

Eeli Kivikaarre

**TIETOVISA-VERKKOSIVUSTON TOTEUTTAMINEN BIOANALYTIIKAN
OPPIMATERIAALIA VARTEN**

**TIETOVISA-VERKKOSIVUSTON TOTEUTTAMINEN BIOANALYTIIKAN
OPPIMATERIAALIA VARTEN**

Eeli Kivikaarre
Opinnäytetyö
Kevät 2019
Tietojenkäsittely
Oulun ammattikorkeakoulu

TIIVISTELMÄ

Oulun ammattikorkeakoulu
Tietojenkäsittely, Web-sovelluskehitys

Tekijä: Eeli Kivikaarre

Opinnäytetyön nimi: Tietovisa-verkkosivuston toteuttaminen bioanalytiikan oppimateriaalia varten

Työn ohjaaja: Pekka Ojala

Työn valmistumislukukausi ja -vuosi: Kevät 2019

Sivumäärä: 27 + 3

Tässä opinnäytetyössä tarkoituksena oli kehittää verkkosivusto ja siihen liittyvä tietovisapeli bioanalytiikan oppimateriaalia varten. Työn tavoitteena oli oppia lisää web-sovelluksen kokonaisvaltaisesta kehittämisestä ja siihen liittyvistä eri osa-alueista, sekä tuottaa oppimateriaali, jota bioanalytiikan opiskelijat sekä myös työelämässä olevat bioanalytikot voivat jatkossa hyödyntää. Aiheen valitsemiseen vaikutti suomenkielisen oppimateriaalin puute verisolujen tunnistamiseen liittyen bioanalytiikan alalla. Työllä ei ollut ulkoista toimeksiantajaa, mutta siinä haluttiin alusta asti saada aikaan jotain, mistä voisi olla hyötyä ihmisille tulevaisuudessa.

Opinnäytetyön tietoperustassa on tutkittu web-sovelluksen kehittämisen eri osa-alueita. Tietolähteenä on käytetty aiheeseen liittyvää kirjallisuutta sekä verkkojulkaisuja aiheista web-sovelluksen koodaaminen, tietokannan suunnittelu ja käyttöliittymän suunnittelu sekä niihin liittyvistä erilaisista alakäsitteistä. Opinnäytetyön käytännön osuudessa on sovellettu opittua teoriaa käytäntöön. Menetelmät ja tekniikat on valittu aiemman kokemuksen ja alan standardien mukaisesti ja raportissa on pyritty perustelemaan käytettyjen teknologioiden valinta tähän projektiin.

Työn tuloksena saatiin hyödyllinen oppimateriaali tukemaan alan opiskelijoiden koulutusta. Lopputuloksena saatu tuote on valmis ja toimiva, mutta sitä olisi mahdollista tulevaisuudessa myös jatkokehittää. Käytännön osuudessa opin lisää ja sain kokemusta web-sovelluksen kehittämisestä ja sen eri osa-alueista suunnittelusta toteutukseen ja testaamiseen. Opinnäytetyötä voidaan hyödyntää jatkossa bioanalytiikan tutkinto-ohjelmassa sekä työelämässä solujen tunnistamistaidon ylläpitämisessä.

Asiasanat: bioanalytiikka, oppimateriaali, tietovisa, web-ohjelmointi

ABSTRACT

Oulu University of Applied Sciences
Business Information Management, Web application development

Author: Eeli Kivikaarre

Title of thesis: Developing a Quiz Website for Biomedical Laboratory Science Study Material

Supervisor: Pekka Ojala

Term and year when the thesis was submitted: Spring 2019 Number of pages: 27 + 3

The topic of this thesis was to develop a website including a quiz that would serve as a study material for biomedical laboratory students. The objective was to learn more about comprehensive web-development from design to production and finally testing. The topic was chosen, when it became apparent that Finnish material exactly like this didn't exist. The thesis was done in cooperation with two biomedical laboratory science students, who produced the text and image content for the website as their own thesis.

Relevant literature and web articles were used as a source when studying different ways to approach the project. This report aims to explain the use and choice of different technologies and techniques, rather than giving a tutorial in subjects that have been covered various times already. Previous experience was a large factor when choosing the techniques.

As a result, a fully functional website with a quiz was created. The finished product serves as a learning material for biomedical laboratory science students but it can also be used to maintain skills with people already practicing careers in biomedical laboratory science and other relevant areas.

Keywords: biomedical laboratory science, learning material, quiz, web development

SISÄLLYS

1	JOHDANTO	6
2	TEKNOLOGIAT	7
2.1	Web-kehityksen perusteknologiat.....	7
2.2	PHP.....	8
2.3	Kirjastot ja web-sovelluskehukset.....	8
2.3.1	Bootstrap.....	9
2.3.2	jQuery	9
2.4	Web-sovelluskehys	10
2.4.1	MVC-periaate.....	10
2.4.2	CodeIgniter	11
2.5	SQL.....	11
2.6	JSON.....	12
3	SUUNNITTELU.....	13
3.1	Rakenteen ja toiminnallisuuden suunnittelu	13
3.1.1	Hallintapaneeli	13
3.1.2	Tietovisa.....	14
3.2	Käyttöliittymän suunnittelu.....	14
3.3	Sisällön suunnittelu	16
3.4	Tietokannan suunnittelu	16
4	TOTEUTUS	17
4.1	Kehitysympäristö	17
4.2	Web-sovelluksen kehittäminen.....	17
4.2.1	Hallintapaneeli	19
4.2.2	Käyttöliittymä.....	20
5	TESTAUS.....	22
5.1	Lähtökohdat.....	22
5.2	Käytettävyyskysely	23
5.2.1	Tulokset	24
6	POHDINTA.....	25
	LÄHTEET.....	26
	LIITTEET	29

1 JOHDANTO

Tämän toiminnallisen opinnäytetyön aiheena on verkkosivuston kehittäminen bioanalytiikan oppimateriaalia varten. Verkkosivuston pääkohde on tietovisa, jossa käyttäjä tunnistaa verisoluja kuvista. Verisolujen tunnistaminen on olennainen osa bioanalyttikoiden toimenkuvaa, ja tämän opinnäytetyön aihe on valikoitunut kysynnän johdosta – bioanalytiikan opiskelijoiden mukaan tämän tyylistä suomenkielistä materiaalia ei opiskelijoille ole ennen ollut saatavilla.

Opinnäytetyö on toteutettu yhteistyössä kahden bioanalytiikan opiskelijan kanssa. Nämä opiskelijat tuottivat omana opinnäytetyönään sivustolle sisällön, kuten kuvat soluista, oppimateriaalin ja muut sivuston tekstit. Yhteistyö ja konsultointi molempien osapuolten osalta on tärkeää kahden erilaisen alan kohdatessa. Projektilla ei ole ulkoista toimeksiantajaa, mutta tarkoituksena oli alusta asti tehdä tuote, josta olisi hyötyä muille – varsinkin opiskelijoille.

Verisolujen tutkiminen liittyy hematologiaan. Hematologia on veritautien tutkimiseen suuntautunut lääketieteen erikoisala (HUS 2019, viitattu 27.1.2019). Oppimateriaalin tietovisaosuudessa käyttäjä tunnistaa valkosoluja. Valkosolut ovat osa ihmisen immuunijärjestelmää. Valkosolutyyppejä on monia ja niiden esiintymisestä veressä voidaan päätellä asioita ihmisen terveydestä.

Hematologian laboratorioissa on käytössä automaatiota, kuten verisoluanalysointilaitteita. Laboratoriohoitajalta vaaditaan kuitenkin taitoa ja kykyä tunnistaa soluja, sillä analysointilaitteet eivät tunnista kaikkia näytteitä. Tämän lisäksi muun muassa koneen ilmoittamat patologiset löydökset tutkitaan aina mikroskooppilla. (Ek 2009, 9.) Solujen tunnistustaidon ylläpitäminen on tärkeää, ja tämän työn tarkoituksena on tarjota suomenkielinen materiaali verisolujen tunnistamisen avuksi niin opiskelijoille kuin työelämässäkin oleville.

Tietoperustana toimii alan kirjallisuus soveltuvista aiheista. Opinnäytetyössä tutkitaan kokonaisvaltaisen web-sovelluksen toteuttamisen kannalta oleellisia aiheita, kuten käyttöliittymäsuunnittelu, web-arkkitehtuurit, tietokannan suunnittelu ja käyttäjäkokemus. Tietoperustassa on rajattu käytettävät teknologiat ja menetelmät. Ne on koottu aiemman kokemuksen sekä intuition perusteella ja jokaisen teknologian tai menetelmän kohdalla on perusteltu niiden valinta tämän työn pohjaksi.

2 TEKNOLOGIAT

Tässä kappaleessa on selitetty opinnäytetyöprojektissa käytetyt teknologiat ja menetelmät. Web-sovelluksen tuottamiseen on olemassa erilaisia lähtökohtia ja menetelmiä ja kappaleissa pyritään perustelemaan eri teknologioiden valitseminen käytettäväksi juuri tässä projektissa.

Yksi vaihtoehto projektin toteuttamiselle olisi ollut mobiilisovelluksen tekeminen. Ulkoisen toimeksiantajan puuttumisen ja resurssien vähyyden vuoksi parempi vaihtoehto tässä tapauksessa oli kuitenkin verkkosivuston toteutus. Kohderyhmä ja käyttötarkoitus olivat myös keskeisiä syitä valinnalle tehdä verkkosivusto – sen käyttäminen ja esittely on vaivatonta, kun ainoat vaatimukset ovat internetyhteys ja verkkoselain.

Lähteet perustuvat alan kirjallisuuteen. Kirjallisuuden valinnassa on suosittu alalla tunnettuja julkaisijoita sekä lähteiden julkaisuaikaa. Alan standardit voivat muuttua nopeastikin, joten liian vanhojen lähteiden käyttäminen ei ole suotavaa.

2.1 Web-kehityksen perusteknologiat

Kaikkien verkkosivustojen pohjalla on HTML ja CSS. HTML (Hyper Text Markup Language) kuvaillee verkkosivun rakennetta, kun taas CSS (Cascading Style Sheets) on joukko erilaisia tyylimäärittäjiä, joilla kerrotaan selaimelle, miltä sivuston tulee näyttää. Pelkästään näillä kahdella teknologialla voidaan luoda staattisia internetsivuja, mutta dynaamisten sivustojen luomiseen on joukkoon lisättävä muita teknologioita.

JavaScript on dynaamisten verkkosivustojen tekemiseen tarkoitettu kieli. Sen avulla luodaan toiminnallisuutta sivuille. Yksinkertaisimmillaan tämä voi tarkoittaa värien vaihtamista, mutta esimerkiksi jonkun sivuston elementin vetäminen hiirellä paikasta toiseen on toteutettu JavaScriptillä. (Nixon 2014, 323.)

JavaScript on HTML:n ja CSS:n tavoin sisäänrakennettu selaimen ja se pääsee suoraan käsiksi web-dokumentin elementteihin. Se tekee verkkosivuista interaktiivisia muun muassa muokkamalla sisältöä ja reagoimalla tapahtumiin, kuten käyttäjän klikkaamiseen jottain tiettyä elementtiä

tai nappia. Se mahdollistaa tietyn sivuston osan lataamisen uudelleen ilman koko sivun päivittämistä – teknologia, joka on käytössä monissa moderneissa websovelluksissa. (Duckett 2014, 4-7.)

Tietovisan käyttäjälle näkyvä osio ja interaktiivisuus on toteutettu JavaScriptillä. Ilman sitä tietovisaosuus todennäköisesti muistuttaisi enemmän lomakkeen täyttämistä; nyt visassa saadaan kuvat näytettyä yksi kerrallaan ja kokemuksesta pelillisempi. JavaScript on vanha teknologia ja sen käyttäminen tietovisa-verkkosivuston tekemiseen takaa sivuston toiminnallisuuden kaikilla käyttäjillä ja selaimilla.

2.2 PHP

PHP on avoimeen lähdekoodiin perustuva palvelinpuolen ohjelmointikieli, joka tuottaa dynaamista sisältöä verkkosivulle. Se on hyvin suosittu kieli dynaamisten verkkosivustojen luomisessa. PHP on alun perin julkaistu vuonna 1994, ja nykyinen versio eroaa alkuperäisestä hyvinkin paljon. Se on yksi käytetyimmistä palvelinpuolen ohjelmointikielistä ja joidenkin lähteiden mukaan jopa kaikista yleisistä palvelinpuolen ohjelmointikielistä. (Tatroe, MacIntyre, & Lerdord 2013, 1.)

PHP suoritetaan palvelimella, eikä käyttäjä pääse koodiin käsiksi. Kieltä käytetään esimerkiksi tietokantakyselyihin (Nixon 2014, 45). Koska PHP:lla on pääsy palvelimelle, sen tiedostoihin ja tietokantoihin, on tietoturvan oltava kohdallaan. Turvallisuutta lisääviä ominaisuuksia tuo myös tässäkin projektissa käytetty CodeIgniter-sovelluskehys, josta tarkemmin seuraavissa kappaleissa.

PHP:n käyttäminen tässä opinnäytetyössä perustuu siihen, että kaikki projektin suunnitteluvaiheissa määritellyt toiminnallisuudet saadaan toteutettua sillä. Se on yksinkertainen, tehokas ja joustava kieli. Suuren käyttäjäkunnan ansiosta tukea erilaisiin ongelmiin on saatavilla paljon, joka tuo helpoutta kehittämistyöhön. PHP:ta on käytetty ja opetettu Oulun ammattikorkeakoulun kursseilla, ja tämä oli yksi tärkeä edellytys teknologian valitsemiselle.

2.3 Kirjastot ja web-sovelluskehukset

Websovelluksen kehitystyössä vastaan voi tulla termejä, kuten kirjasto (library) ja sovelluskehys (framework). Kirjastoista puhutaan web-kehityksessä usein, kun tarkoitetaan jotain kokoelmaa JavaScriptille tehtyjä funktioita ja muuta toiminnallisuutta. Sovelluskehys on laaja termi, jota käytetään

monessa yhteydessä. Kirjastojen ja sovelluskehysten perimmäinen tarkoitus on säästää aikaa sekä kehityksessä, että suunnittelussa. (Meiert 2015, 2.)

2.3.1 Bootstrap

Bootstrap on suosittu verkkosivujen muotoiluun ja tyyllittelyyn tarkoitettu ilmainen avoimen lähdekoodin frontend-sovelluskehys. Se sisältää paljon ominaisuuksia sivuston muotoiluun ja sen avulla sivusto on vaivatonta saada skaalautumaan eri laitteille. (Bootstrap 2019, viitattu 27.1.2019.)

Opinnäytetyössä käytettiin Bootstrapin versiota 3, vaikka tuorempi versio (4) oli jo ollut suhteellisen pitkään saatavilla. Aiemman version käyttäminen perustui kuitenkin kattavaan aiempaan kokemukseen. Uusimmassa versiossa oli paljon uudistuksia toiminnassa, kuten esimerkiksi HTML ”flexible box” -elementtien käyttäminen. Toinen tärkeä syy, miksi tässä projektissa käytettiin vielä vanhempaa versiota oli se, että kaikki käytetyt työkalut eivät tukeneet Bootstrap 4:n moderneja ominaisuuksia. Ohjelmointiympäristö Netbeans 8.2 ei tukenut CSS-muuttujia, joita Bootstrap 4:ssä käytetään, joten oli helpompi käyttää vanhaa tuttua versiota kuin alkaa suunnittelemaan ohjelmointiympäristön vaihtamista.

Bootstrap nopeuttaa sivuston rakenteen ja ulkoasun tekemistä huomattavasti. Sen avulla toimivan ja suhteellisen hyvännäköisen prototyypin valmistaminen oli vaivatonta ja sovellusta sekä sen tärkeimpiä ominaisuuksia pääsi nopeasti testaamaan ja kehittämään.

2.3.2 jQuery

jQuery on JavaScript-kirjasto, joka yksinkertaistaa monia JavaScript-koodaamiseen liittyviä asioita. Sen avulla esimerkiksi elementtien valitseminen ja niiden manipulointi yksinkertaistuvat. jQueryn ja perinteisen JavaScript-koodin yhdistäminen on tehokas tapa lisätä toiminnallisuutta sivustolle. Sen käyttäminen varmistaa myös toiminnallisuuden eri selaimilla. (Duckett 2014, 294, 297.)

jQuery on hyvin suosittu JavaScript-kirjasto. On kehitetty uusia, tehokkaampia teknologioita dynaamisten verkkosivujen kehittämiseen (esimerkiksi React), mutta suosittu teknologian hyödyt – kuten kattava dokumentaatio ja materiaali – helpottivat sen valintaa tähän projektiin. jQueryn valintaan käytettäväksi teknologiaksi vaikutti myös aiempi tuntemus siitä.

2.4 Web-sovelluskehys

Web-sovelluskehys eli web framework on työkalu web-sovelluksen tekemiseen. Se sisältää valmiiksi web-sovelluksen toiminnallisuuden kannalta tärkeitä ominaisuuksia ja komponentteja, jolloin niitä ei tarvitse itse koodata. (Ryabtsev 2016, viitattu 21.1.2019.) Tavallaan sovelluskehys on pohja sovellukselle, jonka päälle oma toiminnallisuus voidaan rakentaa muokkaamalla kehystä ja lisäämällä ominaisuuksia siihen.

Sovelluskehysten käyttäminen ei ole aina paras ratkaisu. Meiert ilmaisee hyvin kirjassaan (2015), kuinka monille kehittäjille on vaikea luopua tutusta ja turvallisesta sovelluskehystä, vaikka se olisi kuinka rikki, epäonnistunut tai vanhentunut. Tunnistin itsestäni tällaisen henkilön piirteitä käyttäessäni esimerkiksi vanhempaa Bootstrapia siksi, että se on tuttu ja turvallinen.

Web-sovelluskehysten käyttäminen helpotti ja nopeutti web-sovelluksen tekemistä. Tämän tekniikan käyttäminen tässä projektissa ei olisi ollut välttämätöntä, mutta siihen päädyttiin silti. Helpottamisen lisäksi yksi suurimpia syitä oli järjestelmällisyyden säilyttäminen. Kun perustoiminnallisuutta ei tarvinnut tehdä alusta asti itse, jäi aikaa ja energiaa sovelluksen päätoiminnallisuuden ja sen kaikista olennaisimpien ominaisuuksien kehittämiseen.

2.4.1 MVC-periaate

MVC-lyhenne tulee sanoista "model", "view" ja "controller" (malli, näkymä, ohjain). Suosituimpien sovelluskehysten rakenne muodostuu erilaisista tasoista, jolloin tietyn toiminnallisuuden laajentaminen on mahdollista. Joustavuus on toinen tärkeä sovelluskehysistä saatava hyöty. (Ryabtsev 2016, viitattu 21.1.2019.)

MVC:ssä malli sisältää kaiken datan, näkymät ovat vastuussa datan visuaalisesta representaatiosta ja ohjaimen tehtävä on toimia näiden kahden välikätenä ja esimerkiksi muokata dataa erilaisiin käyttötarkoituksiin sille annettujen käskyjen mukaisesti. Nämä kolme elementtiä muodostavat pohjan monelle sovelluskehyselle ja niiden oikeaoppisella hallinnalla voi välttää ongelmat ja virheet sovelluksen ajamisessa. (Ryabtsev 2016, viitattu 21.1.2019.)

2.4.2 CodeIgniter

CodeIgniter on palvelinpuolen (backend) web-sovelluskehys, joka on koodattu PHP:lla. Se perustuu löyhästi MVC-periaatteeseen. Tämä tarkoittaa, että web-sovellusten koodaaminen onnistuisi sillä tarvittaessa vain kontrollereita käyttämällä. CodeIgniter on hyvin joustava, mutta selkeän rakenteen noudattaminen helpottaa projektin organisointia ja ylläpitoa. (Ezell 2016, 1.)

Tässä projektissa on kuitenkin noudatettu MVC-mallin periaatteita – suurimpana syynä projektin hallinnan helpottaminen selkeämmän struktuurin avulla. Pelkkiä kontrollereita käyttämällä yksittäisen tiedoston pituus kasvaisi suhteettoman suureksi ja kehitystyö voisi vaikeutua. Keskeisimpiä syitä päätökselle käyttää CodeIgniteria tässä opinnäytetyössä olivat muun muassa sivupohjien tekeminen, URL-reititys ja tietokantakomennot. Esimerkiksi URL-reititys mahdollistaa käyttäjä- ja hakukoneystävällisemmät verkko-osoitteet. Tässä tapauksessa sivujen kokonaislukumäärä ei ole suuri, mutta linkkien jakaminen sivun eri alueille on joka tapauksessa siistimpää ja helpompaa.

2.5 SQL

Opinnäytetyössä päädyttiin käyttämään tietokantaa datan tallentamiseen. Tietokannan käyttäminen on yleinen käytäntö tämän tyyllisissä projekteissa, jossa suurta tai suurehkoa datamäärää tulee voida dynaamisesti hallita.

SQL on relaatiotietokantojen manipuloimiseen suunniteltu kieli. Tietokannalla tarkoitetaan yksinkertaisesti toisiinsa liittyvän tiedon kokoelmaa. Esimerkiksi puhelinluettelo ja paperiset potilastiedot ovat tietynlainen tietokanta. Manuaalisten datantallennusjärjestelmien heikkouksiin kuuluu datan etsimisen vaikeus. Internetissä sijaitseva elektroninen tietokanta antaa mahdollisuudet datan hallintaan ja oikeiden tietojen löytäminen sekä ajan tasalla pitäminen ovat hyvin paljon nopeampaa. (Beaulieu 2009, 1.)

Relaatiomallissa data on jaettu erillisiin, toisiinsa liittyviin taulukoihin. Taulukoita voivat esimerkiksi olla asiakas, tuote, tili. Yhden tietokantataulun ideana on sisältää vain sille olennaisia tietoja ja muut tiedot on eritelty omiin tauluihinsa. SQL kulkee käsikädessä relaatiomallin kanssa, sillä SQL-kyseilyn tulos voidaan myös nähdä tauluna. (Beaulieu 2009, 7.)

```
SELECT id, nimi, kuva FROM solu WHERE solutyyppi="Basofiili" ORDER BY nimi ASC;
```

KUVIO 1. Esimerkki SQL-lauseesta

Tietokannan käyttäminen tässä projektissa oli aiheellista, sillä sivustolle on tarkoitus tallentaa tietoa, mutta sitä pitää olla myös mahdollista muokata. Tietokannan hallintaa varten palvelimelle asennettiin phpMyAdmin-hallintatyökalu. Se on selaimen kautta käytettävä visuaalisen käyttöliittymän omaava ohjelmisto, joka tekee tietokantojen ja datan lisäämisestä sekä muokkaamisesta tässä projektissa kätevämpää. Helppo varmuuskopiotoiminnallisuus oli helpomman hallinnan lisäksi syy käyttää phpMyAdminia.

2.6 JSON

JSON, eli JavaScript Object Notation, on datan merkitsemisstandardi. Sitä tukevat kaikki modernit ohjelmointikielet, tässä projektissa käytettyjen JavaScriptin ja PHP:n lisäksi esimerkiksi Java, Ruby, C# ja Python. JSON on muodoltaan yksinkertainen: se sisältää ohjelmoijalle helppoja rakenteita kuten olioita, taulukoita sekä avain–arvo pareja. JSON on yksinkertaisuutensa ansiosta hitaasti syrjäyttämässä XML-standardia. (Marrs 2017, 7.)

```
1  {
2      "avain": "arvo",
3      "solut": ["Punasolu", "Valkosolu"],
4      "tietovisa": {
5          "Helppo": {
6              "vaikeus": "helppo",
7              "lukumäärä": 10
8          },
9          "Vaikea": {
10             "vaikeus": "vaikea",
11             "lukumäärä": 20
12         }
13     }
14 }
```

KUVIO 2. Esimerkki JSON-datasta

JSON:ia käytetään tässä projektissa tietovisan käyttöliittymässä. PHP hakee tietyn määrän kyselydataa tietokannasta, muuntaa sen JSON-formaattiin ja tallentaa JavaScript-muuttujaan. Tämän jälkeen tietoa on helppo käsitellä JavaScriptillä ja esimerkiksi tarkistaa, oliko käyttäjän vastaus oikein.

3 SUUNNITTELU

3.1 Rakenteen ja toiminnallisuuden suunnittelu

Verkkosivuston suunnitteluvaiheessa mietittiin, mitä ominaisuuksia sivustolta on löydettävä. Esi-merkiksi alusta asti oli selvää, että sivuston ylläpitäminen on tehtävä helpoksi, koska joku muu kuin kehittäjä lisää sisällön sivulle.

Käyttäjälle näkyvä rakenne sisältää vain muutaman sivun. Etusivun lisäksi erillisiä sivuja ovat oppimateriaali sekä itse tietovisa. Kaksi ensimmäistä ovat staattisia sivuja, mutta tietovisasivun sisältö tuotetaan dynaamisesti ja sen sisältö on joka latauskerralla erilainen.

Yksinkertainen rakenne on hyödyllinen, sillä ihmiset eivät läheskään aina lue aivan jokaista sivun elementtiä ja tekstiä läpi, vaan useimmiten peruskäyttäjä vilkaisee sivun nopeasti läpi ja klikkaa linkkiä, josta on kiinnostunut tai on alun perinkin etsimässä. Ihmiset eivät lue sivuja, vaan he skannaavat ne, kuten käytettävyyden ja suunnittelun ammattilainen Steve Krug kirjoittaa (2014).

Sivuston suunnittelussa tärkeää on tuoda selkeästi esille heti etusivulla sivuston identiteetti ja misio. Sivuhierarkian olemassaolo ja esittäminen ovat myös tärkeitä asioita. Näiden avulla käyttäjä tietää, mitä sivulla on tarjota ja mitä hän voi tehdä siellä. (Krug 2014, 88.)

3.1.1 Hallintapaneeli

Sivuston pääsisältö muodostuu solukuvista. Tiedossa oli jo alusta asti, että kuvien lukumäärä tulee olemaan suuri. Oli siis tärkeää kehittää helppo ja hallittava järjestelmä kuvien lisäämiselle, muokkaamiselle ja poistamiselle.

Koska staattisia sisältösivuja ei ollut kuin kaksi (etusivu ja oppimateriaali) ja konsultoinnin perusteella oppimateriaalia ei tulevaisuudessa ole heti tarpeellista päivittää, ei hallintapaneeliin tarvinnut suunnitella toiminnallisuutta yksittäisen sivun tekstisisällön muokkaamiseen. Esimerkiksi oppimateriaalin sisältö (solutyyppien kuvaukset) eivät todennäköisesti tule muuttumaan tulevaisuudessa.

3.1.2 Tietovisa

Tietovisa-sivun rakenteeseen päädyttiin erilaisten tietovisapelien tutkimisen ja oman aiheeseen liittyvän kokemuksen sekä intuition perusteella. Tärkeä osa rakenteessa tällä sivulla on tulokset-sivun toteutus, joka on oppimisen kannalta olennainen käyttäjälle.

Sivuston testausvaiheessa saatujen tulosten avulla tietovisasta saatiin hiottua käyttäjäystävällinen ja kohderyhmää hyvin palveleva toteutus. Testauksesta ja tuloksista tarkemmin tämän raportin loppupuolella.

3.2 Käyttöliittymän suunnittelu

Käyttöliittymäsuunnittelun lähtökohtia olivat modernisuus, selkeys ja käytännöllisyys. Helppokäyttöisyys oli myös tärkeä ominaisuus. Mobiililähtöisyys oli tässä projektissa tärkeä näkökulma ja tämä otettiin huomioon sivuston ulkonäköä ja käyttöliittymää suunnitellessa. Steve Krug kertoo kirjassaan (2014, 25), että käytettävyyden ensimmäinen sääntö on ”don’t make me think” – ”älä laita minua ajattelemaan”. Tämä tarkoittaa, että sivuston tulee olla niin itsestään selvä, että käyttäjä tietää automaattisesti, mitä tehdä tai mistä painaa.

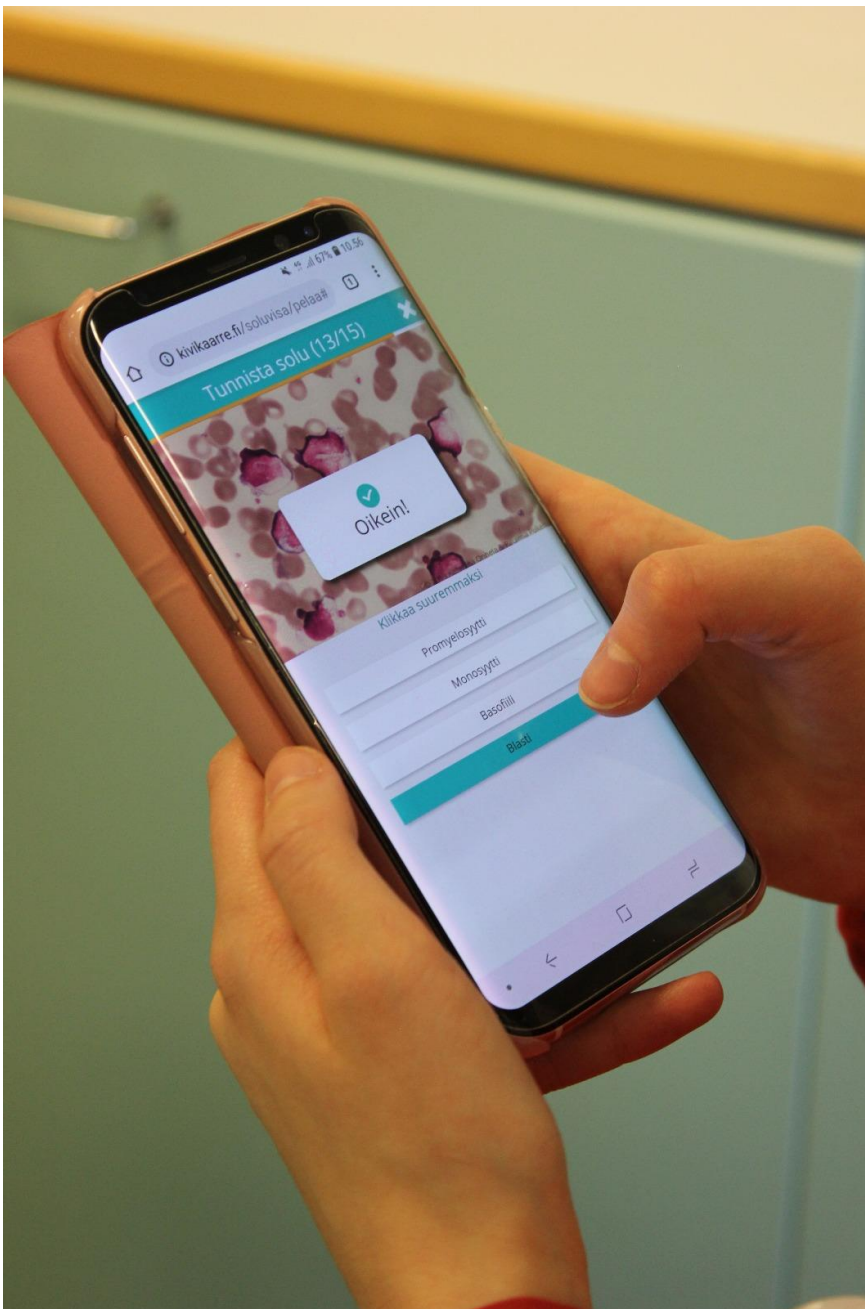
Oppimateriaaliverkkosivuston rakenne on käytettävyyttä ajatellen helppo toteuttaa, sillä sisältöä sivustolla ei ole liian paljon. Sivuston sisältö on hyvin tarkkaan määriteltä, ja kohderyhmään kuuluva käyttäjä todennäköisesti tietää mitä on hakemassa ja osaa hakeutua oikeaan paikkaan.

Sivuston suunnitteluvaiheessa käytettiin apuna rautalankamallia. Siinä on tarkoitus hahmotella sivuston eri elementtien sijainti ilman mitään tyyliä, ainoastaan ääriviivojen avulla. Tämä auttaa hahmottamaan kokonais kuvaa.

Rautalankamallin jälkeen sivuston ulkoasu luonnosteltiin kuvankäsittelyohjelmassa, ja esimerkiksi sivuston väripaletti ja yleinen teema valittiin. Ulkoasuun ei käytetty aivan niin paljon aikaa kuin tekniseen puoleen, ja Bootstrapin käyttäminen osoittautui eduksi. Sen avulla sivuston käyttöliittymän rakentaminen toimivaksi oli vaivatonta. Esimerkiksi navigaatiopalkki voisi alusta asti itse tehtynä

viedä aikaa. Bootstrapin avulla saadaan helposti toteutettua valikko, joka toimii kaikilla laitteilla ja selaimilla.

Eniten suunnittelua vaati itse tietovisan ulkoasu ja tässä asiassa kuunneltiin bioanalyttikko-opiskelijoiden toiveita. Mobiilinäytöllä asettelu on hyvinkin tyypillinen tietovisalle, jossa arvuutellaan kuvia. Samantyyliä sovelluksia tutkimalla ja testaamalla sivustolle saatiin suunniteltua toimiva ja helpokäyttöinen käyttöliittymä.



KUVIO 3. Tietovisan mobiilikäyttöliittymä

3.3 Sisällön suunnittelu

Tähän opinnäytetyöhön ei kuulunut sisällön tuottaminen. Oli kuitenkin olennaista suunnitella ja selvittää, millaista sisältöä sivustolle on tarkoitus laittaa ja etenkin, miten ne asetellaan. Tarvittavan tiedon määrittäminen on vaatimuksena myös tietokannan suunnittelussa.

Tämän työn tapauksessa kyseessä oli oppimateriaali, joten oletettavissa oli alusta lähtien, että tekstiä tulee olemaan paljon. Tiedossa oli myös – kuten jo aiemmin on käynyt ilmi – että kuvat ovat tärkeässä osassa sivuston sisältöä.

3.4 Tietokannan suunnittelu

Suuren datamäärän hallintaan tarvitaan tietokanta, jonka rakenne täytyy suunnitella. Ennen datan lisäämistä oli tärkeää selvittää, minkälaista tietoa sivulle on tarkoitus tallentaa ja miten sitä tarvitsee saada esille.

Tietokantaa suunnitellessa käytettiin apuna visuaalisia keinoja, kuten kaavioiden piirtämistä. Tämä helpottaa datan ja niiden välisten suhteiden hahmottamista. Kaavioiden piirtäminen olikin tässä opinnäytetyössä tehokkain keino luoda tietokanta.

Tietokannasta tuli alkuperäistä suunnitelmaa suoraviivaisempi. Tietokannan lopullisessa versiossa ovat taulut soluille, solutypeille, kuville ja käyttäjille. Myöhemmin sovellukseen lisättiin tilastoille tarkoitettu ”statistics”-taulu. Näistä kolme ensimmäistä ovat varsinaiseen tietovisaan liittyviä tauluja ja käyttäjätaulu sisältää hallintapaneelin kirjautumistiedot. Alun perin tauluja oli suunniteltu myös eri sivujen sisällöille niiden muokkaamista varten, mutta kun sivujen muokkaamissuunnitelmasta luovuttiin, ei näille tauluille enää ollut tarvetta ja tietokannasta tuli paljon yksinkertaisempi.

4 TOTEUTUS

Opinnäytetyön aloituskeskustelu käytiin lokakuussa 2018, ja itse tuotteen suunnittelu ja toteutus aloitettiin pian sen jälkeen. Opinnäytetyö saatiin tammikuussa 2019 siihen vaiheeseen, että sisälönsyöttöä voitiin alkaa suorittamaan. Sivuston osoite voi tulla muuttumaan opinnäytetyön kehittämisen jälkeen, mutta tämän raportin kirjoitusvaiheessa solujen tunnistusvisa -verkkosivusto löytyy osoitteesta <http://kivikaarre.fi/soluvisa/>. Sivuston kehitysvaiheessa käytetty Soluvisa-nimi jäi pysyväksi projektille ja tämän raportin lukija löytänee tulevaisuudessa sivuston parhaiten käyttämällä ”soluvisa”-hakusanaa Googlessa, mikäli käytetty osoite lakkaa toimimasta. Tavoitteena on saada pidettyä verkkosivusto käytössä mahdollisimman pitkään.

4.1 Kehitysympäristö

Projektin alkuvaiheessa hankittiin kehitysympäristö sovelluksen kehittämistä varten. Kehityksen ajaksi projektille saatiin Oulun ammattikorkeakoululla testikäytössä oleva Microsoft Azure -pilvipalvelin. Tämän raportin kirjoitusvaiheessa ei ole varmaa, kuinka pitkäaikainen ratkaisu Azuren käyttäminen on tämän opinnäytetyöprojektin ylläpitämiselle ja on mahdollista, että tulevaisuudessa sivuston ylläpitämiselle joudutaan hankkimaan joku toinen palvelinratkaisu.

Debian Linux -pohjaiselle palvelimelle asennettiin Apache2-palvelinohjelmisto, PHP, MariaDB-tietokanta sekä phpMyAdmin-tietokannanhallintatyökalu. Tätä kuvailtua toteutusta voisi sanoa melko perinteiseksi tavaksi ylläpitää verkkosivustoa, ja nykyaikaisten standardien tasolle sen tuo tässä projektissa pilvipalvelimen käyttäminen.

4.2 Web-sovelluksen kehittäminen

Muutamaa ulkoasukokeilua ja muuta suunnitteluvaiheen testausta lukuun ottamatta sovellus rakennettiin alusta asti CodeIgniter-sovelluskehiksen päälle. CodeIgniterin asentaminen on yksinkertainen prosessi. Ladattava zip-paketti puretaan kokonaisuudessaan palvelimelle, ja asetustiedostoon määritetään polut, verkko-osoite ja tietokannan käyttäjätunnukset.

Etusivu rakennettiin ensimmäisenä, sillä siinä samalla oli helppo määritellä kaikki perustyyliin kuntoon kuten värit, fontit sekä yleisten HTML-elementtien tyyli. Sivustolle antaa ammattimaisempaa kuvaa, kun kaikki etäisyydet ovat tasaiset jokaisella sivulla.

Tietovisaosuudessa kuvat olivat tärkeässä osassa ja niihin liittyvä toiminnallisuus tuli saada toimivaksi. Esimerkiksi mobiililla pienemmän näytön takia kuvien suurentaminen on tärkeä ominaisuus. Onneksi kuvien katseluun, suurentamiseen ja liikuttelemiseen ruudulla oli olemassa Photoswipe-JavaScript-gallerialisäosa. Se mahdollisti kuvien helpon zoomaamisen ja sormenliike-oleet (esimerkiksi zoomaaminen "nipistämällä").

Kun käyttäjä siirtyy tietovisasivulle, hakee PHP-koodi tietokannasta tietyn määrän soluja tietokannasta, ja niihin liittyvät kuvat. Kysymykset muodostetaan myös palvelimella. Solu-aulussa jokaiselle solulle on määritelty oikea vastaus ja muut vastausvaihtoehdot määritetään satunnaisesti. Soluille on mahdollista lisätä niin sanottuja oletusvastauksia (joidenkin solujen tapauksessa kuva voi muistuttaa paljon jotain tiettyä solua, jolloin se kannattaa esittää aina vastausvaihtoehdoissa.)

Sovellus hakee taulukkoon 15 solua vastauksineen ja tulostaa ne sivulle. Koko tietovisa on siis sivulla kerrallaan, ja käyttäjälle esitetään yksi kysymys kerrallaan loppujen ollessa piilotettuina. Toinen vaihtoehto olisi ollut ladata solut ja vastausvaihtoehdot tietokannasta yksi kerrallaan sivulle AJAX-tekniikkaa hyväksi käyttäen. Käytetty menetelmä on kehittäjästä riippuvainen ja tässä projektissa päädyttiin käyttämään kuvailtua tapaa lähinnä oman tottumuksen, ohjelmointityylin ja intuition takia.

Toiminnallisuus kysymyksiin vastaamiseen on toteutettu JavaScript-funktiolla jQueryä hyödyksi käyttäen. Käyttäjälle näkyvän interaktiivisuuden (kuten animaatioiden ja infotekstien) lisäksi kysymyksiin vastatessa vastaus tieto tallennetaan erilliseen taulukkoonsa ja samalla seurataan soluvastauksen edistymistä. Tästä vastaus taulukosta on hyötyä pisteitä laskettaessa ja kyselyn lopuksi näytettävässä tulokset-näkymässä. Tulokset näkymä oli tärkeä ominaisuus, sillä bioanalyttikko-opiskelijoiden mukaan oppimisen kannalta on hyödyllistä nähdä omat vastaukset, oikeat vastaukset ja mahdolliset selitykset, miksi vastaus on väärin. Selitysteksti toivottiin myös esiintyvän jokaisen väärin vastauksen jälkeen, sillä käyttäjä halunee tietää, miksi vastaus oli väärin siinä vaiheessa, kun kuva on vielä tuoreessa muistissa.

4.2.1 Hallintapaneeli

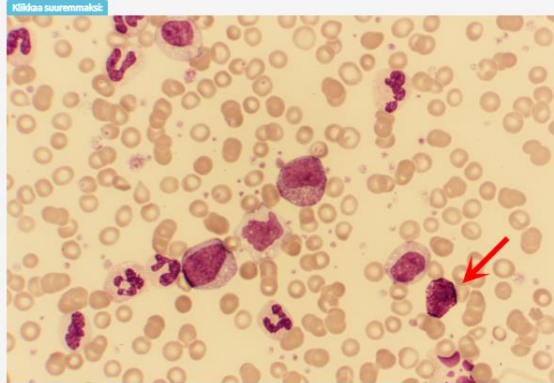
Hallintapaneelin luominen verkkosivustolle voi laajimmillaan olla todella suuri projekti, mutta onneksi tässä tapauksessa vaadittua toiminnallisuutta ei ollut liikaa. Hallintapaneelissa tarvittiin toiminnallisuus kuvien lisäämiselle ja niiden liittämiseksi johonkin tiettyyn solutyyppiin.

Vaikka hallintapaneelin käytettävyys tuli olla selkeä, eivät käytettävyysvaatimukset olleet yhtä vaativat kuin vierailijoille näkyvässä osuudessa. Hallintapaneelin käyttöä varten annettiin käyttäjäkoulutus sisällöstä vastaaville bioanalyttikko-opiskelijoille ja koulutuksessa hallintapaneelin käytettävyydestä saatiin hyvää palautetta.

Solujen, solutyypin ja kuvien lisäämisen ja hallinnan lisäksi hallintapaneelista löytyy toiminnot salasanan vaihtamiseen ja uuden ylläpitäjän lisäämiseen – joskin viimeinen ominaisuus on ainakin tällä hetkellä ainoastaan sivuston kehittäjän hallinnassa.

Takaisin kuvan valintaan

Solu



Mikä solu?

Atyypinen lymfosyytti

Oletusvaihtoehdot

Määritä halutessasi vastausvaihtoehdot, jotka näkyvät kysymyksessä. Jos ei määritetty, vaihtoehdot valitaan satunnaisesti.

Lisää +

Kysymys

(valinnainen)

Oletuskysymys: "Tunnista solu". Muuta jos haluat määritellä tarkemmin, esimerkiksi: "Tunnista oikeanpuolimmainen solu".

Selitys

(valinnainen)

Testin päätteeksi näytetään mahdollinen selitys esimerkiksi mistä tämän solun tunnistaa tai miten se eroaa toisista lähes vastaavanlaisista.

Tallenna

KUVIO 4. Hallintapaneelin solunlisäysnäköymä

4.2.2 Käyttöliittymä

Käyttöliittymän toteutuksessa käytettiin apuna paljon Bootstrapin tarjoamia tyyliä. Bootstrapin dokumentaatio tarjoaa esimerkiksi navigaation tekemiseen hyvän pohjan, josta on helppo muokata CSS-tyyleillä teemaan sopiva. Sivuston rakenteesta tuli suhteellisen yksinkertainen jo pelkästään siitä syystä, että valtavasti sisältöä sivulla ei ole.

SOLUVISA Etusivu Testi Opiskelumateriaali

VERISOLUJEN TUNNISTUSVISA

Bioanalytiikan oppimateriaali

[Testaa taitosi!](#)

[Oppimateriaali](#)

Tietoa

Tämä oppimateriaali on tehty opinnäytetyönä Oulun ammattikorkeakoulun bioanalytiikan ja tietojenkäsittelyn tutkinto-ohjelmassa. Materiaalimme on tarkoitettu bioanalytikoille. Verkkosivulla voit harjoitella solujen tunnistamista tai kerrata tietoasi verisoluisia.

Pelaa

Opi tunnistamaan verisoluja tietovisan avulla. Voit valita visan kahdesta eri kategoriasta: kypsät solut ja kaikki solut.

[Kaikki solut](#)

[Kypsät solut](#)

Oppimateriaali

Voit kerrata tietoasi soluista oppimateriaalista ennen tietovisan tekoa. Oppimateriaali sisältää kaikkien solujen tärkeimmät tunnistuspiirteet sekä esimerkkikuvia.

[Siirry oppimateriaaliin](#)

Soluvisa
Oppimateriaali
Tietovisa

Linkejä
Oulun Ammattikorkeakoulu
Bioanalyttikolliitto
Labquality

Kuvien käyttäminen ilman lupaa kielletty.
© 2019 Katarina Onnela & Pauliina Pukema

KUVIO 5. Etusivun lopullinen ulkoasu

5 TESTAUS

Testaus on olennainen osa sovelluskehitystä. Tämän tuotteen tapauksessa testattiin käytettävyyttä, suorituskykyä ja tietoturvaa. Sivustoa testattiin eri laitteilla ja selaimilla mahdollisimman laajan tuen takaamiseksi.

Selaimet, joilla sivua testattiin:

- Chrome
- Firefox
- Internet Explorer 11
- Edge
- Safari

Testaukseen käytetyt laitteet ja käyttöjärjestelmät:

- PC – Windows 10
- MacBook – MacOS 11
- Samsung Galaxy S8+ – Android

5.1 Lähtökohdat

Käytettävyyden testaaminen oli tärkein asia tämän projektin testaamisessa. Muita huomioita, joihin testauksessa kiinnitettiin huomiota, olivat tietoturva ja virheet eli bugit.

Tietoturvallisuuden näkökulmasta mitään uhkia ei havaittu. CodeIgniterin tarjoamat ja käyttämät tietoturvaominaisuudet luovat turvallisuutta ja estävät hyökkäjiä käyttämästä helpompia ja perinteisimpiä hyökkäystapoja, kuten SQL-injektiota, XSS-hyökkäyksiä ja "Remote code execution" -hyökkäyksiä vastaan. Sivusto ei tallenna käyttäjästään mitään tietoa, joten haavoittuvuuden ilmeessä suurta vahinkoa ei pääsisi tapahtumaan.

Suorituskyvyn testaaminen tapahtui Chrome-selaimen tarjoamilla kehittäjätyökaluilla. Paljon kuvia sisältävällä sivustolla suorituskyvyn heikkeneminen voi näkyä hitaana sivun latausaikana. Hallintapaneelin avulla tietovisaan lisättävät kuvat optimoidaan CodeIgniterin kuvamanipulaatiokirjaston

avulla. Testausvaiheessa ennen sisällönsyöttöä piti löytää asetukset, joilla kuvat ovat tarpeeksi tarkkoja, mutta myös tiedostokooltaan kompakteja. Solukuvien tiedostokoko on noin 200 kilotavua per kuva, joka pitäisi olla modernilla internetnopeudella sopivan kokoinen.

Tietovisa-sivulla kuvien koon tuoma rasitus näkyy hitailla laitteilla eniten, sillä kuvia täytyy ladata 15 kappaletta, joka tekee yhteensä jopa kolme megatavua. Tämäkin on vielä aivan normaali koko modernille verkkosivulle, mutta jos kuvien koko olisi ollut yhtään suurempi, olisi niiden latauksessa pitänyt miettiä jotain muuta vaihtoehtoa kuin kaikkien kuvien lataamista kerralla selaimen muistiin – esimerkiksi jonkinlaista ”lazyload”-kirjastoa, joka lataa kuvat vasta siinä vaiheessa, kun ne näkyvät käyttäjän ruudulla.

Virheiden ja bugien etsiminen ja korjaaminen tapahtui kehityksen kanssa samaan aikaan, eli mahdollisten virheiden korjaaminen tapahtui heti niiden ilmetyä kehitysvaiheessa. Verkkosivuston valmistumisen jälkeen virheitä ei ole tullut ilmi ja sivustoa ei pitäisi normaalitilanteessa saada rikottua. Sivun käyttöliittymän ollessa tehty JavaScriptillä, voisi edistynyt käyttäjä periaatteessa saada rikottua tietovisan käyttämällä selaimen kehittäjäkonsolia ja työkaluja. Normaalilla käytöllä sivusto vaikutti testaamisen perusteella varmatoimiselta.

5.2 Käytettävyyksely

Käytettävyytestaukseen on olemassa monia eri metodeja. On tärkeää testata tuotteita ulkopuolisilla testihenkilöillä, sillä monesti varsinkin ohjelmisto- ja webkehityksessä voi kehittäjälle iskeä ”sokeus”, eikä tämä välttämättä huomaa tuotteessa esille tulevia käytettävyyso ongelmia. Kehittäjä tietää aina tarkalleen, mistä painaa ja mitä tehdä, mutta ulkopuoliselle tämä ei välttämättä ole niin itsestäänselvää.

Käytettävyytestausta olemme harjoitelleet Oulun ammattikorkeakoulussa kurssilla, jossa testasimme eri verkkosivujen käytettävyyttä erillisessä käytettävyysslaboratoriossa. Käytettävyyden testaukseen löytyy monenlaisia kehittyneitäkin työkaluja, ja niillä voidaan seurata esimerkiksi käyttäjän silmän liikkeitä. Näin vaativaa testausta tässä projektissa ei tällä kertaa suoritettu, vaikka niistä saatavat tulokset olisivat voineet olla hyvinkin mielenkiintoisia. Opinnäytetyössä päädyttiin tekemään käytettävyyksely lomakemuodossa. Sivusto annettiin testattavaksi sen oikealle kohderyhmälle, eli bioanalyttikko-opiskelijoille.

5.2.1 Tulokset

Kyselyyn saatiin kuusi vastausta ja siitä oli hyötyä sivuston kehityksessä. Kommentteja saatiin tietovisan toiminnallisuudesta ja responsiivisuudesta. Esimerkiksi kuvien vaihtuminen väärin vastatessa kesti palautteiden perusteella liian pitkään. Lisäksi animaatioiden nopeus oli aluksi liian hidas, mikä aiheutti tuskastumista joillekin käyttäjille. Kyselyä ja pelikokemusta saatiin vastausten perusteella kehitettyä sulavammaksi.

Olen tyytyväinen käytettävyydestäukseen ja siitä oli hyötyä projektille. Testausta olisi voinut suorittaa laajemminkin erilaisilla metodeilla, mutta koen, että tässä tapauksessa testaus oli riittävä ja sen perusteella sivustoa saatiin kehitettyä toimivammaksi, eikä vastaan ole tullut enempää puutteita toiminnallisuudessa tai käytettävyydessä.

6 POHDINTA

Opinnäytetyössä valmistunut tuote onnistui omasta mielestäni hyvin; suurimpana syynä se, miten hyvin se onnistui vastaamaan ulkopuolisiin odotuksiin. Projekti oli haastavuudeltaan sopivalla tasolla ja se eteni odotusten mukaisesti. Itse tuotteen, eli verkkosivun, suunnittelussa ja koodaamisessa oli yhdelle henkilölle työtä juuri sopivasti.

Työtä olisi mahdollista jatkaa pidemmälle lisäämällä ominaisuuksia tietovisaan. Esimerkiksi oman edistymisen seuraaminen, työkalut opettajille luoda tietovisoja esimerkiksi kurssimateriaaliksi ja seurata opiskelijoiden menestystä niissä, sekä muut toiminnalliset tai visuaaliset parannukset. Opinnäytetyössä projektia piti kuitenkin rajata, jotta yhdelle henkilölle ei tulisi liian suuri työmäärä. On mahdollista, että vaikkapa suuren kysynnän takia tulevaisuudessa tuotetta kehitetään tai siitä tehdään uusi versio.

Opinnäytetyöprojekti eteni hyvin, vaikka alkuperäisestä aikataulusuunnitelmasta jäätettiin. Asioita, mitkä johtivat opinnäytetyön valmistumisen viivästymiseen, olivat esimerkiksi samaan aikaan suoritettu ammattikorkeakoulun ammattiharjoittelu. Mainittu harjoittelu vaikutti enemmän raportin kirjoittamisen aikatauluun ja itse tuote, tietovisan sisältävä web-sovellus, valmistui hyvissä ajoin.

Projekti saatiin vietyä loppuun ilman mitään suurempia ongelmia. Yhteistyö opinnäytetyössä kahden eri alan välillä onnistui hyvin ja tärkein asia, eli kommunikaatio, toimi moitteitta. Sujuvan kommunikaation avulla tuotteesta saatiin toivotunlainen.

Opinnäytetyön ohjausseminaarissa itse tuote oli jo lähes valmis, ja raportti noin puolessa välissä. Seminaarin kommenttien ja muistion perusteella raporttia saatiin korjattua ja vietyä oikeaan suuntaan. Itse tuote on alusta lähtien saanut ulkopuolisilta pelkkiä kehuja.

Olen tyytyväinen lopputulokseen. Toiveenani oli alun perinkin tehdä opinnäytetyö, josta voisi olla hyötyä ihmisille tulevaisuudessa, ja koen, että tässä on onnistuttu hyvin. Olen henkilökohtaisesti sitä mieltä, että monet opiskelijoiden tekemät opinnäytetyöraportit saattavat unohtua nopeasti, eikä niitä välttämättä kovinkaan laajasti lueta. Toivon, että tästä työstä on hyötyä ihmisille pitkään. Opin itse paljon projektin aikana eri asioista ja olen tyytyväinen aiheen valintaan sekä lopputulokseen.

LÄHTEET

Beaulieu, A. 2009. Learning SQL. Toinen painos. Sebastopol: O'Reilly Media.

Bootstrap 2019. Viitattu 27.1.2019, <https://getbootstrap.com>.

Duckett, J. 2014. JavaScript & jQuery: Interactive Front-End Web Development. Indianapolis: John Wiley & Sons.

Ek, Annakaisa 2009. Verisolujen tunnistusaapinen. Toinen painos. Kankaanpää: Messon

Ezell, L. 2016. Practical CodeIgniter 3. Viitattu 27.1.2019, <http://api.grave-design.com/practicalcodeigniter3.pdf>

HUS 2019. Hematologia. Viitattu 27.1.2019, <http://www.hus.fi/sairaanhoito/sairaanhoitopalvelut/hematologia>

Krug, S. 2014. Don't Make Me Think, Revisited: A Common-Sense Approach to Web Usability. Kolmas painos.

Marrs, T. 2017. JSON at Work. Sebastopol: O'Reilly Media.

Meiert, J. 2015. The Little Book of HTML/CSS Frameworks. Sebastopol: O'Reilly Media.

Meyer E. & Weyl, E. 2017. CSS: The Definitive Guide. Neljäs painos. Sebastopol: O'Reilly Media.

Nixon, R. 2014. Learning PHP, MySQL, JavaScript, CSS & HTML5. Kolmas painos. Sebastopol: O'Reilly Media.

Ryabtsev, A. 2016. Web Frameworks: How to Get Started. Viitattu 21.1.2019, <https://djangostars.com/blog/what-is-a-web-framework>.

Tatroe, K., MacIntyre, P. & Lerdord, R. 2013. Programming PHP. Kolmas painos. Sebastopol: O'Reilly Media.

Soluvisan käytettävyystestaus

Kiitos, että vastaat kyselyyn koskien Soluvisan käytettävyyttä. Vastaukset kerätään anonyymisti ja niitä käytetään sivuston parantamiseen. Ole siis hyvä, ja vastaa mahdollisimman rehellisesti jokaiseen kohtaan.

*Required

Yleistä sivustosta

Saitko ongelmitta käytettyä Soluvisa-sivustoa ja kaikkia ominaisuuksia *

- Kyllä
- Ei

Jos vastasit ei, mikä oli suurin ongelma?

Your answer

Verrattuna muihin samantyyliisiin sovelluksiin, soluvisaa oli *

	1	2	3	4	5	
Hyvin vaikea käyttää	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Helppo käyttää

Sain käytettyä kaikkia toimintoja, joita odotin sivustolta löytyvän

*

	1	2	3	4	5	
Täysin eri mieltä	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Täysin samaa mieltä

Löysin kaikki toiminnot helposti *

	1	2	3	4	5	
Täysin eri mieltä	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Täysin samaa mieltä

Sivuston ulkoasu ja rakenne oli selkeä *

	1	2	3	4	5	
Täysin eri mieltä	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Täysin samaa mieltä

Soluvisuaali oli mukava käyttää *

	1	2	3	4	5	
Täysin eri mieltä	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Täysin samaa mieltä

Onko sinulla ideoita tai parannusehdotuksia sivuston toiminnallisuutta ajatellen?

Your answer

Sisältö/solukuvat

Kuvien lukumäärä/testin pituus oli mielestäni *

- Sopiva
- Liian pitkä
- Liian lyhyt

Kuvat olivat selkeitä ja monipuolisia *

	1	2	3	4	5	
Täysin eri mieltä	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Täysin samaa mieltä

Kohtasitko virheitä kuvamateriaalissa/teksteissä? *

- En
- Kyllä

Oppimateriaalin tekstiosuus oli hyvä *

	1	2	3	4	5	
Täysin eri mieltä	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Täysin samaa mieltä

Onko sinulla kommentteja tai palautetta koskien sivuston tekstisisältöä tai kuvamateriaalia?

Your answer
