheen eli sen, mille kohderyhmälle ilmoittautuminen on funktion kutsuhetkellä avoinna.

5.3.1 Hallintapuolen toiminnallisuudet

Määrittelyvaiheen käyttötapauskaavion mukaan kurssivastaavan käyttötapauskaavioita on kurssien lisääminen ja poistaminen, ilmoittautumisten hallinta sekä jäsentietojen ja

viitenumeroiden lisääminen. Suurimmaksi osaksi näiden hallinta tapahtuu WordPressin oman käyttöliittymän kautta, eikä tarvitse perustoimintoihin erillistä toiminnallisuuksien rakentamista.

Ilmoittautumisjaksoa lisättäessä on tärkeää, että jakson aika-asetukset ovat kelvollisia. Ilmoittautumisjaksoon asetetaan viisi aikamäärettä, joiden järjestys pitää olla looginen, jotta ilmoittautuminen ei voi esimerkiksi loppua ennen kuin on alkanut. Tällaisten asetusten määrittämiseen on käytetty teeman `functions.php`-tiedostossa suodattimia (filters), jotka tarkistavat ilmoittautumisjaksoa tallennettaessa päivämääräkenttien kelvollisuudet. Kuvassa 17 on esimerkki suodattimesta, joka tarkistaa ilmoittautumisjakson mukautettuun kenttään syötetyn päivämäärän kelvollisuuden vertaamalla sitä suorituksen aikaleimaan sekä muiden kenttien sisältöön.

```
add_filter('acf/validate_value/name=end_ilmo_ennakko', 'validate_trainers_end_date_func', 10, 4);
function validate_trainers_end_date_func($valid, $value, $field, $input) {
    if (!$valid) {
        return $valid;
    }
    $start_key = 'field_5da430220b668';
    $end_key = 'field_5da430a00b669';
    $start_value = $_POST['acf'][$start_key];
    $end_value = $value;
    if ($end_value <= $start_value) {
        $valid = 'Tarkista ilmoittautumisen loppuminen. Ei voi olla ennen ilmoittautumisen alkamista.';
    }
    return $valid;
}
```

Kuva 17. Esimerkki mukautetun kentän kelvollisuuden tarkistamisesta

Hallintapuolen toiminnallisuuksia on laajennettu myös lisäosien avulla. Mukautettujen sisältötyyppien tietojen päivittämiseen WordPressissä on käytetty lisäosaa WP All Import, joka annettujen asetusten mukaan lisää, poistaa ja muokkaa sisältöä.

Järjestelmässä on tarve myös päivittää tietoja järjestelmään ulkopuolisista lähteistä, kuten esimerkiksi viitenumeroistaukset sisältävästä Drive-dokumentista. Tässä projektissa tietojen hakemiseen käytetään Zapier-automaatiotyökalua, joka yhdistää kaksi tai useampia sovelluksia – tässä tapauksessa Google Drive -tiedostohallinnan ja WP All Import -lisäosan. Tämän avulla tiedot saadaan vaivatta ja ajastetusti noudettua TVA Pori ry:n Drive-tiedostoista.

Ilmoittautumistietojen kohdalla järjestelmässä on tarve viedä tietoja myös järjestelmästä ulospäin. Tässä yhteydessä on käytetty myös Zapier-työkalua yhdistäen All Export -lisäosan ja Google Drive –tiedostohallinnan, jolloin järjestelmästä saadaan tarvittaessa vietyä ilmoittautumisjakson ilmoittautumislistauksen TVA Pori ry:n Drive-tiedostohallintaan.

5.3.2 Kurssilistaus

Kurssilistauksen näkyminen ilmoittautumisjärjestelmässä päädyttiin toteuttamaan lohkon avulla, mikä mahdollistaa listauksen lisäämisen myös muulle sivulle kuin pääsivulle sekä sivulla näytettävän ilmoittautumisjakson valitsemisen sivunäkymästä.

Lohkolle on asetettu mukautettu sisältökenttä määrittelemään, minkä ilmoittautumisjakson kurssit lohkoissa näytetään. Tämä kenttä on tyypiltään relationaalinen ja viittaa ilmoittautumisjaksosisältötyyppiin.

Kurssilistauksen kurssit haetaan lohkoon valitusta ilmoittautumisjakso-sisältötyypistä, jonne kurssit on lisätty käyttäen Advanced Custom Fields -lisäosan toista rivejä -ominaisuutta. Kurssien esittämisessä on käytetty Masonrya. Masonry on JavaScript-kirjasto, joka sijoittelee elementit optimaaliseen asentoon hyödyntäen käytettävissä olevan tilan annettujen asetusten mukaisesti (Masonry 2019).

Kurssielementin toimintoja ja tarkistuksia on toteutettu functions.php-tiedostoon sijoitettujen apufunktioiden avulla ja niitä kutsutaan lohkoista.

Yksittäisessä kurssielementissä on kurssin nimen lisäksi näkyvillä tiedot kouluttajista, viikkotunnista tai kurssipäivistä, paikasta, kenelle ilmoittautuminen on sillä hetkellä avoinna, kurssiselosteesta sekä ilmoittautuneiden määrästä. Lisäksi kurssielementti sisältää linkin ilmoittautumislomakkeelle, mikäli kyseisellä hetkellä ilmoittautuminen on avoinna.

Kurssielementin kouluttajatiedoissa kouluttajien nimet on linkitetty tarkempiin kouluttajatietoihin. Linkki ei kuitenkaan ole hyperlink, vaan sisältää JavaScript-

toiminnallisuuden avaten kouluttajan tiedot näkymän ylle modal-ikkunaan. Kurssiseloste toimii samoin, eli ilmoittautumisjakson kurssielementtiin relationaalisesti liittyvän kurssin tarkempi kuvaus avautuu modal-ikkunaan.

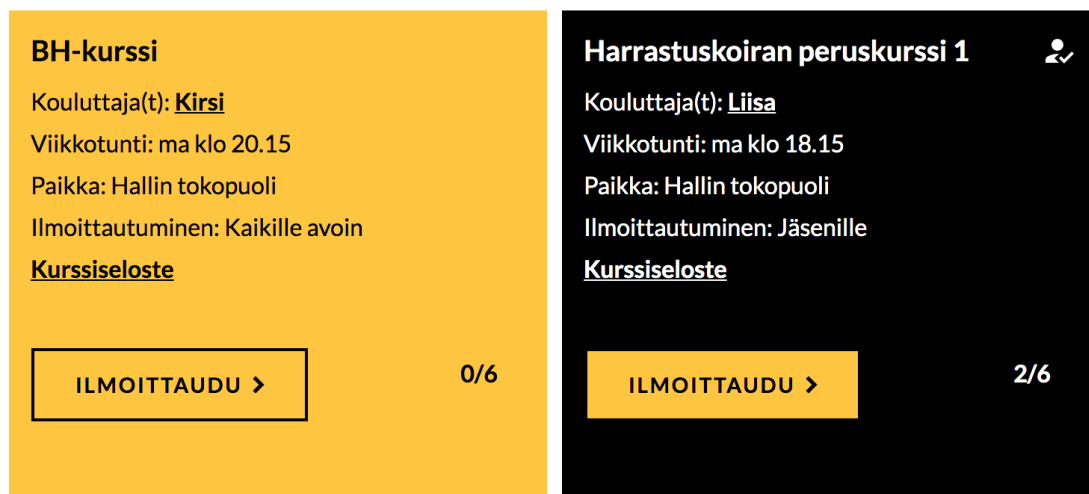
Kurssin ajan ja paikan tietojen näkyminen liittyy siihen, onko kurssille asetettu ilmoittautumisjakson kurssin tiedoissa säännöllinen viikkotunti vai muu kurssiaika, esimerkiksi tietylle viikonlopulle. Mikäli kurssilla on säännöllinen viikkotunti, kurssielementillä näkyy viikkotunnin aika tekstimuotoisen esitystavan lisäksi metatiedoissa numeraalisena määreenä, jonka avulla kurssielementit voidaan järjestää viikkotunnin mukaisesti. Tarkempi kuvaus käyttöliittymässä käytetystä lajittelusta on kappaleessa 5.5.2.1.

Kurssielementin ilmoittaudu-painike lisätään elementtiin vain, jos ilmoittautuminen on avoinna tai ilmoittautumisen alkamiseen on vähemmän kuin 10 minuuttia. Mikäli käynnissä on ilmoittautuminen vain kouluttajille ja hallituksen jäsenille, lisätään painikkeen alapuolella huomautusteksti asiasta. Painikkeen näkymiseen vaikuttaa myös se, onko kurssi jo täynnä. Täynnä olevalle kurssille painikkeen tilalla näkyy tietoa kurssin täyttymisestä.

Kurssielementissä on näkyvissä tieto siitä, montako paikkaa kurssilla on vapaana. Tieto vapaista paikoista sekä kurssin suurin sallittu osallistujamäärä näytetään ilmoittautumispainikkeen vieressä. Apufunktion `is_course_full()` avulla tarkistetaan, onko kurssi täyttynyt.

Kurssielementin yhteyteen lisätään myös ikoni jäsenyysvaatimuksesta, mikäli kurssilla tai tulevalla ilmoittautumishetkellä vaatii jäsenyyden. Jäsenyysvaatimus tarkistetaan apufunktion `get_membership_requirement_type_of_course()` avulla, joka ottaa huomioon ilmoittautumisjakson kurssiin liitetyn kurssi-sisältötyypin tyyppin (perusvai ekstrakurssi) yhdistettynä ilmoittautumisjakson vaiheeseen.

Kuvassa 18 on näkyvissä kaksi kurssielementtiä, joissa ilmoittautuminen on avoinna. Vasemman puoleiselle kurssille ei tarvita ilmoittautumiseen jäsenyyttä, mutta oikealla olevalle kurssi tarvitaan, eli sen yläkulmaan on lisätty jäsenyysvaatimuksesta kertova ikoni.



Kuva 18. Kurssielementit, joiden ilmoittautuminen on avoinna

5.3.3 Ilmoittautuminen

Kurssilistauksen ilmoittautumislinkin kautta tulostuu näkymä, jossa on kurssin ilmoittautumislomake. Linkin kautta saadaan parametrien avulla lomakesivulle tieto siitä, mikä ilmoittautumisjakso ja mikä kurssi on kyseessä, jotta ilmoittautuminen voidaan kohdistaa oikeaan kurssiin,

Ilmoittautumisnäkymään saavuttaessa tarkastetaan ehdot sille, voidaanko lomake näyttää. Apufunktiossa tarkastettavia ehtoja ovat kurssin osallistujamäärä eli se, onko kurssi täynnä, ja se, onko kurssille ilmoittautuminen avoinna tai avautumassa 10 minuutin sisällä. Vaikka nämä ehdot tarkastetaan jo kurssilistausnäkyvässä ilmoittautumispainikkeen näkymisen asetuksen yhteydessä, täytyy tilanne katsoa uudelleen lomaketta avattaessa, sillä kurssilistauksen lataamishetken jälkeen on saattanut kurssi tulla täyteen tai ilmoittautumisaika päättyä.

Lomakkeessa näkyviin kenttiin vaikuttaa myös se, onko ilmoittautumisvaihe vain kouluttajille ja hallituksen jäsenille vai normaalisti kaikille avoinna oleva. Mikäli kyseessä on kouluttajien ja hallituksen jäsenten etukäteisilmoittautuminen, lomakkeella on näkyvissä valintakenttä, jossa ilmoittautuja valitsee, onko ilmoittautuminen kouluttajan vai hallituksen jäsenen roolissa. Tämä vaikuttaa siihen, tarkistetaanko ilmoittautumisen rooli eli jäsennumero kouluttajalistauksesta vai hallitusjäsenlistauksesta.

Valinnalla on myös toinenkin merkitys, eli lisääkö ilmoittautumisessa ilmoittautumistietoihin viitenumero vai ei.

Lomakkeen peruspohja on luotu lomakelisäosan avulla, jolloin lomakekenttien hallinta hoituu WordPressin käyttöliittymästä. Lomakkeen kenttien näkymistä saadaan muokattua kurssikohtaisesti sekä lisättyä tarvittavat arvot kenttiin suodattamalla lomake ensin functions.php-tiedostoon lisätyn suodattimen (filter) avulla ennen lomakkeen lisäämistä sivulle. Kuvassa 19 on esimerkki siitä, miten ilmoittautumislomakkeen tapauksessa lomakkeelle kerrotaan tarvittavat tiedot, kuten jakson ja kurssin id sekä kouluttajavalinnan näkyminen.

```
function filter_ninja_forms_render_default_value( $default_value, $field_type, $settings ) {
    if($settings['key'] == 'cid_1571680442050') $default_value = $_GET['cid'];
    if($settings['key'] == 'rtid_1571679484455') $default_value = $_GET['rtid'];

    if(get_current_registrationperiod($_GET['rtid']) == '1') {
        if($settings['key'] == 'show_trainers_or_admins_choise_1571926949243') $default_value = '1';
    }
    return $default_value;
};

add_filter( 'ninja_forms_render_default_value', 'filter_ninja_forms_render_default_value', 10, 3 );
```

Kuva 19. Lomakekenttien arvojen muokkaaminen suodattimen avulla

Lomaketta lähetettäessä tarkistetaan pakolliseksi asetetut kentät, eli lomake ei lähde käsittelijälle ilman tarvittavia tietoja. Lomakkeen lähetyksen tunnistava koukku on lisätty functions.php-tiedostoon ja se aktivoi ilmoittautumistapahtumaa varten luodun käsittelijäfunktion, joka onnistuneen ilmoittautumisen tapauksessa luo ilmoittautumisen. Lomaketyökalun käytössä on myös se etu, että käsittelyn jälkeen voidaan toiminto palauttaa lisäosan puolelle, jonka kautta lähetetään ilmoittautujalle vahvistussähköposti. Tällöin sähköpostien sisältö voidaan tarvittaessa päivittää myös WordPressin käyttöliittymän kautta.

Vaikka lomake aukeaa täytettäväksi jo 10 minuuttia ennen ilmoittautumisen alkamista, lomaketta ei voi lähettää ennen sitä, sillä lomakkeen lähetyksen painike on ajastettu toimimaan vasta ilmoittautumisen alkaessa.

Lomaketta käsittelevästä funktiosta muodostui lopulta aika laaja, sillä tarkastettavia asioita on paljon. Funktio toimii niin, että mikäli yksikin tarkistettava ehto palauttaa

arvoksi arvon epätosi, tarkistus päätetään, ilmoittautumista ei tallenneta ja ilmoittautujalle kerrotaan ilmoittautumisen epäonnistumisesta sekä syy epäonnistumiseen.

Ensimmäisenä käsittelijäfunktiossa tarkastetaan, että ilmoittautumisjakso ja jakson kurssi ovat olemassa. Tällä vältetään tilanne, että kurssi on juuri poistettu järjestelmästä. Tämän jälkeen tarkastetaan, mikä ilmoittautumisen vaihe on menossa. Tämän vaiheen tiedolla voidaan poissulkea tilanteet, että ilmoittautumisaika on ehtinyt loppua tai ei ole vielä edes alkanut. Vaikka lomakkeen voi täyttää ennen ilmoittautumisen alkua, mutta lähetyispainikkeen toiminta on ajastettu JavaScriptin avulla, on hyvä tässä käsittelijässä varmistaa, ettei ilmoittautuja ole muokannut selaimen kautta JavaScript-asetuksia saaden lähetyispainikkeen toimintaan ja siten yrittää lähettää lomaketta etukäteen.

Ilmoittautumisvaiheen tarkastuksessa saadaan samalla tieto siitä, on ilmoittautumisen vaihe kouluttajien ja hallituksen jäsenten ilmoittautuminen, normaali ilmoittautuminen vai kaikille avoin ilmoittautuminen. Mikäli on käynnissä kouluttajien ja hallituslaisten ilmoittautuminen, tarkastetaan jäsennumeron löytyminen kouluttaja- tai hallitusjäsenlistasta eli näiden sisältötyyppien jäsennumerokentistä. Jos on normaali ilmoittautumisen vaihe kyseessä ja kurssi vaatii ilmoittautujalta jäsenyyden, varmistetaan ilmoittautujan jäsenyys jäsennumeron perusteella jäsennumerot-sisältötyypistä.

Edellä käytyjen tarkistusten myötä on saatu poissuljettua ja monta ilmoittautumisen epäonnistumisen mahdollistavaa tilannetta. Yksi tarkistuksen vaativa tekijä on vielä tarkastaa, että kurssilla on kurssipaikkoja vapaana. Mikäli tämäkin ehto toteutuu, funktio siirtyy ilmoittautumistietojen tallentamiseen ja luo uuden ilmoittautumisrivin ilmoittautumiset-sisältötyyppiin tallentaen kaikki ilmoittautumislomakkeella annetut tiedot sisältötyypin kenttiin sekä kohdistuen ilmoittautumisen relationaalisesti oikeaan ilmoittautumisjaksoon ja kurssiin.

Mikäli kyseessä ei ollut kouluttajan ilmoittautuminen kouluttajien ja hallituslaisten ilmoittautumisvaiheessa, funktio noutaa viitenumerot-sisältötyypistä vapaana olevan viitenumeron ja lisää sen ilmoittautumistietoihin. Tämän noudetun viitenumeron sisältötyyppiin muutetaan käytössä-kentän arvoksi tosi.

Kun funktio on suorittanut kaikki toimenpiteensä, palauttaa se ilmoittautujalle lomakkeen tilalle viestin onnistuneesta ilmoittautumisesta. Taustalla järjestelmässä lomakkeenkäsittelyn suoritus palautuu lisäosalle, joka lähettää vahvistusviestin ilmoittautujan antamaan sähköpostiosoitteeseen.

Tämän opinnäytetyön kappaleessa 4.4.3. pohdittu ja liitteessä 1 kuvattu suunnitelma tarkistustoimenpiteiden järjestyksestä ei aivan toteutunut, sillä toteutusvaiheessa usein vasta huomaa vaikuttavia seikkoja, joiden perusteella järjestys kannattaa muuttaa. Tässä funktiossa päädyin tarkastamaan ensin ilmoittautumisen vaiheen ennen kurssin vapaana olevia paikkoja, sillä ilmoittautumisvaiheen tarkastuksen myötä sai esille monta muuhin tarkastuksiin vaikuttavaa asiaa ja näin apufunktioiden kutsumisen määrä pieneni.

5.4 Tietokannat

Tietokantaan varastoidaan kaikki sivuston tietosisältö. Wordpress käyttää MySQL-tietokantaa ja sen tietokantarakenne on suunniteltu erittäin minimaaliseksi mahdollistaen kuitenkin loputtoman joustavuuden WordPress-ohjelmistoja kehitettäessä ja suunniteltaessa. (Williams ym. 2014, 111–112)

Wordpressin tietokanta koostuu pääsääntöisesti 12 tietokantataulusta. Taulukossa 4 on listattu Wordpressin tietokantataulut ja niiden pääsisältö pääpiirteissään.

Tietokantataulun nimi	Sisältö
wp_commentmeta	Kommenttien metatiedot
wp_comments	Kommentit
wp_links	Linkkiviittaukset (vanhentunut)
wp_options	Sivuston asetukset
wp_postmeta	Artikkelien (post) metadata
wp_post	Artikkelit (post)
wp_terms	Kategoriat
wp_termmeta	Kategorioiden metadata
wp_term_relationships	Artikkelien (post) ja kategorioiden assosiaatiot

wp_taxonomy	Taksonomiat
wp_usermeta	Käyttäjien metatiedot
wp_users	Käyttäjät

Taulukko 4. Wordpressin tietokantataulut (Wordpress 2019g)

Wordpressissä mukautetut tietokantakyselyt voi tehdä esimerkiksi Wordpressin ytimessä määritellyn luokan WP_Query avulla. Se mahdollistaa hakujen rajaamisen sisältökenttien arvojen mukaan sekä tulosten lajittelun annettujen sisältökenttien mukaisesti. (Williams ym. 2014, 88–90)

Ilmoittautumisjärjestelmässä on käytetty ilmoittautumisen tarkistusfunktiossa tietojen noutamiseen WP_Queryn avulla tehtyjä sisältökyselyjä. Kuvassa 20 on esimerkki hallituksen jäsenen jäsennumeron tarkastamisesta. Haussa on annettu rajaaviksi tekijöiksi sisältötyyppi ja jäsennumerokentän sisältö. Mikäli haku tuottaa tuloksia, kertoo se funktiolle sen, että jäsennumerolla löytyy hallituksen jäsen tietokannasta.

```

$args = array(
    'numberposts' => -1,
    'post_type' => 'hallitusjasenet',
    'meta_key' => 'jasennumero',
    'meta_value' => $member_nbr
);
$query = new WP_Query( $args );
if( $query->have_posts() ) { $check_memberid = true; }
else {
    $reg = 0;
    $err = "Jäsennumeroa ei löytynyt hallitustiedoista.";
}

```

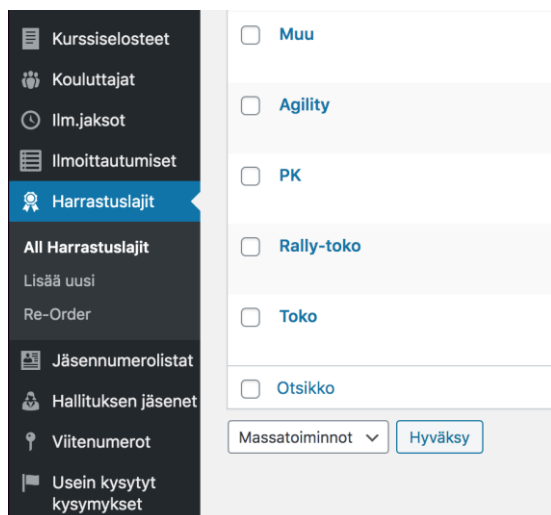
Kuva 20. Esimerkki WP_Queryn käytöstä ilmoittautumisjärjestelmässä

5.5 Käyttöliittymät ja ulkoasu

Sivuston ulkoasussa haluttiin käyttää TVA Pori ry:n päävärejä. Koska seuran kurssi-kaavioissa on totuttu merkitsemään peruskurssit mustalla ja ekstrakurssit keltaisella pohjalla, päädyttiin tässä ilmoittautumisjärjestelmässä noudattamaan samaa käyttäjille jo tutuksi tullutta logiikkaa helpottamaan kurssien silmäilyä.

5.5.1 Hallintapuolen näkymät

Ilmoittautumisjärjestelmän hallinnassa käytetään Wordpressin omaa käyttöliittymää, jossa mukautetut sisältötyypit listautuvat vasemman puolen navigaatioon. Mukautetuille sisältötyypeille on lisätty myös löytymistä helpottavat ikonit. Kuvassa 21 näkyy esimerkkinä valikossa mukautettu sisältötyyppi Harrastuslajit.



Kuva 21. Mukautettujen sisältötyyppien näkyminen Wordpressin hallinnassa

Mukautettujen sisältötyyppien näkymässä on kuitenkin tehty pieniä tyylimuutoksia, jotka helpottavat kokonaisuuksien hahmottamista. Esimerkki tästä on ilmoittautumisjaksoon lisättävät kurssit, joissa on hallintapuolella taustaväreillä eroteltu kulloinkin esillä oleva kurssi. Hallintanäkymän tyylejä voidaan muokata luomalla teeman functions.php-tiedostoon toimintokutsu, joka lisää tarvittavat tyylimäärytykset haluttuun elementtiin.

5.5.2 Julkisen puolen näkymät

Kurssilistaus ilmoittautumisjärjestelmässä tuodaan jo heti pääsivulle. Pääsivun alkuun lisättiin vain tärkein ohjeistus, linkki tarkemmalle ohjesivulle sekä kurssikaaviokartta, jotta tärkeimmät tiedot ovat alussa helposti saatavilla. Tämän alle latautuu kurssilistaus. Kuvassa 22 on valmiin järjestelmän näkymä pääsivusta. Kuvassa oikealla ylhäällä näkyvä on seuran kurssikaavio, joka auttaa ilmoittautujia valitsemaan sopivan kurssitason. Tätä kaaviokuvaa klikkaamalla kuva avautuu modal-ikkunaan. Modal-

ikkuna on graafinen elementti, joka avautuu pääsisällön päälle ja yleensä tummentaen taustan (Bootstrap 2019b). Tämän etuna on se, että sivua ei tarvitse ladata uudelleen.

TVA Pori Ilmoittautuminen

Ilmoittaudu jaksolle X/2020

Jakson XXX ilmoittautuminen alkaa **maanantaina XXX**. Kouluttajien etukäteisilmoittautuminen kurssille on torstaina XXX. Ilmoittautumisen ohjeistuksen ja usein kysytyt kysymykset löydät **ohjesivulta**.

Huom! Ilmoittautuslomakkeet aukeavat täytettäväksi jo 10 minuuttia ennen ilmoittautumisen alkamista. Lomaketta ei voi kuitenkaan lähettää etukäteen.

Näytä lajin mukaan

Kaikki Perusteet Toko Rally-toko PK Agility Muut

Järjestä

Oletus Viikotunin mukaan

Tällä ikonilla merkityille kurssille tarvitset ilmoittautuaksesi TVA Pori ry:n jäsenyyden.

NÄYTÄ VAIN NE KURSSIT, JOILLE EI VAADITA JÄSENYTYTTÄ

Ilmjärjestelmän koulutus	Alkeiskurssi 1	Alkeiskurssi 2
Kouluttaja(t): Marika M.	Kouluttaja(t): Fanni	Kouluttaja(t): Anne
Kurssipäivä(t): La 23.11.	Viikotunti: ma klo 18.15	Viikotunti: la klo 12.00
Paikka: Kurrenkuulma	Paikka: Hallin tokopuoli	Paikka: Hallin tokopuoli
Ilmoittautuminen: Kaikille avoin	Ilmoittautuminen: Kaikille avoin	Ilmoittautuminen: Kaikille avoin
Kurssiseloste	Kurssiseloste	Kurssiseloste

Kuva 22. Ilmoittautumisjärjestelmän pääsivun näkymä

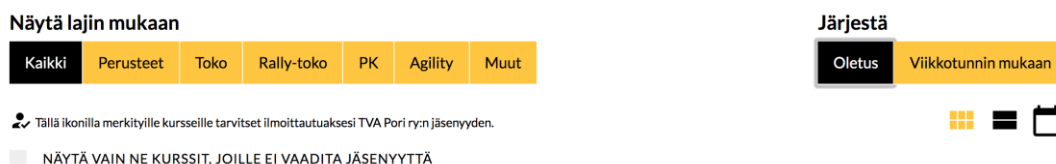
Kurssilistauksessa kurseista näkyy tärkeimmät tiedot, eli nimi, kouluttaja, aika, paikka ja ilmoittautuneiden määrä sekä ilmoittautumisajan ollessa käynnissä linkki ilmoittautumiseen. Mikäli kurssi on täynnä, kurssinäkymän värit ovat osittain läpinäkyvät ja kurssin tiedoissa on merkintä kurssin täyttymisestä.

Kurssinäkymän kouluttaja- sekä kurssiselostelinkit avautuvat modal-ikkunaan, johon haetaan sisältö mukautetuista sisältötyypeistä. Kouluttajanäkymässä näkyy kouluttajan nimi, yhteystiedot, pätevyudet, koulutettavat lajit, kuva sekä esittelyteksti. Kurssilistausnäkymän avulla voi tarkastaa halutun kurssin tarkemman kurssikuvauksen.

Ohjesivulla sisältöä on paljon, joten oli järkevää jäsenellä sisältö niin, että ilmoittautuja löytäisi vastauksen etsimäänsä kysymykseen mahdollisimman helpolla. Ohjesivun usein kysytyt kysymykset –osio päädyttiin toteuttamaan haitarirakenteeksi kutsutulla tyylillä, jossa oletuksena näkyvillä on vain kysymykset ja vastaukset aukeavat kysymysten alle kysymystä klikattaessa.

5.5.2.1 Lajitteluvalinnat

Kurssien lajitteluun kurssilistausnäkymässä tehtiin lajittelu harrastuslajin mukaan sekä erillisellä valinnalla vielä valinta siitä, näytetäänkö vain ne kurssit, joille ilmoittautumiseen vaaditaan seuran jäsenyys. Nämä valinnat piilottavat näkyvistä ylimääräiset kurssit ja jättävät näkyviin vain valintakriteerit täyttävät kurssit. Näiden lisäksi näkymään luotiin järjestysvalinta, jolla kurssit voi järjestää viikkotunnin mukaan. Tehtäessä lajittelu- tai suodatusvalinta kurssit päivittyvät automaattisesti näkymässä heti valinnan jälkeen käyttäen Masonry-kirjastoa. Kuvassa 23 on kurssilistauksen lajittelun ja järjestämisen valintapainikkeet. Painikkeissa on mustalla taustavärillä korostettu valittu vaihtoehto.



Kuva 23. Kurssilistauksen lajitteluvalinnat

5.5.2.2 Listausnäkyvät

Pääsivulla lajitteluvalintojen yhteydessä oikealla on myös ikonit, joiden kautta saa valittua kurssilistauksen esitysmuodon. Oletusasetuksena on laatikkorakenne, jossa laatikot asettuvat käytettävissä olevan tilan mukaisesti enintään kolme laatikkoa rinnakkain. Toinen listausmuoto on perinteisempää listausta muistuttava esitystapa, jossa kurssielementit ovat täysileveinä riveinä. Kuvassa 24 on verrattu näiden kahden edellä mainitun esitystavan näkymäeroa.

The image shows a user interface for selecting courses. It is divided into two main sections: a grid view on the left and a list view on the right. Both sections have a filter bar at the top with categories like 'Kaikki', 'Perusteet', 'Toko', 'Rally-toko', 'PK', 'Agility', and 'Muut'. Below the filter bar, there are instructions and a checkbox to 'NÄYTÄ VAIN NE KURSSIT, JOILLE EI VAADITA JÄSENYTYÄ'. The grid view displays course cards for 'Ilmijärjestelmän koulutus', 'Alkeiskursseja 1', 'Alkeiskursseja 2', 'Harrastuskoiran peruskursseja 1', 'Harrastuskoiran peruskursseja 2', and 'TOKO alo 1'. Each card shows the course name, instructor, time, and enrollment status (e.g., '2/12', '3/8', '1/8'). The list view displays a table of courses with columns for course name, instructor, time, and enrollment status. The table includes courses like 'Ilmijärjestelmän koulutus', 'Alkeiskursseja 1', 'Alkeiskursseja 2', 'Harrastuskoiran peruskursseja 1', 'Harrastuskoiran peruskursseja 2', 'TOKO alo 1', 'TOKO alo 2', 'TOKO alo/avo 1', 'TOKO alo/avo 2', and 'TOKO alo/avo 3'.

Kuva 24. Kurssilistauksen laatikko- ja rivinäköymien vertailu

Kolmas listausmuoto on lukujärjestysnäköymä, jossa kurssit listautuvat viikkotunnin mukaisesti lukujärjestykseen. Koska kaikilla kursseilla ei välttämättä ole säännöllistä viikkotuntia, on lukujärjestyksen yläpuolella huomautus siitä, että lukujärjestyksessä näkyy vain ne kurssit, joilla on säännöllinen viikkotunti. Lukujärjestysnäköymässä lukujärjestyksen laatikkoa klikkaamalla aukeaa kurssin lisätiedot modal-ikkunaan.

Lukujärjestyksen toteuttaminen aiheutti paljon haasteita, sillä samalla hetkellä voi olla jopa kolme eri kurssia, jotka eivät välttämättä ala samaan kellonaikaan, vaan jokin niiden osa on samaan aikaan. Tämän muodostamiseksi käytin paljon aikaa ja erilaisia koodi- ja laskentaratkaisuja löytääkseni parhaan esitystavan. Kuvassa 25 on lukujärjestysnäköymä esitettynä. Haasteena näköymässä on sen runsas tilantarve ja se, että kurssit painottuvat arki-iltoihin ja viikonloppuina aamupäiviin, joten aikaskaalaa ei voi tiivistää.

KLO	MA	TI	KE	TO	PE	LA
9-10						
10-11						TOKO voivai
11-12						
12-13						Aikeiskursi 2
13-14				Testikursi		
14-15						
15-16						
16-17						
17-18		TOKO alo 1				
18-19	Agility alkeisjalko	Agilityn alkeet 1	PK-tottis 1	TOKO alo/avo 2	RT avo/voi	Lapsi ja koira
19-20	Aikeiskursi 1	Harrastustokoran peruskursi 1	Harrastustokoran peruskursi 2	Agility jatko	RT voivai	RT alo/avo 2
20-21	TOKO alo/avo 1	Agility kisaavat 1-ik	RT alo	Agility alkeet 2	RT voivai 1	RT voivai 2
21-22	BH-kursi	Agility kisaavat 2-ik ja 3-ik	Ajätoko	TOKO avo/voi	TOKO alo/avo 3	TOKO alo/avo 4
				PK-tottis 2		TOKO alo 2

Kuva 25. Kurssit esitettynä lukujärjestysnäkyssä

Lukujärjestysnäkyksen etuina on se, että valittaessa vain tietty laji, näkee helposti sen lajin kurssien sijoittumisen aikatauluun. TVA Pori ry:llä on aina ollut tapana työstää kuvamuotoinen näky toko- ja rally-tokokursseista ja siksi päätin toteuttaa lukujärjestyksen yhdeksi näkymävalinnaksi. Määrittelyssä tilaajan kanssa ei ole määritelty erilaisia esitystapoja, joten esitystapojen toteuttaminen on ollut oma lisäni toteutukseen ja se vaikutti valmiina myös tilaajasta hyvältä idealta.

5.5.2.3 Ilmoittautuminen

Ilmoittautumislomake aukeaa pääsivun sisällä kurssilistauksen tilalle. Kuvassa 26 on ilmoittautumislomakkeen näky, jossa kurssin tarkemmat tiedot kurssiselostukseen on vasemmassa palstassa ja varsinainen lomake oikeassa palstassa.

Ilmoittautuminen valitulle kursille

Kurssi: Harrastuskoira peruskursi 1
 Kouluttaja(t): Liisa
 Aika: maanantaisin klo 18:15
 Paikka: Hallin kokousoli

Ilmoittautuminen: Tilille

Kurssiteksti:
 Kurssi on perustettu tökö-, rallykoira-, PK- ja agilitykoirille. Peruskurssilla keskitytään harrastuskoiran peruslajiin, kuten esimerkiksi erilisiin puikkuharjoitukseen, hurrintaan, perusasettiin ja loikkaamiseen. Kurssiin voi uusia tarjottamaa. Kurssi olisi hyvä olla käytynä ennen jatkamista tökö-, rallykoira-, PK- ja agilitykurssille. Harrastuskoiran peruskurssin kesto on 8 treenikertaa, hinta on 30 euroa /kurssi. Kurssille ilmoittautuminen edellyttää jäsenyyttä.

OHJAJAN TIEDOT

ETUNIMI *

SUKUNIMI *

JÄSENNUMERO *

PUHELIN *

SÄHKÖPOSTI *

KOIRAN TIEDOT

KOIRAN KUTSUMANIMI *

KOIRAN IKÄ *

KOIRAN ROTU *

MAHDOLLISET LISÄTIEDOT

TVA-Päri ry:n kurssille ilmoittautuminen on lähtökäytännöllisesti sitova. Tarkemmin [perustusehdot](#).

HYVÄKSYN EHDOT *

LÄHETÄ ILMOITTAUTUMINEN >

Kuva 26. Ilmoittautumislomakkeen näkymä

Mikäli ilmoittautumisessa on käynnissä kouluttajien ja hallituksen jäsenten etukäteis-ilmoittautuminen, tulee lomakkeen yläpuolelle huomautus asiasta sekä tällöin lomakkeella on ylimääräinen kenttä kouluttaja- tai hallitusjäsenroolin valintaan.

5.5.3 Responsiivisuus

Responsiivisuus tarkoittaa sivuston mukautuvuutta erilaisille päätelaitteille, jolloin käytettävyys paranee. Responsiivisuus on nykyään erittäin tärkeä osa sivustojen toteutusta, sillä sivustojen käyttö erilaisilla mobiililaitteilla lisääntyy vuosittain.

Ilmoittautumisjärjestelmässä responsiivisuus on otettu huomioon toteutuksessa ja kaikki sisältönäkymät on tehty käytännöllisiksi laitteen näyttökoosta riippumatta. Responsiivisuuden mahdollistamisessa suurena apuna on ollut Bootstrap, jonka luokkien avulla voidaan muodostaa sarakemainen jaottelu valiten, miten elementit jakautuvat sarakkeiden suhteen kussakin näyttökoossa.

5.6 Testaus

Järjestelmän yksikkötestausta on tapahtunut koko toteutusprosessin ajan jokaisen yksittäisen komponentin ja funktion kohdalla. Yksikkötestauksella saadaan heti poimitua mahdolliset virheet ja korjattua ne.

Järjestelmätestaukseen tässä projektissa on haluttu panostaa paljon, jotta järjestelmä olisi mahdollisimman toimintavarma käytössä. Kurssilistausten toimivuudesta, kurssien ja niiden lisätietojen etsimisestä sekä ilmoittautumistapahtumasta järjestettiin tilaajan testiryhmän kanssa testaushetki jo ennen, kuin koko järjestelmä oli kokonaan valmis. Näiden testauksessa esiin tulleiden huomioiden pohjalta tehtiin järjestelmän käyttöliittymään huomiota ja saatiin ilmoittautumisen poikkeustilanteiden virheellinen toiminta korjattua.

Järjestelmää on aikaisessa vaiheessa jo testikäyttänyt ulkopuolinen testihenkilö, joka ei tunne kurssijärjestelmää. Hänelle annettiin tehtävä etsiä tietyn tyyppinen kurssi ja ilmoittautua sille. Tämän tapaisella testauksella nähtiin, miten henkilö, jolle järjestelmä on vieras ja uusi, osaa käyttää lajittelutoimintoja ja löytää tarvittavat ohjeet ilmoittautumiseen.

Järjestelmää on testattu myös erilaisten päätelaitteiden kanssa sekä erilaisilla selaimilla. Erityisen haasteen toi Internet Explorer –selaimet, joissa jotkin kurssinäkömät eivät heti toimineet.

5.7 Käyttöönotto

Järjestelmän hallintapuolen käytöstä pidetään käyttökoulutus kurssivastaaville. Koska kyseessä on yhdistys, jonka toimihenkilöt saattavat vaihtua, on tarpeellista liittää hallintapuolella myös ohjedokumentaatio erilaisten toimintojen käytöstä.

6 TULOSTEN ARVIOINTI

6.1 Loppukäyttäjien kommentit

Ilmoittautumisjärjestelmää ei ole vielä julkaistu tämän opinnäytetyön kirjoittamisen aikana, mutta sitä on testannut laaja joukko, joka tulee olemaan myös järjestelmän loppukäyttäjiä tulevaisuudessa – niin ilmoittautujan kuin kurssivastaavan roolissa.

Testaajat ovat olleet hyvin tyytyväisiä järjestelmään ja sen muodostettuihin toiminnallisuuksiin käyttöliittymän suhteen. Määrittelyssä ilmoittautumisjärjestelmä määriteltiin vain tarvittavien ominaisuuksien mukaisesti. Ylimääräisenä toteutetut kurssinäkyvät, kouluttaja- ja kurssiesittelyt ovat saaneet positiivisen vastaanoton. Kiitosta sai myös ulkoasu, joka vastaa nyt vanhaa järjestelmää paremmin TVA Pori ry:n graafista ilmettä.

Ilmoittautumisjärjestelmä on tarkoitus julkaista käyttöön vuoden 2020 alussa ja silloin järjestelmä pääsee suuren joukon testiin ensimmäisessä ilmoittautumisjaksossa. Mikäli silloin ilmenee käytön suhteen haasteita, pyritään tarkastelemaan, voidaanko ne ratkaista helposti.

6.2 Projektinhallinta

Ilmoittautumisjärjestelmän määrittely ja suunnittelu alkoi jo keväällä 2019, jolloin tilaajan kanssa käytiin läpi sitä, millainen uusi ilmoittautumisjärjestelmä voisi olla. Varsinainen toteutus ja testaus tehtiin syksyn 2019 aikana. Projektille ei oltu määritelty ehdotonta aikataulua, mikä mahdollisti sen, että pystyin ilman aikataulupaineita toteuttamaan siihen laajempia näkymäratkaisuja.

Oma aikataulutukseni projektille oli se, että toteutus on valmis vuoden 2019 loppuun mennessä. Tilaajan kanssa päädyttiin loppuvuodesta siihen, että järjestelmän käyttöönotto tapahtuu kuitenkin vasta keväällä 2020, jotta löytyy varmasti aikaa perehdyttää kurssivastaavat järjestelmän käyttöön.

7 LOPPUSANAT

Ilmoittautumisjärjestelmän uudistaminen projektina oli minulle hyvin mieluisa sekä aihealueen puolesta että myös siksi, että tässä projektissa minulla oli mahdollisuus suunnitella ja toteuttaa lisäominaisuuksia järjestelmälle parhaaksi katsomallani tavalla ja näin kehittää omaa teknistä osaamista.

Koska ilmoittautumisjärjestelmää laajennettiin lisäominaisuuksilla, esimerkiksi kurssitietojen laajennuksilla, muodostui tästä ilmoittautumisjärjestelmän lisäksi myös eräänlainen kurssijärjestelmä, jota seuran jäsenet voivat käyttää suunnitellessaan omia kurssipolkujaan. Järjestelmä toimii myös kurssivastaavilla työkaluna uusien kurssijaksojen suunnitteluun.

Olen tyytyväinen lopputulokseen, vaikka projektin matkalla oli myös hankalia vaiheita. Lukujärjestyksen toteuttaminen oli työläs ja vei paljon aikaa, mutta antoi myös paljon tietoa minulle ja kehitti minua osaamisessani.

Projektin aikana on otettu huomioon myös jatkokehitysmahdollisuudet seuran www-sivuston liittämiseksi tulevaisuudessa järjestelmän yhteyteen. Toteuttaessani järjestelmää tuli esiin monia ideoita ja suunnitelmia tämän sivuston suhteen ja toivonkin, että tilaajan kanssa voimme jollain aikavälillä sopia jatkoprojektista lisäominaisuuksien kehittämisen suhteen.

LÄHTEET

Advanced Custom Fields 2019. Viitattu 4.12.2019. https://www.advancedcustom-fields.com/resources/acf_register_block_type

Bootstrap 2019a. Viitattu 5.12.2019. <https://getbootstrap.com>

Bootstrap 2019b. Viitatti 15.12.2019. <https://getbootstrap.com/docs/4.0/components/modal/>

Digicert. 2019. What is SSL, TLS and HTTPS? Viitattu 12.12.2019. <https://www.websecurity.digicert.com/security-topics/what-is-ssl-tls-https>

Lano, K. 2009. UML 2 Semantics and Applications. John Wiley & Sons. Viitattu 1.12.2019. <http://ebookcentral.proquest.com/lib/samk/detail.action?docID=711732>

Luukkainen M. 2011. Ohjelmistojen mallintaminen. Viitattu 13.10.2019. <https://www.cs.helsinki.fi/group/java/s11/ohma/kalvot/luento1.pdf>

Luukkainen M. & Laine H. 2010. Ohjelmistojen mallintaminen. Viitattu 13.10.2019. <https://www.cs.helsinki.fi/u/mluukkai/ohmas10/ohma.pdf>

Manifesto for Agile Software Development. 2001. Viitattu 2.12.2019. <http://agilemanifesto.org>

Masonry. Viitattu 10.12.2019. <https://masonry.desandro.com/>

Nykänen P. 2014. Tietojärjestelmäsuunnittelun keskeisiä asioita, 23–27. Viitattu 5.10.2019. http://www.uta.fi/sis/tie/tjsum/index/TJSUM_Luento1_2014_PirkkoNykanen.pdf

Pressman, R. S. 2010. Software engineering: a practitioner's approach. McGraw-Hill Companies.

Seravo. 2019. Tietoturvallinen Wordpress. Viitattu 12.12.2019. <https://wp-palvelu.fi/feature/tietoturvallinen-wordpress/>

StatCounter GlobalStats. Viitattu 3.12.2019. <https://gs.statcounter.com/browser-market-share/all/finland/#monthly-201908-201911>

Stephens, R. 2015. Beginning Software Engineering. John Wiley & Sons. Viitattu 1.12.2019. <http://ebookcentral.proquest.com/lib/samk/detail.action?docID=1895174>

Tampereen Yliopiston kurssimateriaali. Viitattu 26.9.2019. https://coursepages.uta.fi/tiep1/wp-content/uploads/sites/6/2018/06/laki1_2017_luentorunko_luku_02.pdf

Thinking Portfolio 2016. Projektien vesiputousmalli ja sen 5 heikkoutta. Viitattu 13.10.2019. <https://thinkingportfolio.com/projektien-vesiputousmalli-ja-sen-viisi-heikkoutta>

Tietosuojavaltuutetun toimisto. 2019a. Tietosuoja. Viitattu 9.12.2019. <https://tietosuoja.fi/tietosuoja>

Tietosuojavaltuutetun toimisto. 2019b. Kerro käsittelystä rekisteröidylle. Viitattu 9.12.2019. <https://tietosuoja.fi/rekisteroidyn-informointi>

TVA Pori ry:n www-sivut. Viitattu 12.9.2019. <https://www.tvapori.fi/koulutukset/kurssiselosteet>

Web Technology Surveys. Viitattu 1.4.2019. <https://w3techs.com>

Williams, B., Damstra, D. & Sabin-Wilson, L. 2014. Professional WordPress: Design and Development. John Wiley & Sons, Incorporated. Viitattu 4.12.2019. <http://ebookcentral.proquest.com/lib/samk/detail.action?docID=1895867>

Williams, B., Richard, O. & Tadlock, J. 2011. Professional WordPress: Plugin Development. John Wiley & Sons, Incorporated. Viitattu 4.12.2019. <http://ebookcentral.proquest.com/lib/samk/detail.action?docID=661504>

Wordpress 2019a. Features. Viitattu 1.4.2019. <https://wordpress.org/about/features>

Wordpress 2019b. Requirements. Viitattu 1.4.2019. <https://wordpress.org/about/requirements>

Wordpress 2019c. Post types. Viitattu 3.10.2019. <https://wordpress.org/support/article/post-types>

Wordpress 2019d. Plugins. Viitattu 4.12.2019. <https://wordpress.org/plugins>

Wordpress 2019e. WordPress 5.0 “Bebo”. Viitattu 4.12.2019. <https://wordpress.org/news/2018/12/bebo>

Wordpress 2019f. Blocks. Viitattu 4.12.2019. <https://wordpress.org/support/article/blocks>

Wordpress 2019g. Database Description. Viitattu 12.12.2019. https://codex.wordpress.org/Database_Description

WP-palvelun www-sivut. Viitattu 2.12.2019. <https://wp-palvelu.fi>

ILMOITTAUTUMISEN SUUNNITELTU TOIMINTAMALLI

