



LAUREA
AMMATTIKORKEAKOULU
Yhdessä enemmän

Sovellusalustan suunnittelu verkkopalvelulle

Lappalainen, Eero

2016 Laurea



Laurea-ammattikorkeakoulu

Sovellusalustan suunnittelu verkkopalvelulle

Eero Lappalainen
Tietojenkäsittelyn koulutusohjelma
Opinnäytetyö
Tammikuu, 2016

Eero Lappalainen

Sovellusalustan suunnittelu verkkopalvelulle

Vuosi 2016 Sivumäärä 30

Tämä opinnäytetyö käsittelee osaa laajan verkkopalvelun jatkokehitystyöstä. Verkkopalvelun jatkokehityskohteena on luoda sovelluksia ja sivustoa yhdistävä alusta. Opinnäytetyön vaatimuksena oli luoda suunnitelma, kuinka alusta tulisi kehittää.

Opinnäytetyö toteutettiin Linked By Music Oy:lle, joka on suomalainen verkkosovelluksien kehittäjäyritys. Asiakkaan kehityskohde, johon työ tehtiin, on muusikoille suunnattu sosiaalisen median verkkopalvelu. Palvelun tavoitteena on yhdistää musiikin harrastajia ja ammattilaisia, sekä tarjota heille lähtökohtaisesti musiikin osa-alueisiin liittyviä hyötyjä. Sovellusalustan toiminnot tulisivat olemaan Linked By Music:in sekä tämän yhteistyökumppaneiden palveluja yhdistävä toiminnallisuus. Käytännössä suunnitelman pohjalta syntyvä sovellus olisi kahden verkkosivun kokonaisuus, jolla käyttäjät voisivat valita ja käyttää eri ohjelmia verkon välityksellä.

Tässä työssä syntyi suunnitelma, joka sisälsi kolme eri kokonaisuutta, sekä raportti, joka yhdistää kokonaisuudet yhteen. Kokonaisuuksiin kuului visuaalisen ilmeen, toiminnallisten rakenteiden sekä niiden vuorovaikutuksien mallinnus. Työn vaatimukset muuttuivat osittain työn aikana, jolloin taustatutkimukset muovautuivat vain alustaviksi kehitysehdotuksiksi. Puutteita tutkimuksessa olivat yhteensopivuuden haasteet ja toteutuksen tekniikat, joilla sovellusalusta tulisi yhdistämään verkkopalveluun.

Eero Lappalainen

Designing Software Integration for a Website

Year	2016	Pages	30
------	------	-------	----

This thesis covers a part of the development of a vast website. The goals for further development is to connect offline and online software inside a website. The goal for the thesis was to create a development plan for a software integration.

The commissioner for the plan was “Linked by Music OY”, which is a Finnish website and web development company. The main product of the company is social media which connects musicians and provides music-related benefits for the users. The fully functional software integration would connect outside software inside Linked by Music website. In other words Linked by Music’s users could find and use different software inside the website.

The finished plan had three different parts, which were explained in the thesis’ report. Parts of the plan were modelling of the visual appearance, functionalities and interaction between them. The commissioner’s requirements for the work changed during the process. The research work for the compatibility and implementation between the integration and servers wasn’t included in the thesis.

Keywords Website software, plan, modelling

Sisällys

1	Johdanto.....	6
2	Tausta.....	7
2.1	Asiakas	7
2.2	Tarpeellisuus ja kohderyhmä	7
2.3	Hankkeen vaatimukset	8
2.4	Menetelmät	8
3	Suunnittelutyön teoria	9
3.1	Keskeiset käsitteet.....	10
3.2	Toteutussuunnitelman luominen	11
3.3	Toimintojen mallintaminen	12
3.4	Liitosohjelmat	12
3.4.1	HTML5	12
3.4.2	Javascript.....	12
3.4.3	React!	13
3.5	Verkkopalvelun visuaalisuus.....	13
4	Suunnittelutyö.....	13
4.1	Toiminnallisuuden käyttötarina	14
4.2	Pseudokoodi.....	15
4.2.1	Prototyyppi.....	16
4.3	Visuaalinen suunnitelma	17
4.3.1	Rautalankamallit	18
5	Arviointi ja kehitysehdotukset.....	20
6	Johtopäätökset ja yhteenveto	20
	Lähteet	22
	Taulukot	24
	Liitteet.....	25

1 Johdanto

Tämän raportin tarkoitus on toimia toiminnallisen opinnäytetyön kirjallisena osana. Työn aiheeksi valikoitui sovellusalustan suunnittelu kehitysvaiheessa olevalle verkkosivustolle. Opinnäytetyön konkreettinen toiminnallinen osa on toteutus suunnitelma. Aiheeseen valitun tuotoksen tarkoituksena on toimia sekä helposti seurattavana suunnitelmana, että viitteellisenä esityksenä siitä, miten kyseisen sovellusalustan luomisessa tulisi toimia. Lähtökohtaisesti työssä oli päästävä sellaiseen lopputulokseen, joka on ymmärrettävä ja hyödyllinen asiakkaan jatkokehitystä varten.

Sovellusalusta tulisi toimimaan osana verkkosivustoa, jonka kehittäjänä toimii Linked By Music Oy. Kehitystyö verkkosovellukselle avustaa saavuttamaan tavoitteita, joita Linked By Music on asettanut tulevaisuuden jatkokehitykselle. Sovellusalusta tulisi toimimaan yleisesti musiikin harrastamista tai luomista helpottavana tekijänä. Sisällöltään sovellusalusta voisi esimerkiksi tarjota viritinsovelluksen tai musiikinteko-ohjelman. Sovelluksien tuottamisesta tulisivat vastaamaan yhteistyökumppanit, sekä mahdollisesti palvelun käyttäjät. Sovellusalustan toiminnallinen tarkoitus on myös verkostoida asiakkaan palvelun käyttäjiä.

Opinnäytetyön vaiheina oli tuottaa tutkimuksen pohjalle perustuva toimintasuunnitelma ja mallinnus. Mallinnukseen kuuluivat osa-alueina vaadittujen verkkosivujen rautalankamallit, prototyypit sekä tietopankkiohjelmoinnin pseudokoodi. Visuaalisen ilmeen suunnittelussa lähtökohtana oli luoda malli, joka vastasi hankeorganisaation nykyisen sivuston visuaalista ilmettä. Tutkimuksessa otettiin huomioon myös sivuston rakenne, sillä yksi työn vaatimuksista oli selvittää voiko sovellusalustan toteuttaa uudella menetelmällä.

Lopullisen suunnitelman sisältöön jäi vajavaisuuksia yhteensopivuuden ja toteutuksen tekniikoiden haasteissa, sillä opiskelijan käytössä ei ollut tarpeeksi laajaa tietämystä asiakkaan nykyisen palvelun ohjelmointitekniikoista. Työssä syntyneen suunnitelman lopulliseksi käyttötarkoitukseksi jäi toimia osana jatkokehityksen suunnittelua.

2 Tausta

Tämän kappaleen tarkoitus on selventää työn lähtökohtaista tilannetta, sekä selventää työhön liittyneitä asioita. Opinnäytetyön tavoitteena on toimia opiskelijan ammatillisen asiantunte- muksen- ja taidon näytteenä. Koska hankeorganisaatiolla oli tarve verkkosovelluksen sovel- lusalustan jatkokehittämiseksi, työn aiheeksi valikoitui sovellusalustan suunnitelman työstä- minen ja perustelut suunnitelmassa esiin tuleville asioille. Koska vaatimuksissa on tehdä jokin konkreettinen asia, tässä tapauksessa suunnitelma, syntyi tavoitteeksi luoda toiminnallinen opinnäytetyö.

2.1 Asiakas

Työn asiakkaana toimi Linked By Music Oy. Linked By Music on 2010- vuoden jälkeen perustet- tu sosiaalisen median kehittäjäyritys. Linked By Music on pieni alle 10 hengen yritys, ja tuot- taa lähtökohtaisesti muusikoille suunnattua sosiaalista mediaa. Muusikoiden lisäksi musiikin alan muut osapuolet, esimerkiksi soitinkauppiat ja musiikinopettajat ovat myös Linked By Music:in potentiaalisia kohdeasiakkaita. Linked By Music:in tavoitteina on luoda vaikutusval- tainen dynaaminen verkkokokonaisuus, jonka vakituiset käyttäjät toisivat varsinaisen arvon palvelulle. Lisäksi tavoitteisiin kuuluu verkostoida musiikin harrastelijat ja luoda heille täten hyötyä palvelusta.

2.2 Tarpeellisuus ja kohderyhmä

Asiakas oli luonut tilauksen työlle ennen opinnäytetyön toteutusvaihetta. Asiakas ei pitänyt opinnäytetyön tavoitteena luoda tutkimusta sovellusalustan tarpeellisuudesta, jolloin teoret- tisuus olisi suuremmalla jalansijalla. Tutkimuksella olisi saatu selko siitä, kuinka suuri tarve sovellusalustalle tulisi olemaan liiketoiminnallisella kannalla nykyhetkellä. Työn varsinaiseksi tarpeeksi jäi suunnitelman luominen. Suunnitelmassa on kuitenkin otettava huomioon se, että tarpeellisuus hankkeelle voi muuttua myöhemmin. Asiakkaan lopullinen tilannearvio vaikuttaa varsinaisen suunnitelman hyödyntämisessä. Lisäksi myöhemmin esille tulevat yhteistyökump- panit vaikuttavat suuresti siihen, miten suunnitelmaa kannattaa hyödyntää.

Suunnitelman toteuttamiseen liittyy vahvasti tieto siitä, millainen kohderyhmä tulee valmista palvelua käyttämään. Kun puhutaan jonkin palvelun käyttämisestä, tulee esille se seikka, mil- lainen käyttökokemus palveluun jää. Muun muassa mielipiteet palvelun hyödyllisyydestä vai- kuttavat käyttäjän tuntemuksiin hänen käyttäessään palvelua. (Nuutila ym. 2009, 23.)

Asiakasorganisaation toimitusjohtaja (Robertsson, 2015) selventää antamassaan haastattelus- sa, että ihannetapauksessa palvelun kohderyhmä koostuu kaikenikäisistä musiikin harrastajis-

ta, muusikoista, soitinkauppiaista ja sellaisista henkilöistä, jotka hakevat henkilökontakteja musiikin ympärillä. Sovellusalustan kohderyhmä on käytännössä sama, kuin asiakkaan laajemman palvelun kohderyhmä. Yksittäinen käyttäjä voi valita hänen kiinnostuksiaan tai tarpeita vastaavan sovelluksen sovellusalustan vaihtoehtoista.

2.3 Hankkeen vaatimukset

Opinnäytetyön tavoitteena on asiakkaan jatkokehitystyötä helpottava dokumentointi ja mallinnus. Dokumentointi sekä mallinnus helpottavat sovellusalustan suunnittelua. Sovellusalusta on tässä tapauksessa verkkosivujen nippu, jossa toiminnalliset ominaisuudet näkyvät. Vaatimusmäärittelyissä, sekä asiakkaan tarjoamien materiaalien saatavuudessa oli työn aikana epämääräisyyttä. Lisäksi vaatimukset muuttuivat työn aikana. Tästä johtuen muutamia jatkokehityksen kannalta hyödyllisiä osa-alueita ei lopulliseen työhön sisällytetty.

Varsinaisiin vaatimuksiin, joiden perusteella työ toteutettiin, kuuluivat visuaalisen ilmeen eli rautalankamallin, prototyypin, pseudokoodin ja ohjeen laatiminen. Vaatimuksien saavuttaminen työssä oli sovittu opiskelijan ja hankeorganisaation välillä hankkeen aikana. Vaatimuksien toteutumista lopullisessa työssä käsitellään tämän raportin arviointiosuudessa. Yhteistyökumppaneilla tarkoitetaan tässä yhteydessä lähinnä toimijoita, jotka eivät ole kohdeorganisaation tai valmiin sovelluksen asiakasryhmän sisällä, mutta toimivat välillisinä sisällöntuottajina verkkosovellukselle. Tästä olisi esimerkkinä ulkopuolisen sovelluksen tarjoaja sovellusalustalle. Myöhemmän sisällön tuottamisesta vastaavat täten täysin hankeorganisaatio, sekä yhteistyökumppanit. Lisäksi visuaalisen ilmeen trendikkyyttä ei työssä huomioitu, sillä hankeorganisaation vaatimuksina oli käyttää verkkosivuston aiempaa visuaalista ilmettä.

Toiminnalliset sivut tulisivat olemaan valmiissa verkkopalvelussa sovelluksien valikkosivu, yksittäinen sovellussivu, sekä yksittäisen sovelluksen taustatietopalkki. Valikon avulla käyttäjä valitsee haluamansa sovelluksen. Taustatietolohko täsmentää käyttäjälle sovelluksen käyttöehdot sekä pohjatiedot. Taustatietolohko ei varsinaisesti siis ole erillinen sivu, vaan se tultaisiin toteuttamaan erillisellä ohjelmalla valikkosivun sisälle. Taustatietolohkon esiin tuleminen vaatii valikon käyttämistä. Sovellussivulla on mahdollisuus valita sovelluskohtaiset toiminnot. Tässä on hyvä huomioida, ettei sovelluksia voi välttämättä kytkeä sivustolle. Sovelluskohtaisia toimintoja on tällöin rajallinen määrä, tai ei ollenkaan.

2.4 Menetelmät

Jotta suunnitelmasta tulisi vaatimuksia vastaava, on projektin alkuvaiheessa ja toteutusvaiheessa valittava suunnittelu ja työstämismenetelmät. Lisäksi projektin edetessä laitteistovaatimukseen nykyisen verkkosivuston tietokannassa täytyy kiinnittää huomiota niiltä osin, kun on kyse jatkokehityksestä. Palvelun suunnittelussa viitekehyksinä toimii sosiaalisen median pal-

velun ohjelmoinnin, suunnittelun ja verkkotekniikoiden hyödyntäminen. Teoreettisen lähtökohdan kohdalla on myös selvitettävä käyttäjälle saavutettavat hyödyt. Menetelmät suunnitelman rakentamiselle olivat lähinnä asiakkaan yleisten vaatimusten varassa. Jatkokehityksen ottaminen huomioon on ensiarvoisen tärkeää, mikäli opinnäytetyön pohjalta syntyvää tuotosta voidaan hyödyntää.

Työkalut, jotka olivat työn aikana käytettävissä, valikoituivat saatavuuden ja helppokäyttöisyyden mukaan. Opiskelijan käyttämät työkalut koostuivat visuaalisessa suunnittelussa kuvankäsittelyohjelmasta ja rakenteiden suunnittelussa kirjoitusohjelmasta ja sivustotekniikoiden analysointi-ohjelmasta. Suunnittelutyön etenemisessä käytettiin apuna vertailuanalyysiä.

Suunnittelutyön pohjatyö on tehtävä siten että lopullinen palvelu toimisi hyvin ja olisi tuottoisa. Mikäli aiheesta on aikaisempi palvelu olemassa, voidaan apuna käyttää vertailukriteerejä. Vertailuanalyysillä (benchmarking) voidaan luoda neutraaleja tilanearvioita siitä, mihin suuntaan palvelunkehityksessä kannattaa edetä. Vertailuanalyysissä vertaillaan kahden samankaltaisen hyödykkeen ominaisuuksia, ja tehdään siitä kaupalliset johtopäätökset sekä kehitysprosessit. Käytännössä vertailuanalyysillä pyritään neutraalisti tarkastelemaan omaa hyödykettä. (Kaushik 2007, 262) Työssä vertailuanalyysiä tehtiin toisesta verkkosovelluksesta. Raportissa toisen osapuolen verkkosovellusta ei mainita nimeltä juridisista syistä.

3 Suunnittelutyön teoria

Teoreettisesti lähestyessä työn aihetta, tulee ottaa huomioon se, miten toteutussuunnitelman luominen tapahtuu. Suunnitelman ollessa hankeorganisaation niin sanottu mittatilaustyö, on hankkeessa oltu jatkuvassa vuorovaikutuksessa hankeorganisaation kanssa. Jatkuva vuorovaikutus on toteutettu, sillä jatkuva vuorovaikutus on oleellista ymmärryksessä siitä, mihin suuntaan ratkaisua ollaan viemässä ja miksi (Nuutila ym. 2009, 31).

”Ohjelmistosuunnittelussa kuvataan tulevan ohjelmiston rakenne, rakenneosien väliset rajapinnat ja tietosisältö, sekä joissain tapauksissa algoritmi.” (Sommerville 2011, 6) Ohjelmistosuunnittelu liittyy verkkopalvelujen suunnitteluun, mikäli palvelu on tarpeeksi laaja. Sosiaalisen median kehittämisessä voidaan puhua jo ohjelmistosuunnittelusta, sillä rajapintaa käyttöliittymän ja tietokannan välillä on suuresti. Linked By Music -verkkopalvelu on sisällöltään dynaaminen verkkosovellus, ja tarkemmin ottaen sosiaalinen media.

3.1 Keskeiset käsitteet

Tietojenkäsittelyn käsitteiden ympärillä on tärkeää luoda selkeä määritelmä käsitteille. Tämä johtuu siitä, että eri tietolähteissä on usein ristiriitoja ja päällekkäisyyksiä käsitteiden välillä. Esimerkiksi verkkosovellus ja verkkosivusto usein yhdistetään samaksi käsitteeksi.

Dynaaminen toiminto	Dynaamisella toiminnolla tarkoitetaan vuorovai- kutteista ja autonomista toimintoa. Dynaaminen toiminto verkkopalvelussa käytännössä tarkoittaa palvelimen ja käyttöliittymän välillä toimivaa toi- mintoa.
Käyttöliittymä	Käyttöliittymä on välialue käyttäjän ja toiminnon välissä. Käyttöliittymä on pakollinen jotta jotakin toimintoa voidaan käyttää.
Lohko	Lohkolla (englanniksi Block) tässä dokumentissa tarkoitetaan ohjelmoitua visuaalista ja toiminnal- lista ohjelmaa, joka tulee näkyviin käyttäjän toi- minnan perusteella.
Ohjelmointikieli	Ohjelmointikieli on tietojärjestelmälaitteen välilli- sesti ymmärtämää kieltä ja rajapinta käyttäjän ja toimintojen välillä. Ohjelmointikieli käännetään konekieleksi sillä hetkellä kun valmis ohjelma suo- ritetaan. Vasta käänнос välittää halutun tehtävän laitteelle.
Prototyyppi	Prototyypillä tarkoitetaan alustavaa mallia, jonka on tarkoitus selventää tulevaa kehitystyötä jolle- kin asialle. Vaihtoehtoisesti prototyyppi voi toimia valmiina pohjana, jota käytetään tulevissa asiois- sa.
Pseudokoodi	Pseudokoodilla tarkoitetaan ohjelmointikielen, se- kä tavallisen kielen välimuotoa, kuitenkin ole- matta mitään varsinaista koodikieltä. Pseudokoodi välittää tiedon siitä, miten tieto liikkuu ohjelman sisällä, sekä siitä, miten haluttu tieto näkyy käyt-

täjälle. Pseudokoodin tarkoitus on helpottaa ohjelmoinnin tuottamista.

Rautalankamalli	Rautalankamallilla tarkoitetaan alustavaa visuaalista verkkosivuston toiminta mallia. Rautalankamallin oleellisin osa on se että jokainen sivuston yksittäinen sivu otetaan huomioon.
Verkkopalvelu	Verkkopalvelu on itsenäisesti toimiva tai ylläpitoa vaativa verkon välityksellä toimiva palvelu. Verkkosivustot sekä verkkosovellukset ovat molemmat verkkopalveluita.
Verkkosovellus	Verkkosovellus on Internetissä toimiva toimintojen kokonaisuus. Verkkosovellukset sisältävät toimintoja, joilla käyttäjä voi käyttöliittymän avulla luoda ja käyttää sisältöjä. ”Verkkosovellukset ovat toimintalogiikaltaan monimutkaisempia ja vuorovaikutteisempia kuin verkkosivustot.” (Nuutila ym. 2009, 26)

3.2 Toteutussuunnitelman luominen

Mikäli halutaan tuottaa yleisesti ymmärrettävä ja toteuttamiskelpoinen suunnitelma, tulee osata yleiset käsitteet ja teoriat aiheesta. Koska suunnitelman luomisessa opiskelijalla oli vapaat kädet tuottaa toteutussuunnitelma, oli valittava helposti lähestyttävä malli. Lähtökohteisesti suunnitelman malli on oltava kaikkien ymmärrettävissä, mutta samalla hankkeen luonteen johdosta myös yksilöllinen hankeorganisaatiolle. Yleinen malli kuvastaa ratkaisuun liittyvien elementit, niiden keskinäiset suhteet sekä vastuut ja yhteistyön. Lisäksi laatutekijät, tavoitteet, sisällön suunnittelu ja graafisen suunnittelun osa-alueet ovat osana suunnitelman teoriaa. (Ohjelmistotekniikka, 2, 9)

Nuutilan ym. (2009, 41) mukaan verkkopalvelun suunnittelussa nykytilanteen, tavoitteiden ja käyttövaatimuksien määrittelyn jälkeen käynnistetään uuden palvelun suunnittelutyö, jossa palvelun sisältö, rakenne, toiminnot ja visuaalinen puoli suunnitellaan. Työn vaatimuksien vuoksi sisältöä ei tarvinnut tässä tapauksessa suunnitella.

3.3 Toimintojen mallintaminen

Suunnitelman mallinnus on oleellinen osa dynaamisen verkkosovelluksen luomista, sillä jokainen toiminnallisuus tulee mallintaa tarkasti. Käyttövaatimuksista tulee selville käyttäjäpersoonien laatu, sekä toimintatarinat. Käyttäjäpersoonat ovat erilaisten käyttäjien yksilöllisesti kuvattuja malleja. Toimintatarinoilla tarkoitetaan mallia, jossa käyttäjän tekemiä mahdollisia toimia kuvataan yksityiskohtaisesti. Toimintatarinat liitetään käyttäjäpersooniin ja liiketoimintatavoitteisiin, jolloin saadaan aikaiseksi kattava vaatimusten lista. Vaatimukset kartoitettua voidaan mallintaa helposti ne toiminnot, jotka ovat olennaisimpia palvelulle.

3.4 Liitosohjelmat

Mikäli sovelluksia liitetään verkkosivustolle, tulee käyttää liitosohjelmia. Liitosohjelmat upotetaan sivustolle erilaisilla verkkotekniikoilla toimimaan palvelimien välillä. Liitosohjelmat voidaan määritellä joko manuaalisesti tai automaattisesti. Manuaalisesti määriteltäessä ohjelmointikielellä lisätään liitosohjelman määrittelyt sivustolle. Automaattisesti liitosohjelma lisätään kun sivulla on jokin verkkotekniikka joka kommunikoi palvelimen kanssa, ja tietyt kriteerit saavuttaessaan liitosohjelma aloittaa toimintoja. Yleisesti ottaen liitosohjelma toimii yksittäisellä sivulla ja verkkoselain kääntää koodia suoraan käyttäjälle näkyviksi rakenteiksi. Rajapinta on sivuston sisäinen liitos, joka tekee useamman koodin yhteen liittämistä. Liitosohjelmien alustava läpikäyminen kuului osana hankkeeseen, sillä vaatimuksissa oli selvitettävä, mikä saavutettavissa oleva rajapinta sopisi parhaiten käyttötarkoitukseen. Asiakkaan sivustolla työn aikana saavutettavat rajapinnat olivat React!, JavaScript sekä HTML5.

3.4.1 HTML5

HTML on standardisoitu verkkosivustojen ohjelmointikieli, joka toimii jokaisella perinteisellä verkkosivulla. HTML5 on kehityksessä oleva HTML laajennus, jonka ensimmäinen julkaisukelpoinen versio julkaistiin 2014. HTML5 sisältää merkittäviä uudistuksia muun muassa sovellusvälimuistirakenteen, sekä selaimen muistin joiden avulla voidaan luoda dynaamisia toimintoja sivustolle ilman muita ohjelmointiratkaisuja. (Korpela 2011, 13.)

3.4.2 Javascript

Javascript on autonominen käyttäjärajapinta ja ohjelmointikieli verkkoselaimen ja palvelimen välillä. Javascriptillä mahdollistetaan muun muassa vuorovaikutusta palvelimen ja käyttäjän dynaamisissa toiminnoissa. (What is JavaScript, 2015)

3.4.3 React!

Hankeorganisaation verkkosovellus oli aiemmin toteutettu suurimmaksi osaksi käyttäen Facebook-yhtiön luomaa react! -liitosohjelmaa. React! perustuu javascript:iin ja on valmiita malleja (templates) hyväksikäyttävä käyttäjärajapinta. (Why we did build React?, 2013)

React!:in käyttö dynaamisen verkkosovelluksen luomisessa on käytännöllistä, sillä sen tuomat edut palvelimen ja vaihtuvan sisällön ohjelmoimisessa on helppoa.

3.5 Verkkopalvelun visuaalisuus

Yritystoiminnan peruskivinä ovat tuottaa hyödykkeitä, jotka ovat tarpeellisia, trendikkäitä ja samalla taidokkaasti tehty. Verkkopalveluiden tärkeimpiä ominaisuuksia kaupalliselta kannalta ovat trendikkyys ja käytettävyys. Verkkopalveluiden tuottavuus voidaan mitata sillä, että kuinka paljon loppukäyttäjää saadaan vierailemaan sivulla useammin kuin kerran. Mikäli palvelu on miellyttävä ja toimiva, käyttäjä luultavimmin tulee sivulle toisenkin kerran. Trendikkyyteen panostaminen tarkoittaa käytännössä sellaisen visuaalisen ilmeen luomista, joka erottuu kauemman aikaa sitten luoduista palveluista positiivisesti. (Stolley 2011, 70.)

Visuaalisuutta työn kannalta otettiin huomioon alkuperäisten vaatimuksien pohjalta. Vaatimuksien muututtua hankkeen puolivälissä lähemmäksi hankeorganisaation aiempaa ilmettä, ei ollut enää tarvetta luoda tutkimusta uudesta visuaalisesta ilmeestä. Visuaalisuuden tärkeyttä käytettävyydessä hyödynnettiin rautalankamallien suunnittelutyössä vertailuanalyysin pohjalta.

4 Suunnittelutyö

Tässä kappaleessa käydään läpi ne kronologiset vaiheet, jotka kuuluvat opinnäytetyön toteutusosaan. Suunnittelussa käytetyt menetelmät luovat jatkokehityskohteelle selkeyttävän vaikutuksen. Työn lähtökohdissa annetut vaatimukset toimia sovellusalustana asiakasorganisaation yhteistyökumppaneille loivat tarpeen selvitykselle siitä, miten sovellusalusta käytännössä toimii. Sovellusalusta tässä tilanteessa on sivuston toiminnallisuus, johon voidaan lisätä palvelinrajapinnan avulla toiminnallisia sovelluksia.

Suunnittelussa käytettiin hyväksi jatkuvaa vuorovaikutusta hankeorganisaation kanssa ja suunnitteluprosessi, sekä ohje yksilöllistettiin asiakasorganisaatiolle. Lisäksi käyttäen vertailuanalyysiä ulkopuolisen kehittäjän sovellusalustasta voidaan luoda pohja visuaaliselle suunnitelmalle. Ulkopuolisen yrityksen nimeä ei hankeorganisaatio halunnut paljastettavan.

Suunnittelussa eri vaiheille oli käytössä muutamia ohjelmistoja. Rautalankamallit sekä pseudokoodin ilme toteutettiin Gimp-2 - kuvankäsittelyohjelmalla. Gimp-2 valiutui rautalankamallien tuottamiseen, sillä opiskelijalla oli valmiina asiantuntemusta kyseisestä sovelluksesta, sekä sen toiminnallisuuksien vuoksi. Rautalankamalleista tuotettiin myös alustavat paperiveriot, sillä käsin piirtämisessä tulee esille suurempi joustavuus ja nopeus.

Verkkosivujen prototyypit toteutettiin Notepad -kirjoitustyökalulla, johon taustatutkimukseen tarvittu tieto kerättiin Mozilla Firefox-verkkoselaimen Firebug lisäosalla. Lisäosa toiminnot keskittyvät HTML ja tyylikoodien lukemiseen. Nämä koodit ovat perustana verkkosovelluksien rakenteelle. (Firebug, 2015) Lisäksi muutamia prototyyppien visuaalisia testauksia toteutettiin W3schools:in Html Try It Yourself - työkalulla. Kyseisellä työkalulla voidaan luoda dynaamisesti uusia rakenteita. Try It Yourself sisältää useita uusimpia verkkosovelluksien rakenteiden luomisessa käytettäviä rajapintoja. (HTML(5) Tutorial, 2015) Jokainen työhön käytetty ohjelma oli opiskelijalle ilmeikäkäytössä työn aikana.

Kun puhutaan suunnitelmaan käytetyistä työvälineistä, voidaan puhua oikeastaan ohjelmista, joilla suunnitelmaa on luotu, ja ohjelmointikielistä joilla tuleva palvelu oletettavasti rakennetaan. Ohjelmointikielten läpikäyminen avustaa toteuttamiskelpoisen suunnitelman luomista, ja kielen tuntemuksen pohjalta on käytännöllistä luoda jokin alustava malli. Ohjelmistojen alustavassa mallintamisessa käytetään yleensä pseudokoodia, jonka avulla jatkokehitys ohjelmiston ohjelmointiin tehdään helpommaksi. Pseudokoodin luominen tässä hankkeessa tapahtui välillisesti yhdessä aiempaan palveluun käytetyiden kielten kanssa. Rajoitteina on kuitenkin luoda vain viite, jolloin vain perusteet varsinaisten kielten tuntemiselle riittää.

4.1 Toiminnallisuuden käyttötarina

Vaatimusmäärittelyn pohjalta on helpointa luoda niin sanottu storyboard-, eli käyttäjän käyttötarinamalli. Käyttötarinamalli kuvastaa mahdollista käyttäjän toimintaa palvelussa, ja mallit suunnitellaan vastaamaan eri käyttäjien käyttötarkoituksia palvelussa (Nuutila ym. 2009, 171). Tulevien toimintojen määrä hankkeessa rajoitti kuitenkin käyttötarinoiden laajuutta. Tämän vuoksi käyttötarinamalliin oleellisten yksilöllistettyjen käyttäjäprofiilien mallintamista ei ole huomioitu. Yhteistyökumppaneiden tarjoamat sovellukset tulevat olemaan myös vaihtelevia, jolloin sovellusalustan ja verkkosivuston kytkös itse sovelluksiin vaihtelee.

Työhön suunniteltavien sivujen pääsääntöiseen vaatimukseen kuului tuoda esille lista sivustolle yhteen liitetyistä sovelluksista. Käyttäjä on alkutilanteessa siirtynyt sovellusalustalle klikkaamalla sovellusvalikkoa sivuston navigointiominaisuudesta. Listan avulla käyttäjä voi selata liitetyistä sovelluksista häntä kiinnostavan sovelluksen. Valittuaan sovelluksen, aukeaa hänelle sivu, jossa sovellus avautuu käyttöä varten. Sovellussivulla ensisijainen toiminto on käyttää

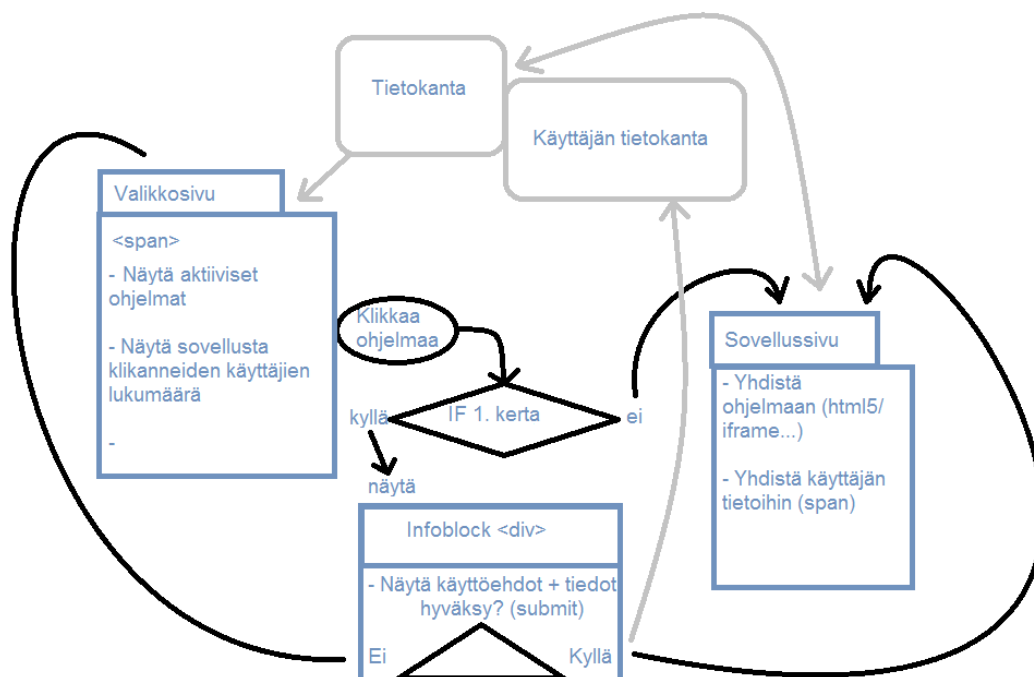
sovellusta. Kuitenkin ensimmäisellä kerralla sovellusta klikatessa valikosta, tulee sovelluksen sivulla esille selvästi erottuva ”tietolaatikko”, joka kertoo valitun sovelluksen ominaisuuksista lyhyesti. Tietolaatikon tehtävänä on myös kertoa tietoa käyttäjälle siitä, mitä vaatimuksia ja ehtoja sovelluksella on. Nämä ehdot voidaan joko hyväksyä tai hylätä käyttäjän valinnan mukaan. Hyväksyttäessä käyttäjä antaa oikeuden palvelun tehdä sovelluskohtaisia asioita hänen omalle sivustosisällölleen. Hylättäessä palaa hän takaisin sovellusvalikon sivulle.

Toinen käyttötarinamalli kertoo lisäominaisuuksista, jotka tulevat sovelluksen ohessa. Käyttäjä voi valita sovellussivulla haluaako hän jakaa sovellustietojaan hänen sivustosisällölleen. Valitessaan jakotoiminnon, hänen sovellustietonsa välittyvät hänen omalle sisältösivulleen. Sivuston muut käyttäjät voivat nähdä tämän jälkeen kuvauksen siitä, mitä sovelluksella on saavutettu. Esimerkkinä tästä voisi olla sovellusalustaan upotettu musiikinteko-ohjelma. Musiikinteko-ohjelman sovellustiedot jakautuisivat käyttäjän sisältösivulle, eli tässä tapauksessa hänen profiilisivulleen. Ohjelma voisi kertoa tässä tilanteessa esimerkiksi sen, että käyttäjä on luonut uuden musiikkiraidan ja haluaa jakaa tämän muiden sivuston käyttäjien kanssa.

4.2 Pseudokoodi

Pseudokoodi on suunniteltu verkkopalvelun rakentamisen kannalta kevyimpään muotoon. Pseudokoodin luominen avustaa lopullista käyttäjärajapinnan ja palvelimen vuorovaikutuksen toteuttamista. Vaikka jatkokehitystä varten olevat kielet olisivat selvillä, on silti helpompaa ja nopeampaa suunnitella ohjelma pseudokoodiin riippumattomalla kielellä. (Pseudocode: An Introduction, 2015)

Pseudokoodin toteuttaminen lähtee liikkeelle yleensä selvittämällä perustat ohjelmoimiselle. Pseudokoodin ohjelmointi perustuu loogisiin operaattoreihin, muuttujiin, toistorakenteisiin ja symboleihin. Näillä voidaan täsmentää haluttujen ominaisuuksien toteutumista. Pseudokoodi toteutettiin luomalla vaadittu lopputulos, jonka tehtävä oli näyttää sovellus omalla sivullaan, ja esittää kaikki välitoiminnot sovellusvalikon ja profiilin välillä. Koodin väliin luotiin kautta-viivoilla (//) kommentit, jotka toimivat selvennyksinä ohjelmiston toiminnasta. Koodi toteutettiin hyvin alkeellisesti sen vuoksi, että sen päälle voidaan tilannearvion mukaan valita haluttu ohjelmointikieli tai rajapinta. Koodin pohjalta luotiin helpommin ymmärrettävä kaavio-kuva. Kuvasta selviää miten koodille oleelliset toiminnot liittyvät tietokantoihin ja sivuihin.



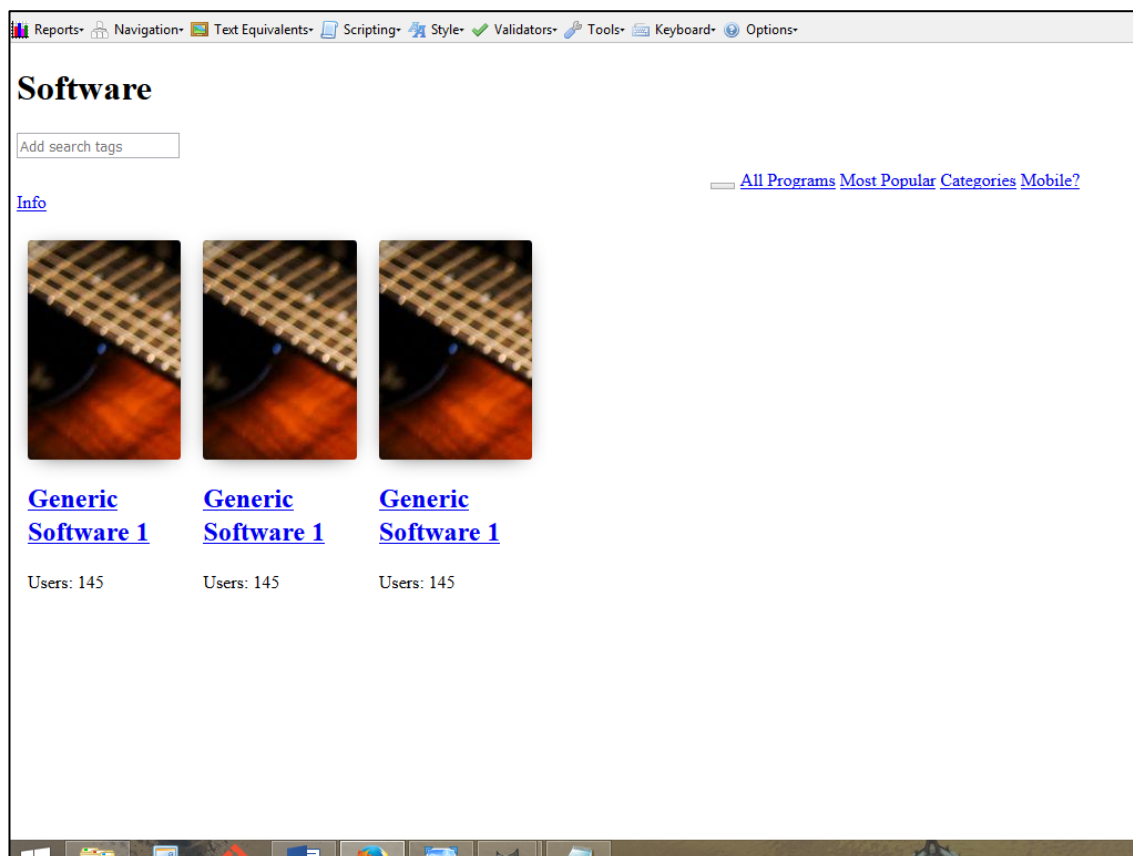
Kuva 1: Pseudokoodin pohjalta luotu kaavio

4.2.1 Prototyyppi

Prototyyppi luotiin vastaamaan verkkosivuston ilmettä. Tässä työssä sivuston prototyyppi kuvastaa HTML -ratkaisua sovellusvalikossa sekä yksittäisellä sovellussivulla ilman tyylimääritteitä. Tyylimääritteet voidaan toteuttaa välittömästi jatkokehityksen aikana, mikäli HTML -ratkaisu on luotu vanhan pohjan mukaiseksi tarpeeksi yksinkertaisena. Prototyypin toteutus oli yksinkertainen. Prototyyppi sisälsi elementtiluokkia, linkkejä, tekstiä ja dynaamisen ominaisuuden viittauksia. Viittaukset ovat viitteitä pseudokoodin pohjalta syntyviin dynaamisiin ratkaisuihin sivun sisällä. Prototyypin hyödyllisyys perustuu pseudokoodinratkaisupaikoille, sillä html-koodin luomisratkaisut ovat täysin jatkokehityksen varassa.

Valikkosivun prototyyppi luotiin käyttämällä Firefox-selaimen Firebug-ohjelmaa, jolla asiakkaan verkkopalvelua tutkittiin rakenteellisesti. DIV-rakenteilla, eli tyylimääritteiden jakamista helpottavilla rakenteilla toteutettu sivu tuli järjestellä uudelleen, jotta vaaditut lista-elementit näkyisivät oikein. DIV-rakenteet järjesteltiin prototyyppiin vastaamaan listaa kuvan (kuva 4) mukaisesti. DIV-rakenteiden sisällä oli pseudokoodin mukaiset viittaukset tietokantaan. Viittaukset eivät toimi prototyypissä sellaisinaan, vaan ne tulee ohjelmoida palvelun jatkokehityksen aikana. Yksittäisen sovelluksen prototyyppi toteutettiin samalla tavalla kuin

valikkosivu, sillä poikkeuksella että listan sijaan näkyy iframe-viittaus ulkopuolisen sovelluksen upotukselle. Sovelluksen upotus toimii valmiissa verkkosovelluksessa sivun sisällä. Upotuksen toteutus on raportissa esiintyvien ehdotuksien mukaan toteutettavissa jatkokehityksen aikana.



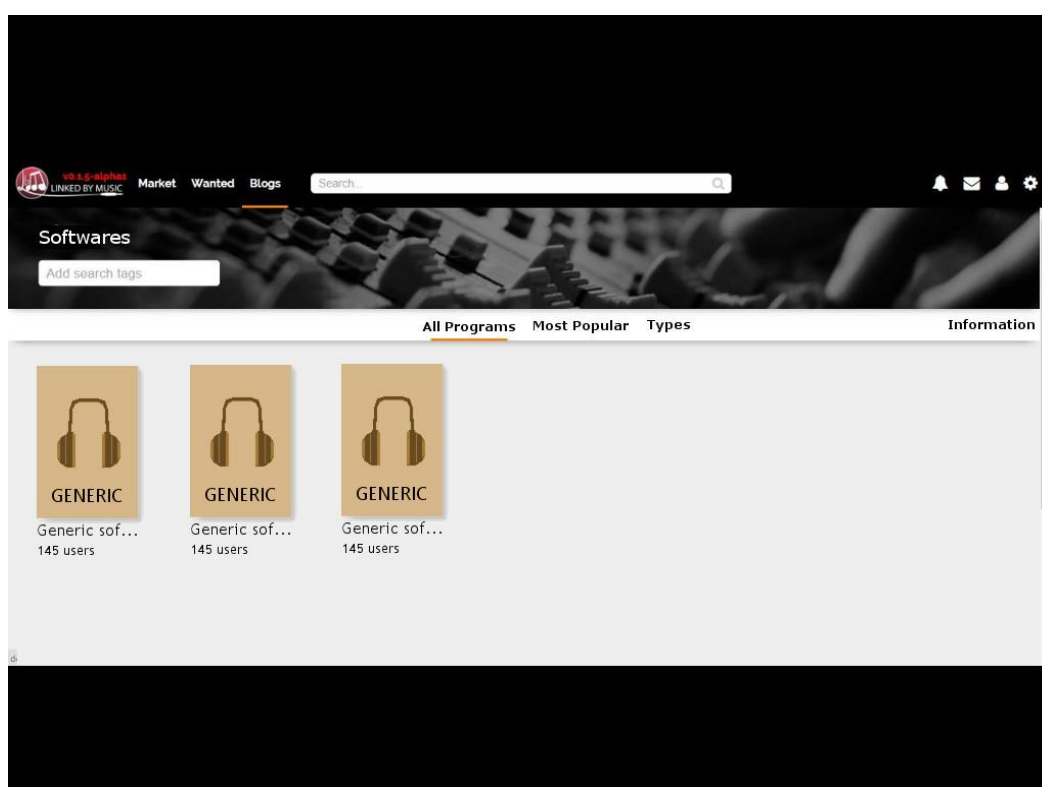
Kuva 2: Pelkistetty HTML Prototyyppi valikkosivusta

4.3 Visuaalinen suunnitelma

Visuaalisen ilmeen vaihtoehdot ovat helpoimmin vaihdettavissa, mutta vaikuttavat suuresti myöhemmissä kohdissa tuleviin seikkoihin. Trendikkyden ja käytettävyyden suunnittelu on tässä vaiheessa otettu laajalti huomioon. Asiakkaan sivuston perusteella luotu visuaalinen ilme on suunniteltu vastaamaan tavoitteita. Lähtökohtaisesti tavoitteisiin kuuluvat kolmen sivun suunnitelmat ovat asiakkaan vaatimukseen pohjautuvia. Lisäksi opiskelijan tuomat lisäykset (mm. sivupalkit toiminnallisuuksineen) ovat lisätty asiakkaan mallin pohjalta.

4.3.1 Rautalankamallit

Rautalankamallien suunnittelutyö alkoi suunnittelemalla paperille käsin piirretyillä malleilla. Paperisen mallin pohjalta syntyneen ensimmäisen version pohjalta asiakas sai arvioida ovatko vaaditut asiat tulleet esille. Asiakkaan arvion jälkeen päädyttiin suunnittelemaan rautalankamallit olemassa olleen verkkosivuston kehitysmallin mukaiseen visuaaliseen ilmeeseen. Kuvankäsittelyohjelmalla otettiin kuvankaappaus eräästä valmiista sivusta, joka muokattiin vastaamaan uutta sivua. Täten pystyttiin luomaan alustava malli sivuston yksittäisestä sivusta.



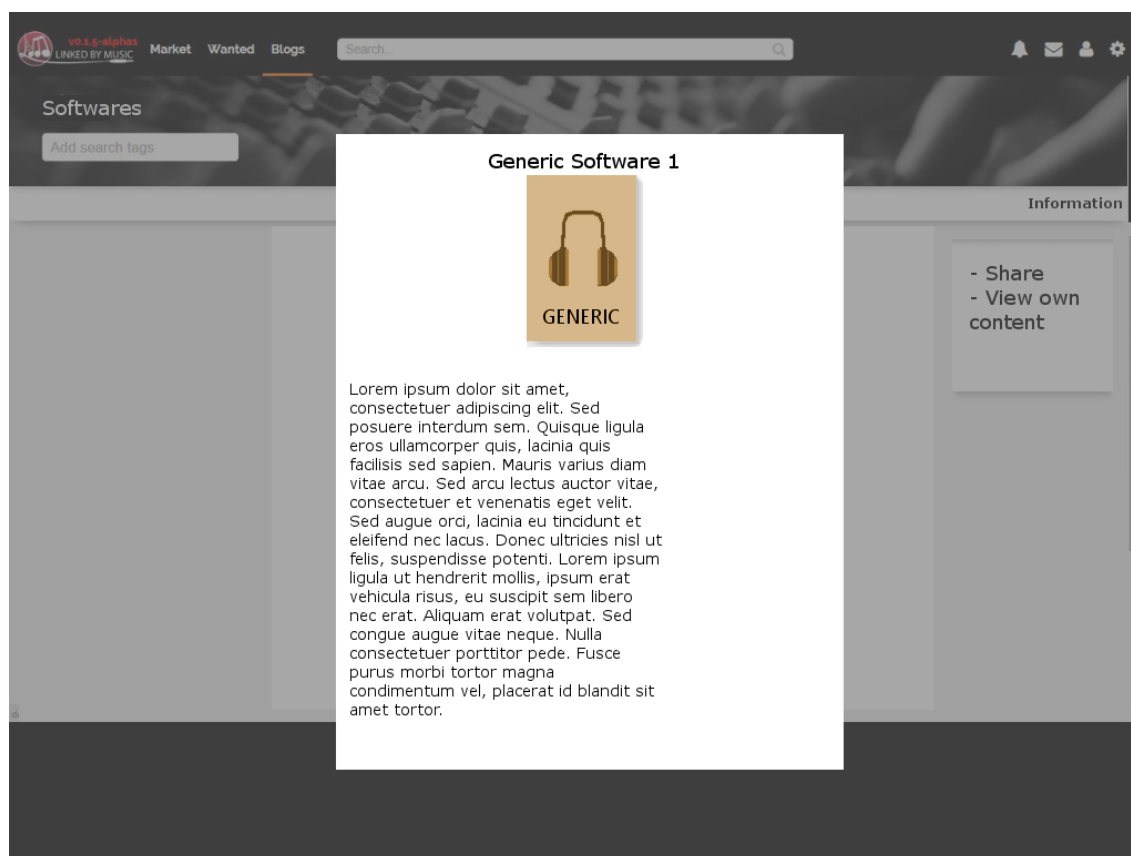
Kuva 3: Valikkosivu

Sovellusvalikon valikkosivulle oleellisia osia olivat: lista sovelluksista, navigointipalkki sekä hakukenttä. Navigointipalkki sekä hakukenttä olivat valmiina verkkosovelluksen aiemmissa sivuissa. Uutena elementtinä tulisi olemaan lista käytettävistä sovelluksista. Sovellukset näkyvät listalla kuvina, joiden alapuolella on sovelluksen nimi (Generic Sof...) sekä sitä käyttäneiden rekisteröityneiden käyttäjien lukumäärä (145 users). Sovelluksen kuva, tiedot sekä käyttäjien lukumäärä muuttuvat dynaamisesti pseudokoodista esille tulleiden asioiden mukaisesti.

Tyylillisesti mallit seurailevat verkkosovelluksen tyyliä. Tyylimäärittäisiin voidaan asettaa sivuston aiemman tyylitiedoston määrittäet, kun HTML-prototyypin pohjalta on tehty määrittelmät uudelle sivulle, esimerkiksi div-määrittäisiin. Vaihtoehtoisesti tyylin voi muokata opiskelijan ehdotusten mukaisesti (kaavio X).

	backgroundcolor	Padding	margin	font size
Kuva/tausta	#d6b789	15px	na	na
Name/sovellus	na	1px	1 em 0;	15p
Users	na	inherit	inherit	11pt
background	#a0a0a0	na	na	na

Taulukko 1: Tyylimäärittelyt



Kuva 4: Sovelluksen info-blockki

5 Arviointi ja kehitysehdotukset

Suunnitelmaan tuodut kehitysehdotukset pohjautuvat kirjoittajan aikaisempiin kokemuksiin verkkosovelluksien luomisesta, sekä aiheen pohjalta tehtyyn tutkimustyöhön. Alkuperäisen tutkimustyön vaatimuksina oli luoda kokonaisuudessaan uusi sovellusalustasuunnitelma ja käyttää viitekehyksiä helppokäyttöisten verkkosovellusten luomisessa. Hankkeen puolivälissä tulleiden uusien vaatimuksien vuoksi ei täysin uutta suunnitelmaratkaisua tarvinnut luoda. Hankeorganisaation verkkosovelluksen visuaalinen ilme, sekä käytetyt ohjelmointiratkaisut otettiin käyttöön työn valmistelemiseksi. Muuntuneita vaatimuksia korvattiin kehitysehdotuksilla ja tarkemmalla suunnittelulla tietyissä osuuksissa, esimerkiksi HTML-prototyypeillä.

Mikäli opinnäytetyössä luotua suunnitelmaa hyödynnetään verkkosovelluksen jatkokehityksessä, on yleisesti osattava tulkita opinnäytetyössä syntynyttä raporttia. Asiakkaan palvelun tekniikoita ei hyödynnetty projektin aikana. Tekniikoiden hyödyntämisessä olisi ollut suuri merkitys työn jatkokehitysarvolle. Kuitenkin opiskelijan aiemmat kokemukset, sekä aikaisempi yhteistyö hankeorganisaation kanssa toivat mahdollisuuksia työn hyödyntämiselle. Suunnitelman ymmärtämisessä ovat suurimmat kriteerit hankkeen arvioinnissa ja uskottavuudessa.

Asiakkaan mielestä opinnäytetyössä syntynyt suunnitelma oli erittäin hyvä pohja jatkokehityksen kannalta. Asiakkaan lopulliset tavoitteet olivat luoda tutkimus tavoista, joilla verkkotekniikat tulisivat toimimaan asiakkaan palvelinympäristössä. Lisäksi palvelimen toiminnan tekniikoita ei työssä tutkittu. Asiakkaan mielestä visuaalisen puolen suunnitelman toteutus oli onnistunut.

Kehitysehdotukset asiakkaalle pohjautuvat lähinnä tutkimuksien ja laajamittaisen tuotekehityksen ympärille. Mikäli jatkokehityskohteissa on selkeä tieto kaupallisten palveluiden kannattavuudessa, on oleellista asettaa heti aluksi tarkat vaatimukset palveluiden toiminnallisuuksiin. Toiminnallisuudet voidaan luoda täten ketterämmin ja arvailuille jätetään mahdollisimman vähän tilaa. Jos kannattavuustutkimusta palvelusta ei ole tehty, palveluiden jatkokehittämisellä voi olla tappiollinen vaikutus rahallisesti. Tämän työn kannalta kannattavuustutkimuksen tekeminen on suhteellista, sillä työlle ei määritelty hintaa.

6 Johtopäätökset ja yhteenveto

Tässä työssä tekniikan valinnat keskittyivät aluksi laajamittaisten tutkimuksien ympärille, joita olivat mm. tutkimukset käytännön hyödyistä ja tarpeellisuudesta, sekä näiden pohjalta tehdyistä toiminnallisuuksista. Asiakkaan vaatimusten puitteissa ei kuitenkaan kyseistä tutkimusta tarvittu, vaan vaatimukset keskittyivät työn edetessä sovellusalustan suunnittelutyölle. Sovelluksen vaadittujen ominaisuuksien lisäksi suunnitelmassa pitää keskittyä siihen, miten viitekehykset muodostavat halutut ominaisuudet teknisesti. Sovellusalustan ominaisuudet

keskittyivät sovelluksien yhdistämiseen. Viitekehysten tekniikat luovat teknisen toteutuksen. Käytännössä tämä johtaa siihen, että hyvä suunnitelma on mahdollisimman toteutuskelpoinen ja noudattaa selkeää linjaa.

Jatkokehityskohteet keskittyvät HTML-ratkaisujen ja visuaalisen ilmeen ympärille. Valmis HTML-koodi antaa asiakkaalle valmiin pohjan luoda sovellusalustan sivut omien tarkoitusperiensä mukaisesti. Pseudokoodin rooli on helpottaa hahmottamista palvelinympäristön liittämässä sovellusalustan sivuille. Visuaalisen ilmeen mallintaminen yhdessä valmiiden tyyliä-ritteiden kanssa antaa lopullisen ilmeen sivustolle, jolloin asiakkaan ei tarvitse käyttää aikaa visuaalisuuden viimeistelyyn. Lisäkehityskohteina sivujen toiminnalle tehtiin taustatietolaa-tikko, kytkökset käyttäjätunnusten tietokantaan, sekä sivujen yleinen rakenne toimia listana valittavissa olevista sovelluksista.

Projekti eteni suurelta osin työn alkuvaiheessa tapahtuneen suunnitelman mukaisesti ja työn lopputuloksissa näkyvät alustavat menetelmät verkkosovelluksen suunnitteluun. Nykyaikana tekniikoita verkkosovellusten luomisessa on lukuisia. Riippuen käyttötarkoituksista voidaan valita käytännöllisin tekniikka halutun toiminnallisuuden toteuttamiselle. Tekniikka voi koostua esimerkiksi viitekehyksistä, tai laajamittaisemmista kehitysalustoista.

Lähteet

Kirjat

Kaushik A. 2007. Web Analytics: An Hour a Day. : Sybex

Korpela J. 2011. HTML5. Jyväskylä: Docendo

Nuutila E., Sinkkonen I. & Törmä S. 2009. Helppokäyttöisen verkkopalvelun suunnittelu, 2. painos. Helsinki: Tietosanoma

Sommerville, I. 2011. Software engineering, Ninth Edition. Boston: Addison-Wesley

Sähköiset

Tuntematon julkaisija. 2015. Pseudocode: An Introduction. Viitattu 23.9.2015.
(<http://faculty.ccri.edu/mkelly/COMI1150/PseudocodeBasics.pdf>)

Hewitt J. 2015. Firebug, Viitattu 18.11.2015.
(<https://addons.mozilla.org/fi/firefox/addon/firebug/>)

Tuntematon julkaisija. 2015. Viitattu 18.11.2015.
(<http://www.w3schools.com/html/>)

Lappalainen J. 2015. Ohjelmistotekniikka, luento 7. Viitattu 20.11.2015.
(https://noppa oulu.fi/noppa/kurssi/811335a/luennot/811335A_luento_7.pdf)

Neter P. 2015. What is JavaScript?. Viitattu 18.11.2015.
(https://developer.mozilla.org/en-US/docs/Web/JavaScript/About_JavaScript)

Hunt P. 2013. Why we did build React?. Viitattu 19.11.2015.
(<https://facebook.github.io/react/blog/2013/06/05/why-react.html>, viitattu 18.11.15)

Julkaisemattomat

Robertsson V. Linked By Music, toimitusjohtajan Vesa Robertsson haastattelu, Espoo 2015

Kuviot

Kuva 1: Pseudokoodin pohjalta luotu kaavio.....	16
Kuva 2: Pelkistetty HTML Prototyyppi valikkosivusta.....	17
Kuva 3: Valikkosivu	18
Kuva 4: Sovelluksen info-blockki	19

Taulukot

Taulukko 1: Tyylimääritys.....	19
--------------------------------	----

Liitteet

Liite 1 Linked By Music, toimitusjohtajan Vesa Robertsson haastattelu	26
Liite 2 Prototyypikoodi sovellusalustan valikkosivusta	27
Liite 3 Pseudokoodit.....	29

Liite 1 Linked By Music, toimitusjohtajan Vesa Robertsson haastattelu

Linked By Music on tällä hetkellä kehitysvaiheessa oleva sosiaalinen media. Sivuston tarkoituksena on toimia “muusikoiden sosiaalisena mediana”, joka tarjoaa LinkedIn tyyppistä ratkaisua musiikista kiinnostuneille käyttäjille. Ensimmäisessä julkaisuvaiheessa palvelu vastaa musiikin soittamista harrastavien yksityishenkilöiden tarpeisiin. Tällaisia tarpeita ovat esimerkiksi soittajien ainainen dilemma, eli soittimia on joko liikaa, liian vähän tai vääränlaisia sekä mikäli soittajalta puuttuu esimerkiksi yhtye tai yhtyeestä jäseniä. Käyttäjän profiili toimii hänen musikaalisena CV:nään, se kertoo, millainen käyttäjä on musiikkialan ihmisenä. Sivustolla on kauppapaikka missä käyttäjät voivat ostaa, myydä ja vuokrata kaikenlaisia musiikkiin edes jollain tasolla liittyviä tavaroita. Linked By Music ei itse myy kauppapaikalla tavaroita, joten omaa verkkomaksupalvelua ei tarvitse toteuttaa. Verkkosivusto toimii vain “välikätenä” käyttäjien välisessä kaupankäynnissä, tarjoten käyttäjille mahdollisuuden helposti löytää ja myydä tarvitsemiaan tavaroita.

Linked By Music tarjoaa lisäksi ns. Wanted-ympäristön, missä käyttäjät voivat etsiä yhtyeitä, yhtyeisiinsä jäseniä, musiikinopettajia, tuuraajia, dj:tä, miksaajia, valomiehiä ja “keikkapaikkoja”. Tarkoituksena on tarjota yksinkertainen tapa yhdistää musiikin harrastajat toisiinsa, ja täten tehdä esimerkiksi yhtyettä vailla olevan muusikon elämä hyvin paljon helpommaksi.

Muita Linked By Musicin tarjoamia toimintoja ovat mm. helppokäyttöinen blogityökalu, jossa käyttäjät voivat pitää omia blogejaan ja seurata muiden blogeja, sekä sovelluspohja, jossa käyttäjät voivat hieman Facebookin tapaan käyttää erilaisia tarjolla olevia sovelluksia. Nämä sovellukset eivät kuitenkaan ole Linked By Musicin kehittämiä, vaan ne ovat eri yhteistyökumppaneiden toteuttamia.

Liite 2 Prototyypikoodi sovellusalustan valikkosivusta

```
<span data-reactid=""></span>
<span data-reactid=""></span>
<div class="soft-ter" data-reactid="" >
<a class="" data-reactid="" href="/software">
<h2 data-reactid="">Generic Software 1</h2>
</a>
<div data-reactid="">
<div class="ui-user-count" data-reactid="">
<span data-reactid="">Users:</span>
<span data-reactid=""> </span>
<span data-reactid="">145</span>
</div>
</div>
</div>
</div>
</div>
<div class="ui-card" data-reactid="" style="padding:5px 0;margin:0;position:relative;">
</div>
</div>
<div data-reactid="" style="width:33.33333333333336%;float:left;">
<div data-reactid="" style="width:33.33333333333336%;float:left;">
<div data-reactid="" style="clear:both;">
</div>
</div>
</div>
</div>
<div data-reactid="" style="padding:0px;">
<div data-reactid="" style="width:10%; float:left;">
<div data-reactid="" style="padding:10px;" style="background-
image:url(http://img.linkedbymusic.com/587bc46d437e29afeda0bd31693b1bb7.jpg);">
<div class="" data-reactid="" style="padding:1px 0;margin:0;position:relative;">
<div class="soft-header" style="background:#fff;box-shadow:0px 3px 20px -10px #000;border-
radius:3px;background-
image:url(http://img.linkedbymusic.com/587bc46d437e29afeda0bd31693b1bb7.jpg);">
<div class="blog-post-body" data-reactid="" style="font-size:0.8em;visibility:hidden;">
<p></p>
<p> </p>
</div>
```

```
</div>
<span data-reactid=""></span>
<span data-reactid=""></span>
<div class="soft-ter" data-reactid="" >
<a class="" data-reactid="" href="/software">
<h2 data-reactid="">Generic Software 1</h2>
</a>
<div data-reactid="">
<div class="ui-user-count" data-reactid="">
<span data-reactid="">Users:</span>
<span data-reactid=""> </span>
<span data-reactid="">145</span>
</div>
</div>
</div>
</div>
</div>
<div class="ui-card" data-reactid="" style="padding:5px 0;margin:0;position:relative;">
</div>
</div>
<div data-reactid="" style="width:33.333333333333336%;float:left;">
<div data-reactid="" style="width:33.333333333333336%;float:left;">
<div data-reactid="" style="clear:both;">
</div>
</div>
</div>
</div>
</div>
</body>
</html>
```

Liite 3 Pseudokoodit

Valikkosivu

```
// käyttäjät jotka ovat joskus klikanneet listalla olevaa sovellusta
```

```
get list_active_software {
```

```
    //laskee sovellukselle lukumäärän käyttäjiä esim. (span) -tyylisesti.
```

```
    count_users from URL1 and URL2/database
```

```
}
```

```
clicked software_list_link {
```

```
    // mikäli käyttäjä ei ole aiemmin käyttänyt valitsemaansa ohjelmaa
```

```
    if user_first_time_using_software {
```

```
        // näytä ensikerran infoblockki, tummenna taustaa
```

```
        show info_block
```

```
        background set brightness -50% and contrast -50%
```

```
        //valinta haluaako käyttäjä jatkaa luettuaan käyt-
```

```
töehdot
```

```
        if clicked yes {
```

```
            proceed to software_page
```

```
            reset brightness -50% and contrast -50%
```

```
            add current_user's_click to URL2/database
```

```
    }  
  
    else {  
  
        reset brightness -50% and contrast -50%  
  
    }  
  
    else {  
  
        proceed to software_page  
  
    }  
  
}
```

Sovellussivu:

```
//käyttäjän klikatessaan jako-toimintoa
```

```
clicked share {
```

```
    //ota sovellustiedot sovelluksesta X ja yhdistä nykyisen käyttäjän profiiliin
```

```
    get software_information_X to current_user_data
```

```
}
```

```
//käyttäjän klikatessa näytä oma sisältö liittyen sovellukseen
```

```
clicked show_own_content {
```

```
    goto current_user_data (URL...)
```

```
}
```