

Työ junior ohjelmistokehittäjänä Innofactorilla

Sami Salmimaa

Tekijä(t) Sami Salmimaa	
Koulutusohjelma Tietojenkäsittelyn koulutusohjelma	
Opinnäytetyön otsikko Työ junior ohjelmistokehittäjänä Innofactorilla	Sivu- ja liite-sivumäärä 36
Opinnäytetyön otsikko englanniksi Work as Junior Software Developer at Innofactor	
<p>Tämä opinnäytetyö keskittyy minun työtehtävääni junior ohjelmistokehittäjänä Innofactorilla ja kohdistuu ajalle 19.10.2020 – 10.01.2021. Päiväkirjamuotoinen opinnäytetyö koostuu päivittäisestä raportoinnista ja viikoittaisista analyyseistä, joissa tarkastelen omaa oppimista ja kehitystä.</p> <p>Innofactor on johtava modernin digitaalisen organisaation edistäjä Pohjoismaissa. Yrityksen tehtävänä on auttaa asiakkaita digitalisoimaan liiketoimintaansa, edistämään yhteisöllistä työntekoa ja kehittämään liiketoimintaprosesseja turvallisten pilvipalveluiden avulla.</p>	
Asiasanat Ohjelmistokehitys, kehitys, oppiminen, asiakastyö	

Sisällys

1	Johdanto	1
1.1	Tietoperusta.....	1
1.2	Keskeiset käsitteet	2
2	Lähtötilanteen kuvaus	3
2.1	Oman nykyisen työn analyysi.....	3
2.1.1	Työtehtävien kuvailu	3
2.1.2	Oman osaamisen arviointi.....	4
2.1.3	Oma ammatillinen kehittyminen	4
2.2	Sidosryhmät työpaikalla	5
2.3	Vuorovaikutustaidot työpaikalla.....	6
3	Päiväkirjaraportointi.....	6
3.1	Seurantaviikko 1	6
3.2	Seurantaviikko 2	10
3.3	Seurantaviikko 3	13
3.4	Seurantaviikko 4	17
3.5	Seurantaviikko 5	21
3.6	Seurantaviikko 6	24
3.7	Seurantaviikko 7	27
3.8	Seurantaviikko 8	30
3.9	Seurantaviikko 9	31
3.10	Seurantaviikko 10	33
4	Pohdinta ja päätelmät.....	34
	Lähteet	36

1 Johdanto

Tämä opinnäytetyö keskittyy työtehtävääni Innofactorilla ja kohdistuu ajalle 19.10.2020 – 10.01.2021. Päiväkirjamuotoinen opinnäytetyö koostuu päivittäisestä raportoinnista ja viikoittaisista analyyseistä, joissa tarkastelen omaa oppimista ja kehitystä.

Omassa työtehtävässäni tarvitaan olio-ohjelmoinnin tuntemusta ja C# kielen perusteita. Projektien välillä saattaa kielet ja teknologiat vaihtua, mutta pääsääntöisesti kielenä on ollut C# ja HTML5 sekä Episerver. Jokaisessa projektissa käytetään versionhallintaa ja Git-versiohallinnan tuntemus on tullut tarpeen. Käytämme projekteissa Visual Studiota ja Visual Studio Codea. Tiimissä käytämme ketterää ohjelmistokehitys mallia Scrumia. Työnkuvaani kuuluu kehitysympäristöjen asennus virtuaalikoneille ja niiden ylläpito.

Opinnäytetyö keskittyy minun työnkuvaani junior ohjelmistokehittäjänä Innofactorilla. Innofactor on johtava modernin digitaalisen organisaation edistäjä Pohjoismaissa. Yrityksen tehtävänä on auttaa asiakkaita digitalisoimaan liiketoimintaansa, edistämään yhteisöllistä työntekoa ja kehittämään liiketoimintaprosesseja turvallisten pilvipalveluiden avulla. Innofactorilla työskentelee 500 asiantuntijaa Suomessa, Ruotsissa, Tanskassa ja Norjassa. Itse työskentelen Espoon toimipisteessä, jossa on noin 250 työntekijää. Innofactor on jaettu eri yksiköihin ja näistä itse kuulun Digital Experience -yksikköön, jossa toimin junior ohjelmistokehittäjänä. Digital Experience -yksikön sisällä on tiimejä, jotka toimivat erilaisissa asiakkuuksissa omien taitojen ja mielenkiinnon mukaan. Itse kuulun kahdeksan hengen tiimiin ja meillä on seitsemän eri asiakasta, joille ylläpidämme ja jatkokehitämme digitaalisia palveluita.

1.1 Tietoperusta

Työssäni käytän tietoperustana Pluralsight verkkokurssi -sivustoa. Työnantaja on maksanut minulle sinne vuoden lisenssin, jolla pääsen kaikkiin sivuston kursseille. Tällä hetkellä minulla on kesken ASP.NET MVC ja C# -kurssit. Muita en ole vielä ehtinyt aloittamaan. Olen myös ostanut Packt> sarjan videokursseja ja kirjoja liittyen C# ja ASP.NET Core -ohjelmointikieliin. Tietysti myös työkavereiden kanssa tehdään paljon yhteistyötä ja heiltä tulen varmasti kysymään apua.

Kaikki projektit ovat MVC-mallin projekteja, joihin tarvitsen projektinhallinnan Front-End ja Back-End tuntemusta. Back-End puolella käsitellään tietoa ja tieto muutetaan haluamaan muotoon ja tallennetaan tai haetaan tietokannasta. Front-End puolella taas voidaan muokata ja esittää tietoa eri tavoin riippuen siitä, miten asiakas sen haluaa. Yleisimmät

1.2 Keskeiset käsitteet

Bugi: Ohjelmaan päässyt ominaisuus/virhe mikä ei ole haluttu.

HTML: Kieli, jolla tehdään elementtejä, jota selaimet osaavat tulkita ja näyttää.

CSS: Cascade Style Sheet, Kielellä tehdään erilaisia tyylejä jamaokkauksia HTML elementteihin.

IIS: Internet Information Services. Microsoftin kehittämä palvelinohjelmistokokonaisuus.

CI: Jatkuva integraatio (continuous integration) on menetelmä, jolla voidaan jatkuvasti liittää uutta koodia projektiin.

Git: Hajautettu versionhallintajärjestelmä, hallitaan projektin versioita ja pitää projektin järjestyksessä.

MVC: Model, View, Control. Kuvaa projektin mallin rakennetta.

Episerver: Sisällönhallinta työkalu.

.Net: Framework, ohjelmistokomponenttikirjasto.

ASP.NET: Lisätyökalu .Net alustaan. Lisää työkaluja ja kirjastoja.

Backlog: Kanban taulun yksi sarake minne asiakkaan kanssa luodaan erilaisia muutoksia ja päivitys tehtäviä.

Front-End: Käyttöliittymä, osa ohjelmistoa, jota käyttäjät näkevät ja käyttävät.

Back-End: Ohjelmiston osa, joka on piilossa ja hoitaa tiedon siirron ja muokkauksen.

Script: Komentokieli, korkean tason ohjelmointikieli.

E2E: End to End.

BitBucketia ja SourceTree: Versiohallinta sovelluksia, jotka käyttävät Gittiä

2 Lähtötilanteen kuvaus

2.1 Oman nykyisen työn analyysi

2.1.1 Työtehtävien kuvailu

Työtehtävät:

- Verkkosivujen ylläpito
- Verkkosivujen kehitys
- Tiedonhallinnan asiantuntija tehtävät

Työtehtäviini kuuluu asiakkaiden verkkosivujen ylläpito ja kehitys. Keskiössä ovat asiakasprojektit, jotka määrittelevät mitä teknologioita ja järjestelmiä käytetään. Pääsääntöisesti projekteissa pystytetään kehitysympäristö koneelle, usein kyseessä on virtuaalikone, jonne haetaan kehitettävä projekti versiohallintatyökalulla. Omassa tiimissäni asiakkaat käyttävät C#, ASP .NET ja episerveriä. Käytämme Visual Studio -sovellusta editorina ja SourceTree-sovellusta versionhallinnassa.

Työtehtävät vaihtelevat pienistä tyyli muutoksista isompiin ominaisuusmuutoksiin tai uusiin kokonaisuuksiin. Minä olen uusi työntekijä, joten en vielä ole päässyt mukaan asiakasta-paamiin, joissa keskustellaan muutoksista, aikatauluista ja luodaan backlogia.

Minulle tulleissa työtehtävissä tarvitaan käyttöliittymän hallintaa sekä hyvää HTML5:n ja CSS3:n tietämystä. Suurin osa tehtävistä keskittyy ulkoasuun ja näitä päivitetään paljon, jotta sivut näyttäisivät moderneilta. EU:n on laatinut säädöksiä, joka tuo paljon muutoksia saavutettavuuteen. Saavutettavuudella tarkoitan verkkosivuston käyttömahdollisuutta kävijästä riippumatta, jokaisella pitää olla tasavertainen mahdollisuus käyttää sivustoa. Laki velvoittaa noudattamaan saavutettavuusvaatimuksia. Muutokset eivät pelkästään ole visuaalisia, vaan niiden taakse kätkeytyy myös logiikkaa, esim. käytettävyys on iso osa mitä mietin, kun teen uutta tai muokkaan vanhaa ominaisuutta. Mitä tapahtuu napin painamisen jälkeen, minne käyttäjä ohjataan tästä tilanteesta ja miltä se näyttää yms.

Kyseisissä työtehtävissä tarvitaan hyvää ohjelmointiosaamista ja tiimityötä. Pitää pystyä hakemaan työkaverilta apua, jos jää jumiin ja pitää myös osata auttaa kaveria, jos hän pyytää apua. Tiiminä me hoidamme asiakkaan tarpeet ja muutokset heidän palveluissaan. Olemme yhtiö, joten tarkoitus on laskuttaa tehdystä työstä, sen vuoksi annamme backloggissa oleville tehtäville aika-arvion, jonka mukaan asiakas voi ne budjetoida. Sitä mukaan kun asiakas ne hyväksyvät, me alamme töihin.

2.1.2 Oman osaamisen arviointi

Tässä työssä vaaditaan paljon käyttöliittymän tuntemusta eikä se ole minun vahvin osaamisalueeni, joten arvioin itseni näihin työtehtäviin aloitteleva toimija. Vahvimmillani olen Back-End puolella.

Tarvitsen vielä paljon apua työkavereilta työtehtäviin liittyen ja selontekoa tehtävien kuvauksissa. En tarvitse niinkään paljon apua ohjelmoinnin logiikassa tai toiminnallisuudessa, ja pidänkin tietojenkäsittelystä koodissa ja sen viemistä eteenpäin. Paljon on myös ollut työtä saavutettavuuksien kanssa, mikä on ollut minulle aivan uusi asia ja tässäkin tarvitsen paljon apua.

Episerverin opin nopeasti ja sen jälkeen, kun ensimmäisen sivun tai blokin on luonut, on siitä helppo jatkaa. Tässä luokittelisin itseni taitavaksi suoriutujaksi, koska voin itsenäisesti luoda sivupohjia ja blokkeja sekä hyötyä käyttää niitä.

C#-kieli oli helppo opetella, koska minulla on vahva Back-End Java -tausta ja C# muistuttaa paljon Javaa. Syntaksi ja valmiiden metodien käyttö vahvistuu koko ajan, joten ohjelmointilogiikka ei pääse unohtumaan.

2.1.3 Oma ammatillinen kehittyminen

Oma osaamisen taso on tällä hetkellä alhainen. Koulusta oppimani on enimmäkseen Back-End puolelta ja tässä tiimissä olen saanut enimmäkseen käyttöliittymän puolen tehtäviä. Olen kuitenkin nopea oppimaan, enkä karta mitään tehtäviä. Perus olio-ohjelmointi on minulle tuttua ja sitä näissäkin projekteissa käytetään. MVC-malli on myös tuttu, mutta siinä haluan kehittyä paljon.

Ketterä ohjelmistokehitys ei ole niin tuttua, koska koulussa meillä oli todella isot ryhmät ja niistä oli scrum kaukana. Ketterä ohjelmisto kehitys on haasteellista aloittelijalle ja varsinkin nyt, kun korona on iskenyt ja joudun oppimaan kaiken etänä. Onneksi minulla on kuitenkin hyvä tiimi ja jokainen on valmis auttamaan, kun apua tarvitsen.

Uusia asioita ovat ASP .NET ja C# ja muut komentokielet. Vahvin kieleni on Java ja eniten opittavaa ASP .Net alustasta. Projektin rakenne on hankala ja suuri, kun asiakkaiden verkkosivut ovat vaativia ja niissä on monia ominaisuuksia. Minulla menee paljon aikaa, jotta pääsen jokaiseen projektiin sisään, jotta työni helpottuu. On paljon helpompaa aloittaa työt, kun tietää suurin piirtein, missä mikäkin on ja mitä lähtee korjaamaan tai uudistamaan.

Jatkossa aion keskittyä syventämään osaamista C#-kieleen ja MVC projektien tiedostorakenteisiin. HTML5 ja CSS3 kielet ovat myös suuri osa työpäivääni ja niihin haluan tutustua enemmän. Scrum koulutusta meille tarjottiin ensimmäisinä työpäivinä ja haluan pitää kiinni scrum ohjeista ja tehdä työtä ketterästi, omatoimisesti ja itsenäisesti.

2.2 Sidosryhmät työpaikalla

Sisäisiä sidosryhmiä

- Esimies: Henkilö ML
- Tiimi A:
 - Henkilö J
 - Henkilö K
 - Henkilö A
 - Henkilö T
 - Henkilö H
 - Henkilö SB
- Yksikkö:
 - Digital Experience DE

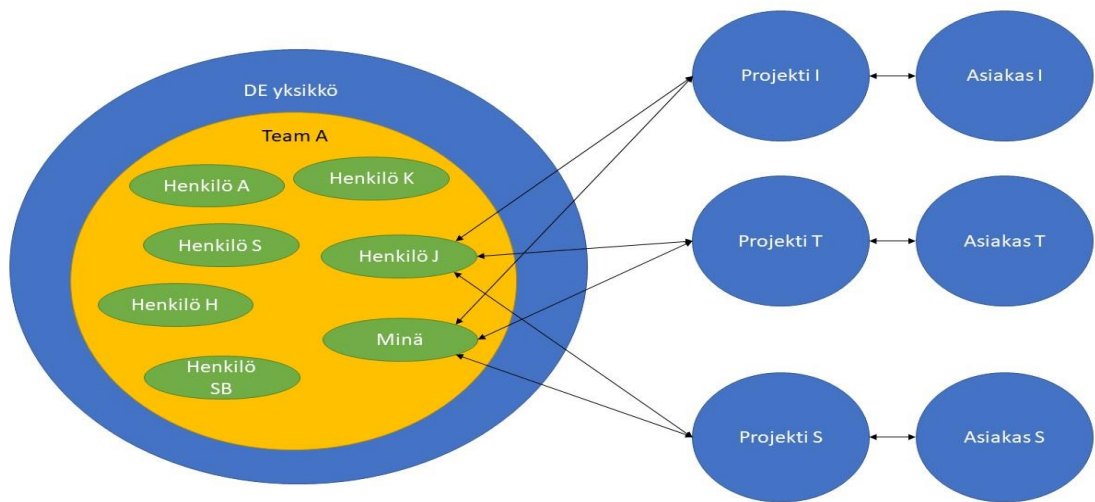
Ulkoisia sidosryhmiä

- Asiakkaat:
 - Asiakas S
 - Asiakas T
 - Asiakas I

Katso kuva 1.

Työssäni pääsääntöiset sidosryhmät ovat oma tiimini ja asiakkaat. Sisäiseen sidosryhmään kuuluvat tiimini kuusi jäsentä, esimieheni ja kollegat omassa yksikössäni (DE). Oman tiimini kanssa jaetaan työtehtäviä ja keskustellaan miten tehtävät tehdään.

Ulkoisia sidosryhmiä ovat asiakkaat. Asiakkaiden kanssa luodaan kehityskohteita ja backlogia. Asiakaspalaveriini en vielä pääse, mutta tiimin kanssa keskustelemme asiakkaiden tarpeista päivittäin.



Kuva 1 Sidosryhmäkaavio

2.3 Vuorovaikutustaidot työpaikalla

Vuorovaikutustilanteita tulee tiimin kanssa joka aamuissa palaverissa, jossa keskustelme päivän töistä ja eri projektien tilanteesta.

Koronan ja etätyön takia minulle ei tule kontakteja yhtiön muiden työntekijöiden kanssa. Se on minusta huono asia, sillä haluaisin tutustua laajemmin yhtiön muihin työntekijöihin.

Omaan yksikköön vuorovaikutus syntyy yksikön ja johdon järjestämällä tiedotustilaisuuksissa ja muissa tapahtumissa, joita yritetään järjestää mahdollisimman paljon, jotta etätyö ei tuntuisi niin pahalta ja pysyy joku kontakti muihin työkavereihin.

Asiakkaiden kanssa minulla ei ole mitään vuorovaikutusta, kun en vielä ole päässyt mukaan asiakastapaamisiin tai backlogin luontiin. Mutta se on tavoitteeni.

3 Päiväkirjaraportointi

3.1 Seurantaviikko 1

Maanantai 26.10.2020

Tänään palautan asiakas S:n tyyli muutokset koskien evästekyselyä. Siinä on pieni varjostus, jotta se ei häviä taustan sekaan sekä pieni teksti ja hyväksy nappi. Palautan muutokset työkaverille, joka käy ne läpi. Jos on jotain muutettavaa, käydään ne yhdessä läpi ja jatkan muutoksien tekemistä. Minulla on myös työn alla TestCafe E2E -testauksen opiskelua ja testien tekoa.

Tyylimuutokset olivat työkaverin mielestä hyviä ja kyseiset muutokset vietiin testiin. Tämän jälkeen jatkoin TestCafe-verkkokurssia plurarsightissa, jossa käsiteltiin ympäristöön tarvittavia asennuksia ja asetukset-tiedoston sisältöä.

Kurssin loputtua asensin projektiin TestCafe-ohjelman, joka pyörittää testejä ja aloitin asetukset-tiedoston tekemisen. Asetukset-tiedostoon laitetaan tietoa, jotka toimivat testien "asetuksina". Näillä asetuksilla määritetään miten alusta toimii.

Tiistai 27.10.2020

Tänään pystytän uuden asiakkaan kehitysympäristöä (Asiakas T) ja jatkan TestCafe-testien tekemistä. Pystytyksen kanssa tuli ongelmia ja olen yrittänyt korjata asiaa, mutta saan työkaverin minua auttamaan.

Ongelma tuli, kun yritin päästä asiakkaan versionhallintaan, josta voin kopioida projektin omalle koneelle. Asiakas haluaa käyttää BitBucketia ja minulle ei ollut lisätty käyttäjää sinne, joten en luonnollisesti päässyt sisään. Muuten pystytys sujui hyvin, sain tietokannat ja Visual Studion toimimaan.

TestCafessa teen ensimmäistä testiä samalla, kun katson Plurarsightista verkkokurssia aiheesta. Testit ovat aika monimutkaisia, sillä en ole paljoa käyttänyt komentokieltä. Ensimmäisessä testissä pyrin saamaan "sivua ei löydy"-sivun ja testin toteamaan, että kyseinen sivu tulee esiin. Projektissa on siis itse tehty muokattavissa oleva "sivua ei löydy"-sivu ja tällä testillä pyrin testaamaan, että sivu tulee näkyviin.

Keskiviikko 28.10.2020

Sain tänään tehtäväksi tehdä asiakkaan sivulle React & Share elementin. React & Share on elementti, jota käytetään nopeaan ja helppoon palautteen keräämiseen.

Tehtävä meni oikein hyvin. Kyseisessä tehtävässä asiakas halusi sisällöntuotantoon valintaruudun, josta voi päättää onko ominaisuus käytössä vai ei. Lisäksi vielä kentän mihin syötetään avain millä haetaan oikea sisältö React & share palvelusta.

Tämän jälkeen tein sivuille HTML elementin, jonne lisäsin R&S:ltä saamani komentokieli komennon, joka hakee sisällön palvelusta. Tämän elementti pitää laittaa monelle sivulle, joten loin oman tiedoston, johon voidaan sivulla viittaa, jos kyseisellä sivulla on tämä

ominaisuus käytössä. Tällöin minun ei tarvitse laittaa kopioida koko R&S elementtiä jokaiselle sivulle.

Lisäsin pääsivun alalaitaan kyseisen viittauksen kyseiseen ominaisuuteen, joka vielä tarkistaa sisällöntuotannosta valintalaatikon, jos kyseinen valintalaatikko on päällä elementti näytetään käyttäjälle.

R&S-tehtävä oli aika yksinkertainen, mutta projektin laajuuden takia siihen meni aikaa, jotta ymmärsin missä R&S elementti rakentuu ja mihin kohtaa se saadaan. Ja tietenkin piti luoda sisällöntuotannon sivuille kenttä mihin avain tulee, jota käytetään tietojen haussa ja valintaruutu, josta tarkastetaan, halutaanko toiminto näyttää käyttäjälle. Lopuksi tein elementin ympärille IF-lauseen, joka tarkistaa näytetäänkö elementti vai ei.

Torstai 29.10.2020

Tänään jatkan TestCafe-testin tekoa ja yritän saada sen valmiiksi.

Ensimmäisen testin teko olikin isompi tehtävä, kuin miltä se ensin näytti. Jo valmiiksi isoon projektiin tällaisen teko on aika suuri tehtävä. Jokaiseen testattavaan nappiin, input- ja tekstikenttään pitää luoda oma tunniste, johon testissä sitten viitataan.

"sivua ei löydy"-testiin lisäsin yhteen etusivun nappiin linkin, joka vie tahallaan väärälle sivulle, jota ei ole olemassa, jotta saadaan "sivua ei löydy"-sivu näkyviin. Tähän yritän testillä päästä käsiksi, että testi painaisi tätä nappia ja sivun päivitettyä se varmistaisi, että ollaan "sivua ei löydy"-sivulla. Tein tehtävää rinnakkain Plurarsightin verkkokurssin kanssa ja eteneminen oli todella hidasta huonon komentokielen tuntemuksen takia. Aloitin silti tekemään elementeille TestCafen omia tunnisteita, joita voin käyttää testeissä.

En saanut valmiiksi ensimmäistä testiä enkä päässyt kokeilemaan. Pitää vielä miettiä miten varmennan, että oikea "sivua ei löydy"-sivu on näkyvässä ja miten testissä totean testin onnistuneeksi tai epäonnistuneeksi.

Perjantai 30.10.2020

Tänään aloitan asiakas S:n käytettävyys muutoksien tekemisen. Aloitan kysymysmerkkikuvakkeiden korjauksista ja niiden saatavuudesta. Kysymysmerkkikuvakkeet auttavat käyttäjää käyttämään lomakkeita oikein. Kun niiden päälle laittaa hiiren aukeaa info teksti

Tällä hetkellä kyseisiin kysymysmerkkikuvakkeisiin tai niiden sisältöön ei pääse ollenkaan näppäimistöllä. Tehtäväksi sain muokkaa kysymysmerkkikuvakkeet niin että ovat saavutettavissa niin hiirellä kuin näppäimistöllä.

Alkuun tuli ongelma, missä nämä kysymysmerkit rakennetaan, koska projektissa on monta kerrosta ja pääkoodi on pelkistetty hyvin, jotta se olisi helppolukuista. Hyvästä tiedostorakenteesta huolimatta oli kyseinen elementti vaikea löytää.

Kyseisille kysymysmerkkikuvakkeille oli tehty oma malli, jota käytetään rakennettaessa. Episerverin puolelta haetaan sitten teksti mitä kuvake näyttää. Löydettyäni rivin, missä kyseinen kuvakkeen kuva asetetaan ja mistä se hakee sisällöntuotannosta tekstin, lisäsin kyseiselle riville attribuutin, joka antaa tabulaattorille luvan käyttää. Kyseisellä attribuutilla päästään näppäimistön TAB-napilla kiinni kysymysmerkkiin eikä hiirtä tarvita.

Kysymysmerkin sisältö ponnahtaa auki fokusoituessaan hyvin, kun siihen päästään TAB-napilla ja siihen ei tarvita muutoksia.

Viikkoanalyysi

Viikko oli kokonaisuudessaan oikein opettava. Alkuviikon tyyli muutoksissa oli CSS- (Christopher Schmitt 2009) ja HTML-kieltä (HTML Tutorial). Näistä kielistä minulla on koulusta perusteet, mutta näissä oli kumminkin opettelemista tiedosto rakenteen, luokkien ja vanhempi ja lapsi -kerroksista. Ihan uutena oli Less-tiedostojen rakenne, käyttö ja kääntäjään tutustuminen. Projektin koko rakenne ja tiedostojen sijainti tuli hyvin tutuksi oikeita luokkia etsiessä. Tehtävässä olleet tyyli muutokset eivät olleet haastavia, kun löysin vain oikeat tiedostot ja luokat, niin pääsin aloittamaan tyylien muokkaamisen.

Tällä viikolla pystyitin myös uuden asiakkaan kehitysympäristön ja pääsin käyttämään Windowsin IIS ja SQL serveriä. SSMS-ohjelmaa (SQL Server Management Studio) käytin myös, jonne lisäsin tietokannan palautustiedostosta ja oman käyttäjän sisäänkirjautumistunnuksen sekä oikeudet käyttää kyseistä tietokantaa. BitBucketin versiohallinta oli uusi juttu, ja se tuntuu vähän kankeammalta kuin SourceTree, joten siihen minulla menee varmasti hetki aikaa tottua. Kyseisessä ohjelmassa on aika huono graafinen käyttöliittymä, kun on tottunut käyttämään visuaalisesti vähän parempaa. Projekti ei kuitenkaan käynnistynyt, vaan minun piti vielä käydä episerveriin lisäämässä URL, joka luo polun minun projektilleni.

Asiakkaan T R&S-tehtävässä tuli taas sama ongelma kuin asiakas S:n tyyli muutoksissa. Projekti on iso ja tiedostoja paljon, enkä oikein tiennyt mistä löytäisin rivit, mitä pitää muokata. Pienen salapoliisityön tuloksena pääsin loppujen lopuksi jäljille, missä nämä sijaitsevat ja sain hyvän käsityksen siitä, miten episerver ja ASP .NET pelaavat yhteen. Kuten MVC-mallin projekteissa, tässäkin on luotu mallit, jota HTML-elementit käyttävät, jolloin voit sisällön tuotannossa luoda ja lisätä mitä haluat, ja nämä mallit näyttävät asiat aina samalla tavalla mutta eri sisällöllä.

Asiakas S:n toinen tehtävä on yksi ison kokonaisuuden tehtävistä. Asiakas haluaa käytettävyyteen ja saatavuuteen parannuksia ja minulle annettiin tämä pieni tehtävä monen joukosta. Yksinkertainen tehtävä sai kuitenkin käänteeseen, johon olin törmännyt alkuviikosta, en tiennyt missä nämä kysymysmerkit rakennetaan ja miten pääsen niitä muokkaamaan. Toivoin ehdin etsiä, kunnes löysin paikan ja rivin muokkaaminen alkoi.

Viikosta jäi hyvin käteen ASP .NET ja episerverin projektin koko rakenne ja opin todella paljon siitä missä mikäkin on. Nyt jos etsin jotain tiettyä paikkaa tai riviä, tiedän mistä aloitan etsimään. Projektin tiedostopuu on hyvin jäsennelty. MVC-mallin projekteihin tämä viikko on tuonut todella paljon infoa. Nyt ymmärrän paljon paremmin mallien ja käyttöliittymän tiedostojen tarkoituksen, ja tästä eteenpäin on helpompaa kehittää verkkosivuja (Scott Allen 2019).

Tällä viikolla pääsin asiakastyöhön kiinni ja laskutin ensimmäiset tunnit oikealta asiakkaalta. Aikaisemmin olen tehnyt töitä sisäisesti Innofactorin omia verkkosivuja kehitellen. Tässä vaiheessa tuntuu, että olen päässyt eteenpäin kehittäjänä ja työntekijänä, kun pääsen tekemään asiakkaan toivomia muutoksia ja jättämään oman kädenjäljen koodiin.

Tällä viikolla olen saanut tehtävistä hyvää palautetta työkavereiltani ja meillä otettiin kokeiluun positiivinen perjantai, jossa annetaan yksi positiivinen palaute työkaverista kuluneelta viikolta. Olen saanut hyvää palautetta ja sulautunut tiimiin hyvin.

3.2 Seurantaviikko 2

Maanantai 02.11.2020

Sairaana

Tiistai 03.11.2020

Sairaana

Keskiviikko 03.11.2020

Päivä alkaa kesken jääneellä tehtävällä asiakas S:lle. Tehtävässä parannetaan käytettävyyttä kysymysmerkkikuvakkeiden pääsyyn näppäimistöllä, ja niiden fokuoitessaan näyttäisi tiedon samalla lailla, kuin jos menisi hiirellä kuvakkeen päälle. Tämän jälkeen jatkan asiakas T:N R&S-tehtävää, kun asiakas haluaa siihen vielä lisää muutoksia.

Asiakas halusi, että React & Shrare siirrettäisiin sivuston alalaidasta lähemmäksi sivuston sisältöä. Asiakas antoi muutaman sivun missä tämä pitää olla.

Sain kysymysmerkkitehtävän valmiiksi ja palautin sen työkaverille, joka hyväksyi sen ja lisäsimme sen seuraavaan testiajoon. Tehtävässä ei ollut paljoa enää jäljellä, tarkistin vain, että se toimii ja fokuoitessaan aukeaa oikein.

Asiakas T:n R&S-tehtävään tuli lisäyksiä, että asiakas ei haluakaan kyseistä kyselyä sivuston alalaitaan, vaan se olisi suoraan kyseisen sivun sisällön alla. Sain listan tärkeimmistä sivuista, mistä ominaisuus pitää löytyä ja siirsin div-elementin IF-lausekkeineen jokaisen sivun sisällön alle. Kokeilimme työkaverin kanssa, että se toimii oikein ja ilmoitimme asiakkaalle, että on siirto tehty.

Torstai 05.11.2020

Tänään jatkan asiakas S:n käytettävyyss tehtävillä. Valitsin tehtävän, jossa parannetaan lomakkeiden käytettävyyttä. Lomakkeen kentän pitäisi ohjeistaa mitä kenttiin tulee ja missä muodossa ja että tieto olisi mahdollista myös syöttää itse. Tällä hetkellä päivämäärää syötettäessä käytetään kalenteri näkymää josta käyttäjä valitsee päivän.

Kyseessä oli siis vain lomakkeen toisessa vaiheessa oleva päivämäärä kenttä, jossa oli käytetty Vuen kalenteri näkymää. Kyseinen ominaisuus varmistaa päivämäärän oikean muodon vasta kun yritetään mennä seuraavaan vaiheeseen. Muokkasin kenttää siten, että se tarkistaa kyseisen arvon heti kun käyttäjä siirtyy pois kentästä. Lisäsin myös aria-labelledby, -describedby attribuutit. Lisäsin pattern attribuutin, jotta käyttäjää ohjattaisiin käyttämään oikeaa muotoa, ja jos muoto ei ole oikea, kenttä ilmoittaa oikean muodon kyseisen kentän alapuolella punaisella.

Tänään on myös Jira Service Desk -koulutus ja asiakkaan jira koulutus, jossa käydään yhteinen toimintamalli läpi.

Perjantai 06.11.2020

Työkaverin kanssa käydään asiakas S:n käytettävyystehtävä kalenteri näkymän osalta läpi ja sovitaan muutoksista. Sen jälkeen alan tekemään saman asiakkaan muiden lomakkeiden käytettävyyssuutoksia. Muutamia muutoksia tuli kalenteri näkymään ja tarkoitus oli vielä laittaa siihen jonkin sortin ajastin, jotta heti ensimmäisen numeron kirjoitettua se ei ilmoita virheestä, vaan saa rauhassa laittaa siihen koko päivämäärän ennen kuin se tarkistetaan ja ilmoitetaan mahdollisista virheistä.

Tämän jälkeen alan tutkimaan ARIA-rooli/live -muutoksia, jotka pitää saada tehtyä asiakas S:n lomakkeisiin. Totesin ajastinta tutkiessani, että ei tarvitsekaan sitä laittaa. Koodi alkoi toimimaan siten, että kun käyttäjä poistuu kentästä kyseinen kenttä tarkastetaan.

Tehtävässä keskityin lomakkeiden kenttien virheilmoituksiin.

Kenttien virheilmoitustekstit ovat merkitty elementeihin niin, että kenttään kohdistuu kaksi elementtiä tai ne on jätetty kokonaan kytkemättä kenttään.

Tehtävässä käytetään ARIA-live tai rooli ja ARIA-labelledby-attribuutteja virhetilanteiden kommunikointiin. Lisäksi virheellisiin kenttiin ARIA-invalid-attribuutti todeksi ja tarvittaessa tarkastetaan myös pakollisten kenttien kommunikaatiota lomakkeissa.

Kyseisessä tehtävässä pääsin tutkimaan ARIA:n käyttöä. Tehtävässä oli tarkoitus antaa ARIA merkit kentille ja myös kääntää virheviestit käyttämään ARIA-roolia, jotta käytettävyys ja saatavuus paranisi. Myös koko kenttä oli laitettu merkityn elementin sisään, joka piti siirtää sieltä pois.

Ensin laitoin merkitylle kentälle tunnusteen, johon labelledby-attribuutti viittaa, kun olen siirtänyt sen pois merkityn kentän sisältä. Sitten lisäsin describedby-attribuutin, joka viittaa virheviestiin, kun semmoinen tulee se näytetään kyseisen kentän alla.

Kyseisessä projektissa on paljon räätälöityjä lomakkeita, ja jokaisessa lomakkeessa on paljon kenttiä, mitkä pitää käydä läpi, joten tehtävä jäi kesken ja jatkan sitä maanantaina.

Viikkoanalyysi

Pari päivää olin kipeänä enkä voinut tehdä töitä. Keskiviikkona voin jo paremmin ja jatkoin työtekoa.

Asiakkaan kanssa oli koulutus, missä sovittiin Jira kanban boardin yhteisistä pelisäännöistä eli miten sitä käytetään, jotta kaikki toimisivat samalla tavalla. Omassa sisäisessä koulutuksessa käytiin läpi uuden toimintamallin Service Deskin käyttöä, mikä oli mielenkiintoista. Jatkossa asiakas tekee tikettejä ja Service Desk laittaa niitä meille eteenpäin arvioitavaksi ja tehtäväksi.

Loppuviikosta aloin tutkimaan ARIA:n käytettävyyttä ja tämä on minulle uusi juttu. ARIA (Accessible Rich Internet Applications) on teknologia, jolla lisätään sivun käytettävyyttä ja saatavuutta. Tähän asiaan oli kiva tutustua ja oppia lisää. Ensin oli haasteita ymmärtää mitä eroa on labelledby:lla ja describedby:lla, mutta sain työkaverin selventämään asian minulle, ja nyt ymmärrän asiaa vähän paremmin. (Anniina Saapunki 2018)

Internet on tehty kaikkien ihmisten käytettäväksi, riippumatta ihmisen käyttämästä laitteesta, kielestä tai kyvyistä. Kun tämä tavoite saavutetaan, erilaiset verkkopalvelut ovat käytettävissä myös näkö- tai kuulovammaisille, liikuntarajoitteisille. Verkon palvelut ja sisällöt voivat jopa poistaa sosiaalisia esteitä ja kommunikaatiovaikeuksia, joita ihmiset kohtaavat elämässään. (Anniina Saapunki 2018)

Tehtävässä keskityttiin lomakkeisiin ja niiden rakenteisiin. Lomakkeiden rakenteessa tulee ottaa huomioon muutamia asioita, kuten kenttien nimikkeet ja selitteet, sekä viestit saavutettavuuden parantamiseksi. Käyttäjälle tulee tarjota ohjeita kenttien täyttämiseen. Käyttäjälle näytettävät tarkastukset tai virheviestit voi myös toteuttaa saavutettavasti käyttämällä attribuuttia ARIA-describedby. (Anniina Saapunki 2018)

ARIA:n kolme pääominaisuutta ovat roolit, ominaisuudet sekä tilat. Roolit kertovat, mikä elementti on. Ominaisuudet kuvaavat elementtien ominaisuuksia, jotka voivat antaa niille lisämerkitystä. Esimerkiksi ARIA-required kertoo kentän täyttämisen pakollisuudesta. Aria-attribuuttia aria-labelledby käytetään antamaan tunniste elementille ja referoimaan sitä missä vain muualla sivustoa. (Anniina Saapunki 2018)

3.3 Seurantaviikko 3

Maanantai 09.11.2020

Jatkan asiakas S:n lomakkeiden kenttien saavutettavuus muutosta, mikä jäi kesken perjantaina.

Työtä oli todella paljon. Lomakkeita oli enemmän kuin mitä luulin ja jokaisessa lomakkeessa oli todella monta kenttää, joka piti muuttaa. Sama työ jatkuu tiistaina.

Tiistai 10.11.2020

kenttien muutoksien tehtävä jatkuu ja vaihtelun vuoksi aloin etsimään tehtävään liittyen missä luodaan virheviesti elementti. Tehtävässä oli myös, että kentille annetaan ARIA-invalid attribuutti ja tämän attribuutin muuttuessa, niin tulee virheviesti näkyviin. Tähän liittyen myös virheviesti elementtiin pitää lisätä rooli ja describedby attribuutti ohjaa sen oikeaan paikkaan.

virheviesti elementtiin lisätään ARIA rooli error. Kysyin työkaverilta apua, kun en löytänyt missä tämä virheviesti elementti luodaan. Työkaverin ohjeistuksella löysin jQuerysta kohdan, missä tämä virheviesti elementti tehdään ja lisäsin sinne ARIA roolin "error". Nyt jokaiseen virheviestiin tulee automaattisesti rooli.

Sitten jatkoin kenttien muokkaamista loppupäivän.

Keskiviikko 11.11.2020

Tänään jatkan eiliseltä kesken jäänyttä lomakkeiden muutostehtävää asiakas S:lle. Minun otti yhteyttä markkinoinnin puolelta työkaveri ja kysyi jos pystyisin muokkaamaan hubSpotissa sisältöä. Olen hänen pyynnöstään opetellut hubSpottia ja tehnyt hänelle pieniä muutoksia sivustoille.

Kyseessä oli pieni muutos sivun alalaidan sisällössä. Alalaidassa oleva sisältö on toteutettu kyseisellä sivulla templatella ja minun olisi pitänyt muuttaa templatea, jotta sivustoihin päivittyy oikea tieto. Hetken tutkittuani asiaa huomasin, että minun käyttäjälläni ei ollut lupaa muuttaa templateja, joten tämä työ tyssäsi alkuunsa. Pyysin työkaveria nostamaan käyttäjäni, jotta voisin muokkauksen tehdä, mutta hänelläkään ei ollut valtuuksia semmoista tehdä. Työ annettiin eteenpäin käyttäjille, jolla on valtuudet.

Jatkoin kenttien muokkaamista ja tänään sain kaikki tehtyä, joten kommentoin muokkaukset ja puskin muutokset versiohallintaan.

Torstai 12.11.2020

Tänään jatkan asiakas S:n saavutettavuus muutoksia. valintalaatikoissa ja radionapeissa käytetään kustomoitua toteutusta, jossa 1) fokus ei erotu kunnolla 2) on kontrastihaasteita. Kontrastihaasteita on myös yleisesti syötekenttien erottuvuudessa.

Radionapit ja valintalaatikot muutetaan saavutettaviksi ja fokusointi näkyväksi. On myös käytävä läpi kaikki räätälöidyt lomakkeet, joissa toteutus saattaa olla hieman erilainen.

Seuraavassa tehtävässä ollaan samojen asioiden äärellä. Nyt vain ei muuteta tekstikenttiä, vaan myös radionappeja ja valintalaatikoita. Samoissa räätälöidyissä lomakkeissa mistä muutin kenttiä, löytyy myös radionappeja ja valintalaatikoita. Näihin ei tällä hetkellä pääse TAB-näppäimellä valitsemaan eikä ne fokusoidu ollenkaan.

Ongelma oli siinä, että radionappeja ei ollut ryhmitetty ollenkaan. Jotta näihin pääsisi TAB-näppäimellä ja valitsemaan nuolinäppäimillä oikean kohdan, niille jokaiselle pitää antaa attribuutti, jotta tietävät kuuluvansa samaan ryhmään. Lisäsin myös attribuutin, jotta TAB-napilla pääsee kyseiseen radionappi ryhmään kiinni. Valintalaatikoissa ei ollut niin paljoa työtä, lisäsin vain niihin myös TAB attribuutin, jotta niitä pääsee myös valitsemaan näppäimistöllä.

Fokusointi ei myöskään näkynyt radionapeissa tai valintalaatikoissa. Tähän löysin ratkaisun CSS:stä. Etsin käsiini oikean Less-tiedoston, missä käsiteltiin radionappien ja valintalaatikoiden tyylejä ja muotoja. Siellä oli kohta, jossa käsiteltiin näiden luokkien fokusointia. Näihin lisäsin komennon, missä käytetään rajausta. Rajaus on siis elementin ympärillä oleva näkyvä viiva. Komento ottaa rajauksen käyttöön, kun käyttäjä aktivoi radionapit tai valintalaatikon, ja näin se toimikin. Kun menee TAB-napilla tai painaa hiirellä kohtaa missä on radionappi tai valintalaatikoita niin rajaus tulee näkyviin.

Perjantai 13.11.2020

Jatkan tänään asiakas S:n radionappien ja valintalaatikoiden tehtävällä ja aloitan kontrasti muutokset. Tämän jälkeen aloitan tehtävän, missä pitää parannella navigaation fokusointia.

Yksi muutoksen aihe tässä tehtävässä oli myös kontrasti. Kontrasti oli heikko, koska radionappien sisäinen pallo, mikä merkkää mikä nappi on valittuna ja valintalaatikoiden valitsin, oli liian himmeä verrattuna taustaan. Kyseinen merkkääjä häviää tällöin taustaan ja on

vaikea huomata. Tämä saatiin korjattua myös CSS:llä ja löydettyäni oikean Less-tiedoston muutin valitsimien väriä mustaksi, jotta erottuu hyvin taustasta.

Seuraavassakin tehtävässä keskityttiin fokuointiin.

Tehtävässä näppäimistöfokos on molemmilla sivustoilla piilotettu ainakin hampurilaismenun kuvakkeella ja päänavigaation ylimmällä tasolla. Korjataan näppäimistöfokos näkyväksi kaikissa elementeissä. Tyylikorjaus sivustoille, että näppäimistöfokos on näkyvä. Nyt toimii esim. ylänavigaatiossa ”pääsivu” otsikon kohdalla, mutta ei muiden otsikoiden kohdalla.

Tässä tehtävässä navigaation nappien fokuointi ei näkynyt ollenkaan ja jos liikut TAB-näppäimellä et tiennyt ollenkaan missä kohtaa ollaan. Tässä tehtävässä minulle tuli vähän ongelmia, kun en saanut tyyli muutoksia käyttöön, vaikka kuinka muutin ja kääntäjällä rakensin tyyli tiedostot uudestaan, ei mikään muuttunut. Enkä millään meinannut löytää oikeaa paikkaa, missä saisin fokuoinnin muutettua. Tehtävä jää kesken ja jatkan viikonloppun jälkeen.

Viikkoanalyysi

Viikon teemana oli taas saavutettavuus ja käytettävyys. Jatkoin kenttien muokkausta, niitä oli todella paljon. Uutena asiana tuli jQueryn virheviestin elementin muokkaus ja se, miten se toimii, sekä miten sitä pystyy noinkin helposti muokkaamaan (Custom HTML on error element #2170). Kenttien muokkaamisessa oli iso työ. Alkuun työ tuntui loputtomalta ja ennen kuin ehdin tutustumaan asiaan syvemmin, työ tuntui vähän turhalta. Mitä syvemmälle pääsin tutkimuksissa, sitä enemmän ymmärsin kuinka tarpeellisia nämä ovat henkilöille, jotka eivät käytä verkkosivuja samalla tavalla kuin minä.

Viikon lopulla tehtävät keskittyivät näppäimistön käyttöön ja kontrasteihin.

Yleisiä painikkeita näppäimistöllä toimivilla sivustoilla ovat esimerkiksi tabulaattori, jolla liikutaan linkeistä, lomake-elementeistä tai painikkeista toisiin. Shift + Tab -yhdistelmällä liikutaan edelliseen elementtiin. Enter-painikkeella aktivoidaan valittu linkki tai painike.

Space-painiketta käyttämällä valitaan valintalaatikoita tai aktivoidaan painike, joka on fokuoituna. Ylös ja alas nuolinäppäimillä liikutaan radionappien tai alasetoalikkoiden elementtien välillä. Oikealle ja vasemmalle nuolinäppäimillä voidaan liikkua joissain tapauksissa valikkojen elementtien välillä tai säätää ääni ja video elementtien liikusäätimiä. Escape-painikkeella suljetaan elementtejä (Anniina Saapunki 2018).

Koska monet käyttäjät eivät käytä verkkosivustoilla liikkumiseen hiirtä, on sivustot tehtävä saavutettaviksi myös näppäimistöllä. Kaikkien interaktiivisten elementtien tulee olla saatavilla näppäimistöä käyttämällä. Elementille tulee lisätä focus-attribuutti käyttäen CSS:ää, sillä ilman indikaattoria siitä, mikä elementti milloinkin on valittuna, minkä takia on palvelua on mahdoton käyttää näppäimistöllä (Anniina Saapunki 2018).

Tiettyjä väriyhdistelmiä on myös syytä välttää, sillä ne voivat olla vaikeasti havaittavissa esimerkiksi puna-vihersokeille. Värien ja kontrastien tarkastukseen on olemassa paljon erilaisia työkaluja. Käytin tällä viikolla kontrasti konetta, joka kertoi minulle, mitkä väriyhdistelmät ovat hyviä ja mitkä huonoja. Näistä valitsin mustan valitsimen vaalean siniseen taustaan (Anniina Saapunki 2018).

3.4 Seurantaviikko 4

Maanantai 16.11.2020

Jatkan Asiakas S:n navigaatio fokusointi tehtävän parissa ja sen jälkeen valitsen jonkun uuden tehtävän.

Seuraavassa tehtävässä tuotehaussa käytetyillä kuvakkeilla ei ole tekstivastineita, sen paremmin kuin myöskään tilauslomakkeen vaiheita indikoivilla pallukoilla sivun oikeassa laidassa. Lisään kuvakkeille tekstivastineet.

Tuotehaun koodiin on lisättävä tekstivastineet tarvittaville elementeille, jotta sisällöntuotannon Media-kansion kuvakkeille määritellyt tekstivastineet näkyisivät sivustolla.

Navigaation fokusointi tehtävän sain valmiiksi ja olinkin etsinyt kyseistä CSS-luokkaa väärästä paikasta. Oikean tiedoston löytyessä lisäsin Focus luokan ja sille rajauksen. Navigaatio sai mustat reunat fokusoituaan.

Toisessa tehtävässä tuotehaku sivulla kuvilla ei ollut tekstivastineita ja nämä piti lisätä. Koodissa kuitenkin oli FOR rakenne, joka kävi listan tuotteita läpi ja jokaisella tuotteella oli eri kuva riippuen tuotteesta. En voinut laittaa siihen kiinteää tekstiä, koska jokaisella kuvalla olisi sitten ollut sama teksti, vaikka olivatkin eri kuvia. Otin FOR rakenteesta sillä hetkellä olevan objektin ja siitä otin tekstivastineen kuvalle, ja näin sain jokaiselle kuvalle oman tekstivastineen..

Tiistai 17.11.2020

Tämä on saavutettavuusraportin ulkopuolinen asia, jossa asiakas toivoo parannusta siinä, miten tietotaulukot toimivat kapeilla mobiilinäytöillä. Nykyisellään taulukot eivät toimi hyvin mobiilinäytöllä. Taulukko on liian leveä mobiilinäytölle, eikä taulukkoa ole mahdollista vierittää sivusuunnassa. Etsitään tähän toimiva ratkaisu, seuraavista vaihtoehdoista:

- Mahdollistetaan taulukon vierittäminen sivusuunnassa, jos ei mahdu näytölle
- Sisältöä skaalataan niin, että mahtuu näytölle
- Taulukko esitetään kahdella rivillä, jos rivi ei mahdu näytölle

Tehtävässä taulukko oli laajennettu niin isoksi, että se ei responsiivisesti reagoinut resoluutiomuutoksiin. Taulukot tuli reunoista yli, kun ei oltu kokonäyttö tilassa. Lähdin ensin tutkimaan, miten saan taulukosta responsiivisen ja löysin, että muuttamalla display komentoa tämä onnistuisi, mutta piti myös korjata tämä yhden ruudun laajennus.

Taulukon laajennus aiheutti sen, että taulukko oli liian suuri eikä reagoinut resoluution muutoksiin. Etsin luokan missä tämä laajennus säädetään ja pienensin sitä hieman, sekä lisäsin porrastetusti @media luokalla, joten kun resoluutio menee tietyn asteen alle, niin laajennus komento muuttuu myös hieman pienemmäksi, jolloin taulukko mahtuu koko ajan HTML elementin sisään.

Toisessa kohdassa oli niin, että kun resoluutio menee liian pieneksi, se häviää näytön reunasta eikä ole saavutettavissa eikä liikuteltavissa. Ohjeissa oli, että taulukko hajoaisi kahden riviin näin käydessä, mutta en nähnyt sitä hyvänä vaihtoehtona. Tutustuin erilaisiin mahdollisuuksiin lisätä vierityspalkkia, jolla pääsisi vierittämään horisontaalisesti koko taulukkoa.

Käytin taas @mediaa hyväksi, sillä kun resoluutio menee tarpeeksi pieneksi, muutan display komentoa ja taulukon virtausta, jotta saadaan taulukon sisältöä liikutettua horisontaalilla tasolla.

Keskiviikko 18.11.2020

Tänään jatkan taulukoiden muokkaamisella. Seuraavassa tehtävässä taulukot eivät käytä otsikkorivejä ollenkaan vaan kaikki on rivi kenttiä. Lisään taulukoille otsikkorivit ja tarvittaessa otsikkosarakkeet TH-elementtejä hyödyntäen.

Teen tyylikorjauksen, jotta taulukoiden otsikot näkyisivät. Sen jälkeen sisällöntuotannossa sisältöeditorissa syötetyt taulukot käyttäisivät taulukko-otsikkoelementtejä.

Uudessa tehtävässä keskityttiin taulukon otsikko elementtiin. Huomasin, että CSS-luokassa on myös tehty valmiiksi tyylimäärittelyt otsikko elementeille, mutta ne eivät toimineet itse taulukossa. Pienen tutkimisen jälkeen huomasin, että taulukko ei käytä ollenkaan otsikko rivejä ollenkaan, kun taulukkoa rakennetaan sivulle.

Aloin etsimään paikkaa missä koko taulukko kootaan ja päädyin episerverin sisällöntuotannon sivulle, missä oli Excel-tyylinen työkalu, jossa taulukko tehdään. Tässä vaiheessa rivejä ja soluja ei ollut muokattu siten, että ensimmäinen rivi olisi otsikkosolu vaan oli jätetty normaaliksi soluksi. Ongelma ratkesi, kun muutin ensimmäisen rivin solun otsikko soluksi ja taulukon otsikko otti käyttöön otsikko tyylit. Kerromme asiakkaalle, että nämä pitää sisällöntuotannoissa merkitä otsikkoriveiksi muuten ne eivät toimi.

Torstai 19.11.2020

Tänään teen kielivalikon nimien korjauksen ja meillä on Sifi (hallituksen tiedotustilaisuus) + kickoff (DE-yksikön virtuaalinen kick-off ensi vuodelle) jonka jälkeen katson itselleni seuraava tehtävän.

Ensimmäisenä sain tehtäväksi Asiakas S:n kieli muutoksen. Kyseessä on siis kielivalikon teksti, jossa tällä hetkellä lukee FI/EN. Asiakas haluaa niihin Suomi ja In English -tekstit. Tehtävä ei ollut suuri, joten sain muutettua tekstit otsikkokentässä.

Seuraavassa tehtävässä laskuri päivittyy heti kun kenttiin laittaa arvoja. Tätä logiikkaa on ruudunlukukäytössä hankala hahmottaa. Lisään laskuriin erillisen painikeen, joka laskee arvot annetuilla syötteillä. Lisään "Laske"-nappi, jota käyttäjän on painettava, kun haluaa tehdä laskutoimituksen.

Aloitin asiakas S:n laskurikorjauksen, mutta Vue ei ole minulle kovin tuttu, joten päätin aloittaa Vuen perusteista ja syntaksin opettelusta.

Hallituksen tiedotustilaisuudessa käytiin läpi toteutuneita lukuja ja miten ylipäänsä eri yksiköillä menee, ja mitä on luvassa tulevaisuudessa sekä lopuksi oli hallituksen roasting.

Myöhemmin päivällä oli DE-yksikön kick-off, jossa käytiin läpi, miten yksikön sisällä eri tiimeillä mennyt ja tavoitteista ensivuodelle. Saimme myös antaa palautetta miten omasta mielestä mennyt ja hyviä palautteita tulikin.

Perjantai 20.11.2020

Tänään keskityn Vuen opiskeluun.

Tänään jatkoin Vuen opiskelua ja sen rakenteen oppimista. En oikein ymmärrä, miten tämä laskuri toimii ja mitä eroa on Vuen method ja computed -sarakkeilla. Yritän selvittää miten laskuri toimii, tehtävässä haluttiin nappi, joka laskee tuloksen.

Koko päivä meni tähän, mutta nyt ymmärrän jotenkin mitä eroa on method ja computed -sarakkeilla. Computed suoritetaan aina, kun arvo kentässä muuttuu ja method vasta kutsuttaessa. Yritin muuttaa computed-sarakkeen methodin sisään, jotta lasku aktivoituisi vasta nappia painettaessa.

Viikkoanalyysi

Tällä viikolla pääsin taas tekemään paljon tyylimuutoksia. Nyt olen tehnyt niitä sen verran, että voisin sanoa, ainakin tässä asiakas S:n projektissa, mistä mikäkin löytyy. Mitä enemmän teen tyylimuutoksiin liittyviä tehtäviä, sitä itsevarmemmaksi tulen ja sitä mukavampaa on niitä tehdä. CSS-luokat löytyvät helposti, kun nyt olen ymmärtänyt miten tiedostojen puurakenne on tehty ja tiedostojen nimet ovat tuttuja.

Tällä viikolla opin CSS:n @meidan käyttöä ja kuinka sitä kannatta hyödyntää. Nykyään on paljon erilaisia päätelaitteita, jokainen toimii vähän eri lailla ja eri resoluutioilla, joten on tärkeää tehdä verkkosivuja siten, että jokainen päätelaite voi niitä käyttää sujuvasti (CSS @media Rule). Myös työni alkaa nopeutua tyylimuutoksia tehdessä, kun alkaa jäämään muistiin, mikä komento on mikäkin ja mitä muuttamalla pääsen tavoitteeseen ilman useita kokeilukertoja.

Tällä viikolla oli myös yhtiön tiedotustilaisuuksia, ja oli oikein kiva nähdä muitakin työkavereita, kuin oman tiimin jäseniä. Tilaisuuksissa kerrottiin yhtiön tilanteesta ja koronasta sekä etätyön jatkumisesta. Yhtiöllä menee hyvin ja oletettavasti palkataan vielä lisää työntekijöitä. Sen vuoksi käynnistetään taas digiStar-haku, jota kautta itsekin pääsin Innofactorille. DE-yksikön kickoff-palaveri järjestettiin virtuaalisena. Kyseinen tilausuus olisi pitänyt järjestää jo keväällä, mutta koronan takia se jouduttiin perumaan ja nyt järjestettiin virtuaalinen vastike. Kickoffissa oli puhumassa yksikön johtoa ja muita people managereita. Yksiköllä menee taloudellisesti hyvin, myös henkilökunta voi hyvin.

Tällä viikolla pääsin opettelemaan myös Vueta. Monessa projektissa on eri teknologioita käytössä esim. Angular, Vue, JQuery, JScript yms. Tässä käytettiin Vueta.

Ensin Vue näytti todella sekavalta, mutta toisaalta vähän samalta kuin JQuery. Jonkin aikaa ehdin opetella Vuen syntaksia ja miten se toimii, mutta vaikea asia oli sen ymmärtäminen laskuriin ja koko osion ymmärtäminen (Class and Style Bindings). Laskurissa oli monta muuttujaa ja laskukaavat todella pitkiä. Oli vaikeaa erotella sieltä mitä pitää muuttaa ja mitkä ovat itse laskukaavaa mikä pitää säilyä ennallaan.

Sitten löysin verkosta sivun (Computed Properties and Watchers), jossa kerrottiin Vuen data, method ja computed -osioista ja sen luettuani ymmärsin, miksi ne olivat erillään. Yhteinen computed-osio käytetään tässä sen takia että kun kentän arvo muuttuu, se laukee tämän osion ja tekee laskutoimituksen. Juuri tästähän me halusimme pois. Pidemmälle luettuani aloin ymmärtämään, että minun pitää luoda metodi, jonka sisällä on koko laskukaava ja tämä lasku lasketaan, kun nappia painaessa tätä metodia kutsutaan.

Vue on kyllä ainakin näin alkuun vaikeaa, mutta tämän tehtävän avulla varmasti opin lisää ja jatkossa on helpompi työskennellä Vuen kanssa.

3.5 Seurantaviikko 5

Maanantai 23.11.2020

Tänään jatkan asiakas S:n laskurin Vue-korjausta.

Jatkoin laskurin muokkausta ja sain sen valmiiksi. Tehtävässä haluttiin, että laskuri ei laske automaattisesti lukuja, kun niitä syötetään, vaan saisi rauhassa laittaa kaikki numerot ja sitten sinne lisättäisiin nappi, jota painamalla laskuri laskee tuloksen. Lisäsin napin ja nappia painamalla kutsutaan Vue-metodia, joka tekee laskusuorituksen. Poistin kokonaan computed-osion, koska tätä ei enää tarvittu. Tähän meni koko päivä, kun piti vielä opiskella, miten tuo Vuen metodi toimii ja miten saan vietyä lasketut arvot näkyviin niiden omille paikoille.

Tiistai 24.11.2020

Tänään sain tehtäväksi autocomplete määrittelyyn liittyvän tehtävän tutustun tehtävään ja aloitan tehtävän teon.

Asiakkaan sivustolla lomakkeissa ei ole autocomplete ominaisuutta tällä hetkellä. Autocomplete tarkoittaa, että käyttäjällä voi olla valmiiksi luotuja yhteystieto lomakkeita, joita selain täyttää valmiiksi lomakkeisiin näin halutessaan.

Episerver-lomakkeella autocomplete ei toimi. Sitä ei ole explisiittisesti otettu pois päältä.

Lisään Innofactorin räätälöityjen lomakkeisiin autocomplete-attribuutit.

Aloitin tehtävän, eikä ongelmia tullut. Lisäsin yhteystieto kenttiin autocomplete-attribuutin ja nimesin sen, jotta valmiiksi täytetyt lomakkeet tietävät, mihin kohtaan tulee esim. osoite, nimi ja puhelinnumero. Testasin vielä, että lomake toimii lisäämällä omat esitäytetyt osoitetiedot. Nyt lomakkeet täyttävät kohdat itse, jos niin halutaan.

Keskiviikko 25.11.2020

Tänään valitsen uuden tehtävän. Tuotehakusivulla on toisteisia Tutustu tuotteeseen ja Tila -elementtejä. Näihin elementteihin olisi saavutettavuusmielessä hyvä kytkeä kyseisen tuotteen nimi, vaikka aria-describedby attribuutilla.

Lisään tarvittavat ARIA-elementit.

Samalle asiakkaalle uusi tehtävä. Tehtävään tutustuminen ja toteutus.

Tehtävä ei ollut kovin työläs. Sain aika nopeasti kiinni, mitä lisätään ja minne, kun edellisissä tehtävissä on tehty samoja juttuja. Sain tehtävän valmiiksi ja aloin tutkimaan lisää TestCafe-tehtävää, mikä minulla on kesken plurarsightissa. Jouduin palaamana taaksepäin aika paljon koska en muistanut oikein enää mihin jäin ja on niin pitkä aika siitä, kun viimeksi tähän paneuduin.

Torstai 26.11.2020

Tänään meillä on tiimin retrospektiivi, johon on varattu melkein koko päivä.

Retron jälkeen teen asiakas S:n pyytämän korjauksen sähköposti kenttään.

Päivä meni retrossa ja siinä käytiin läpi tiimin asioita: onnistumisia ja rakentavaa palautetta. Sovittiin myös yhteiset kehityskohteet.

Asiakkaalta oli tullut korjauspyyntö sähköposti kenttään, missä kentän arvion tarkastus ei toiminut, eikä pystynyt syöttämään kuin yhden sähköpostin. Tarkistin tarkastuksen ja lisäsin kentälle attribuutin multiple joka mahdollistaa monen sähköpostin syötön kenttään.

Torstai 26.11.2020

Tänään ei ole muita tehtäviä, kuin TestCafen opiskelua ja testien tekoa.

Sain jatkettua testien tekoa ja TestCafeen tutustumista. Komentokieli ei vieläkään oikein aukea ja on haasteita kirjoittaa tai ymmärtää testien rakennetta tai miten ne toimivat. Yritin tehdä "sivua ei löydy" sivun testin, mutta en päässyt kovin pitkälle. Pyydän myöhemmin työkaverilta apua ja jatkoin Azure-sertifikaatin opiskelua.

Aiheena oli logic apit Azuressa - miten ne toimivat ja miten niitä käytetään. Aihe oli todella mielenkiintoinen, mutta pelkkiä ohjeita lukiessa en opi kovin hyvin, vaan minun olisi parempi päästä itse tekemään ja kokeilemaan, niin opin paremmin. Aiheessa kuitenkin käytiin erilaisia sovelluksen tyyppisiä ja miten niitä kutsutaan tai laukaistaan. Yksi vaihtoehto on http kutsulla tai ajastimella, jonka voi laittaa laukaisemaan sovelluksen vaikka 5 minuutin välein tai vaikka vain maanantaisin.

Perjantai 27.11.2020

Tänään työkaveri auttaa TestCafen kanssa, jos ehtii, muuten jatkan Azuren sertifikaatin opiskelua.

Työkaveri soitti ja auttoi TestCafen kanssa. Hän vinkkasi, että on olemassa Chrome DevTools: puppeteer recorder, joka generoi komentokieltä, kun sinä liikut graafisessa käyttöliittymässä. Aloin tutkimaan tätä työkalua ja miten sitä voisin hyödyntää tässä asiassa.

Viikkoanalyysi

Viikko ei ollut kovin kiireinen. Muutama pienempi tehtävä ja muuten vain itsenäistä opiskelua.

Asiakas S:n laskurin muokkaaminen oli kiva tehtävä ja opin siinä Vueta ja sitä, miten sitä voi hyödyntää. Tässä tehtävässä opin Vuen rakenteesta sen, mihin ja miten eri osioita käytetään (Computed Properties and Watchers). Esim. metodian ja computedin erot ja miten dataa käytetään näissä. Isoin ongelma taisi olla, miten saan tuloksen näytettyä samalla tavalla kuin vanhassa, eli siirtää computedista metodiin, ja miten saan puskettua tulokset HTML-elementteihin, jotta käyttäjät näkevät ne (Class and Style Bindings). Toivon, että tulee muitakin tehtäviä, joissa pääsen käyttämään Vueta.

TestCafe opiskelussa en päässyt pidemmälle. Komentokieli ei ole vahva osaamiseni ja jarruttaa pientä projektiani paljon. Itseasiassa koko TestCafe kokonaisuus on vielä aika hataralla pohjalla, kun olen vain pluralsightissa käynyt verkkokurssia (Marques Woodson 2019) ja siellä ei ole ihan samanlaista tehtävää, kuin minulla, joten on vaikeaa soveltaa

sitä omaan projektiin. Tarvitsisin ehkä semmoisen esimerkkitapauksen, missä sitä olisi käytetty samassa ympäristössä ja epserverin kanssa, jotta voisin ottaa siitä palasia ja soveltaa niitä omaan käyttöön.

Sain tosiaan työkaverilta vinkin puppeteer recorderista, jolla voisin tehdä testejä helpommin, mutta en vielä päässyt kokeilemaan sitä.

Ensin minun pitää selvittää, miten työkalu toimii ja miten sitä käytän, jotta voisin soveltaa sitä omaan projektiin. Ohjeet kyseiseen työkaluun ovat vähän huonot, joten toivottavasti pääsen käyttämään työkalua, jotta testien teko helpottuisi.

Ehdin myös vähän opiskella Azuren Logic -appeja. Aineisto on kovin teknistä tekstiä ja haluaisin päästä kokeilemaan ja tekemään itse kyseisiä toimintoja, jotta näkisin miten ne oikeasti toimivat ja käyttäytyvät. Kyseiset logic-apid kuulostavat hyödyllisiltä, kun niillä voi ilman omaa serveriä luoda toimintoja, joilla voi yhdistää palveluja toisiinsa tai suorittaa laskutoimituksia tai muita loogisia tehtäviä.

3.6 Seurantaviikko 6

Maanantai 30.11.2020

Asiakas T:n laskurien päivitys tehtävä.

Asiakkaalla on verkkosivuillaan laskureita, joihin joka vuosi muutetaan parametrit vastamaan kyseisen vuoden arvoja. Tänään tutustun laskureiden logiikkaan eli mitä pitää muuttaa ja mistä kohtaa.

Kyseinen tehtävä ei tule olemaan helppo, tänään sain osan laskureita käytyä läpi ja ne ovat tehty Angularilla. Minulla ei ole tietoa, miten lain mukaan laskukaavat toimivat ja todella heikko käsitys siitä, mitä pitää muuttaa, jotta laskurit laskevat oikein.

En edennyt tehtävässä koska minulla meni koko päivä tutustuen laskureiden toimintaan ja siihen, miten Angular käyttäytyy tässä tilanteessa. Laskureita on yhteensä seitsemän. Aion kysyä lisätietoa tehtävään liittyen.

Tiistai 01.12.2020

Tänään jatkan Asiakas T:n laskurien päivitystehtävää, ja toivon, että pääsen muuttamaan parametrejä ja kokeilemaan laskeeko laskurit oikein.

Sain lisätietoa tehtävään liittyen ja tiedon, mistä saan uudet laskurin parametrit. Nyt pääsin eteenpäin ja tiedän ainakin osan numeroista, mitkä pitää päivittää. Päivä meni logiikkaa tutkiessa ja oikeiden tietojen etsimisessä. Löysin muutaman parametrin, mikä piti päivittää, mutta osa laskureista on tehty siten, että pitää lisätä taulukkoon uudet numerot, joita laskuri osaa käyttää automaattisesti vuodesta riippumatta. Angulariakin ehdin hetken aikaa opiskella. Nyt alkaa olla hyvä käsitys siitä, miten Angular siirtää ja käsittelee tietoa.

Sain numerot vaihdettua, ja myöhemmin käymme tehtävän läpi työkaverin kanssa. Viemme muutokset testiserverille, jotta asiakas voi testata toimiiko ne.

Keskiviikko 03.12.2020

Meillä on töissä käytössä itse tehty ResourcingTool, jonne merkitään työntekijän kohdalle missä projektissa on ja kuinka kauan, sekä päivittäin tehdyt työtunnit. Tällä työkalulla pyritään laskemaan ja tarkkailemaan työllisyysastetta, jotta kaikilla olisi tarpeeksi töitä. Sain tehtävän kyseiseen työkaluun, kun siellä oli huomattu, että PowerBI-raportit näyttävät väärinä merkintöjä. Sain tehtäväksi etsiä syyn.

Päivä meni nopeasti ja löysinkin pari virheellistä asiaa. Kyseisessä työkalussa käytetään kalenterinäkymää, johon merkitään työntekijöiden projektit ja tunnit. Kun varaus varataan, niin lähdekoodi lajittelee, onko kyseinen päivä pyhäpäivä tai viikonlopun päivä, jolloin koodi poistaa kyseisen päivän. Emme halua, että sinne lisätään työpäiviä, jolloin ei olla töissä. Kyseisessä varauksessa oli semmoinen häikkä, että jos varaus alkaa lauantaina lähdekoodi lisää kyseisen aloitus päivän työpäiväksi, jolloin kyseisessä varauksessa on yksi päivä liikaa.

Toinen ongelma, minkä löysin oli "split" ominaisuudessa. Ominaisuus mahdollistaa jo olemassa olevan kirjauksen jakautua kahteen osaan. Aina kun se teki näin, lähdekoodi jää jotenkin jumiin ja tietokantaan kirjataan kahteen kertaan kirjauksen ensimmäinen osa ja toinen osa jää kokonaan kirjaamatta.

Torstai 04.12.2020

Tänään mietin miten kirjauksen ensimmäinen päivä voitaisiin jättää pois periodin tietokantakirjauksesta.

Aloin tutkimaan miten lähdekoodissa on tehty työpäivien erottelu vapaapäivistä ja huomasin, että tässä toteutuksessa on käytetty moment-business-days npm -kirjastoa. Mitä

enemmän asiaa tutkin, niin huomasin, että tämä kirjastohan on oikein loistava työkalu tähän kyseiseen tarpeeseen, mutta jonkun syyn vuoksi se jättää varaukseen kuitenkin ensimmäisen päivän, vaikka se olisi lauantai tai muu pyhäpäivä.

En keksinyt vielä ratkaisua, joten tehtävä jatkuu seuraavana päivänä.

Perjantai 05.12.2020

Päivä alkaa kesken jääneellä tehtävällä ja yritän ratkaista, miten saan lähdekoodia muutettua, jotta kirjauksen ensimmäinen päivä ei mene tietokantaan, jos se on viikonloppu tai pyhäpäivä.

Lähdekoodissa on tosiaan käytetty momentin business-day-kirjastoa, joka lajittelee työpäivät. En vielä kukaan saanut selville, miksi se ei poista päivän ensimmäistä kirjausta, jos tämä on viikonloppu päivä tai pyhäpäivä. Kokeilin että jos käyttäisin business-day kirjaston omaa metodia, joka tarkistaa onko päivä vapaa vai työpäivä ennen kuin se vie tietokantaan. Tämä metodi otti onnistuneesti pois päivän, jos se ei ollut työpäivä, vaikka se olisi kirjauksen ensimmäinen päivä.

Tämän jälkeen aloin tutkimaan, miksi split-toiminto kyykkää, eikä vie tietokantaan oikeaa tietoa. Debuggerilla kävin rivi riviltä läpi, missä tieto häviää ja miksi se ei toimi. Vian etsintä jää kesken, kun työpäivä loppui, joten jatkan ensi viikolla.

Viikkoanalyysi

Viikko oli oikein hyvä ja opin taas paljon tästä työstä. Sain tehtäväksi muuttaa asiakas T:n laskurien parametrejä ja koin, että tehtävä oli vaikea, koska minä en ole itse rakentanut laskureita eikä minulla ollut ohjeita, miten ne on rakennettu tai miten laskurien pitää laskea laskut.

Olisi ollut helpompaa, jos olemassa olisi laskurin tekijän muistiinpanot, miksi tämä tekee näin ja millä laskukaavalla nämä pitää laskea. Mutta onneksi minulle annettiin aikaa ja sain rauhassa tutkia, laskukaavoja sekä opetella Angularia (Understanding Controllers), jotta tiedän miten ohjaimet toimivat. Voi olla mahdollista, että palaan laskurien ääreen vielä, koska en ole yhtään varma menikö kaikki oikein. Mutta asiakas testaa nämä ja palaa asiaan, kunhan saamme ne testille.

Viikossa oli oikein hienoa, että sain rauhassa tehdä työtä asiakkaalle, jota voin laskuttaa ja näin kantaa oman korteni kehoon liiketoiminnan kannalta. Tehtävä oli vaikea ja mieltäni

nostattaa se, että työkaverini luottavat minuun, saan työni tehtyä ja asiakas saa palvelua, josta maksaa. Asiakkaaseen en vielä ole ollut yhteydessä, mutta jos päädyn hoitamaan tätä projektia on minun alettava olemaan myös asiakkaaseen yhteydessä.

Tällä viikolla mielestäni kehityin rauhallisempaan suuntaan, minulla ei ollut kiire saada tehtävää valmiiksi, vaan rauhassa mietin ongelmaa ja miten sen saisi ratkaistua. Niin asiakas T:n laskureissa, kuin myös ResourcingTool-tehtävän kanssa. Oli kiva päästä ensin tutkimaan ja sitten ratkaisemaan ongelmaa, ja kun ratkaisu toimi, niin siitä tuli hyvä onnistumisen tunne. Pysin jatkossakin tekemään niin, että suunnittelen ensin ja sitten teen, kuin että alan tekemään ja kokeilemaan vähän kaikkea, mikä sopisi ja toimisi.

Viikolla opin myös lisää C#-kielestä (Mohammed Ezzat) ja Angularista (Understanding Controllers), joka ei ole niin tuttu minulle. Angular ohjaimia tuli tutkittua eniten, koska niitä en ollut ennen käyttänyt.

Suurin työ viikossa oli viimeisin vianetsintä tehtävä. Split-toiminnossa on selkeästi jokin vikana, joten on suuri työ käydä askel askeleelta läpi, miksi se ei toimi. Jatkan kyseisestä tehtävästä maanantaina ja annan viikonlopun aikana hautua, mistä vika voisi syntyä.

3.7 Seurantaviikko 7

Maanantai 7.12.2020

Tänään jatkan ResourcingToolin vianetsintää, miksi split-toiminto ei toimi.

Tänään löysin ainakin yhden vian, miksi split-toiminto ei toimi kunnolla. Lähdekoodissa oli ennen tietokantaan viemistä metodi, joka jakoi kirjauksen kahtia ja loi kaksi uutta kirjausta, joka sitten vei nämä tietokantaan. Metodin lopussa oli IF-lause, joka erotteli ensimmäisen osan ja toisen osan toisistaan ja tästä vika löytyi. Siinä ei ollut kunnolla määritelty osia, vaan katsottiin vain objektin nimeä ja lähdekoodi tulkitsi osion aina samaksi osaksi ja tällöin kirjaus kirjasi aina samat tiedot. Täten tietokantaan meni aina kaksi samaa tietoa, lisäsin puoliskoille uudet tunnisteet, jota IF-lause tarkistaa ja näin sain eroteltua oikeat puoliskot ja nyt tietokantaan kirjautuu oikeat tiedot.

Tiistai 8.12.2020

Tänään tarkistetaan asiakas T:n laskurit työkaverin kanssa ja jos löytyy korjattavaa, niin korjaan ne. Jos ei löydy korjattavaa, niin viedään testiserverille muutokset ja asiakas saa alkaa testaamaan. Sitten jatkan resourcingtoolin korjausta.

Työkaveri totesi, että tehtävä on valmis testiin, joten jatkoin resourcingtoolin tehtävää. Luettuani virheviestejä löysin kohdan, mistä virhe tulee. Debuggasin kyseisen kohdan ja virheviestissä luki, että käytössä oleva kalenterityökalu ei löydä kyseisellä tunnisteella mitään, jolloin koko koodi pysähtyy. Taustalla koodi siis pysähtyy, mutta käyttäjäliittymä sen sijaan jatkaa toimimista, jolloin koko systeemi menee sekaisin ja tietokantaan menee väärä tietoja.

Aloin tutkimaan miksi kyseisellä tunnisteella ei löydy kirjausta, vaikka semmoinen pitäisi löytyä.

Keskiviikko 9.12.2020

Tänään jatkan resourcingtoolin tehtävää ja selvitän miksi tunnistetta ei löydy.

Etsin lähdekoodista, miksi tunnistetta ei löydy ja löysin vastauksenkin. Kyseinen tunniste, mitä haettiin, kuuluu kirjaukselle mikä jaettiin kahtia, mutta lähdekoodissa kyseinen kirjaus poistetaan heti jakamisen jälkeen ja myöhemmässä osiossa tunnistetta ei ole enää olemassa.

Lähdin ratkaisemaan ongelmaa siten, että etsin ensin missä molemmat puoliskot luodaan ja niiden tunnisteet. Muutin kyseisessä metodissa tämän vanhan tunnisteiden uusiin ja nyt, kun jakauksen tekee koodi ei jumii ja tietokantaan menee molemmat puoliskot.

Torstai 10.12.2020

Tänään on palaveri työkaverin kanssa resourcingtooliin liittyen ja päätetään mitä tehdään, kun olen saanut lähdekoodin korjattua.

Palaverin jälkeen tultiin tulokseen, että koodi toimii ja sen voi viedä testiin muiden testattavaksi. Tietokantaongelmaa ei ole vielä korjattu. Tietokannassa on väärää tietoa ja ne pitää korjata. Seuraava tehtäväni koski tietokannan korjausta ja minun pitäisi tehdä koodi, joka korjaisi väärät rivit tietokannasta.

Aloitin ensin tutkimalla SQL-kieltä, että saisinko sillä tehtyä koodin, jonka voisin syöttää suoraan SSMS-työkaluun ja se korjaisi ongelman. Tutkittuani asiaa, totesin ettei

tietotaitoni SQL:ssä riitä, eikä SQL oikein taivu kaikkiin korjauksiin, mitä vaadittaisiin. Joten päätin, että teen Visual Studiossa konsoli sovelluksen C#-kieltä käyttäen.

Perjantai 11.12.2020

Tänään aloitan konsoli sovelluksen tekemisen ja teen taustatutkimuksia, miten saan yhdistettyä konsoli sovelluksen azure-tietokantaan.

Console apin tekeminen on todella mielenkiintoista. Ensin kävin selvittämässä käyttäjätunnuksen ja salasanan sekä URL-osoitteen, mitkä liitän yhdistys asetuksiin. Tällä yhteys asetuksilla haetaan yhteys tietokantaan. Tämän jälkeen voi hakea SQL:llä tietoa kannasta ja muokata sitä haluamallaan tavalla.

Aloin miettimään, miten minä tiedän missä riveissä on väärää tietoa ja miten ne saisi korjattua. Tulin siihen tulokseen, että poistan kaikki varaus kirjaukset ja luon ne uudestaan.

Viikkoanalyysi

Tällä viikolla sain tehdä asiaa, josta todella pidän: Back-End koodia ja tietokannalla asioita. Tällä osa-alueella olen vahvimmillani, vaikka siitä on aikaa, kun olen tänne päässyt koodia tekemään. Vähän tarvitsi muistutella, miten asiat toimivatkaan, ja mikä oli minkäkin komento, mutta sain yhteyden tietokantaan ja haettua tietoa sieltä (Tutorial: Create a simple C# console app in Visual Studio ja SqlConnection.ConnectionString Property).

SQL-kieli oli melkein kokonaan unohduksissa, mutta nopean kertauksen ansioista pääsin taas kärryille, mitä milläkin komennolla tehdään (Introduction to SQL). Hain tietokannasta tietoa ja aloin sitä käsittelemään saadakseni objektin tiettyyn muotoon. Se olikin aika haastavaa, kun en millään muistanut mitä eroa on float tai double -numeroilla ja tietokanta ei millään antanut päivämääriä oikeassa muodossa, joten jouduin niitäkin formatoimaan ja muokkaamaan.

Kun sain haettua varauksien rivit kannasta ja luotua niistä kaikista listan, oli seuraavana tehtävänä luoda luokka, johon aloin kasaamaan varauksien listasta tietoja, jotka jossain vaiheessa viedään kantaan. Luokassa ei ole paljoa tietoa, ainoastaan päivämäärä, mihin varauksen päivä sijoittuu, aika (minuuteissa) ja tunniste jotta tiedetään mihin varaukseen rivi kuuluu. Tämän jälkeen aloin luomaan FOR rakennetta, joka käy kaikki varauksen listan rivit läpi ja luo niistä luokan mukaisia objekteja, jotka viedään sitten kantaan, kun vanhat ovat poistettu.

Kuluneella viikolla oli oikein mukava tehdä töitä, kun pääsin ensin ratkomaan ongelmia ja sitten luoda niihin ratkaisuja. Tulee aina hyvä mieli, kun oma ratkaisu toimii. Konsoli sovelluksen tekeminen oli oikein mukavaa ja paluu omaan elementtiin (Tutorial: Create a simple C# console app in Visual Studio). Olen paljon tehnyt vain tyylimuutoksia tai muita käyttöliittymä juttuja. On nekin kyllä mukavaa hommaa, mutta mielellään teen enemmän Back-End hommia ja logiikka muutoksia. Konsoli sovelluksen teossa opin taas paljon C#-kieltä, kun Java on ennen ollut vahvin kieleni, mutta C# on yhtä hyvä ja monipuolinen kieli, jota mielelläni opin lisää. C#-kielessä on paljon omia metodeja, joita ei tarvitse itse aina tehdä. Kyseinen konsoli sovellus oli ensimmäinen, minkä olen tehnyt itse alusta asti, enkä malta odottaa, että saan sen valmiiksi ja voin käyttää sitä korjaamaan tietokantarivejä.

Asiakas T:n laskuritehtävästä ei vielä ole tullut palautetta eikä korjauspyyntöjä. Semmoinen fiilis on, että tulee vielä jotain korjattavaa, mutta se jää nähtäväksi. Oli oikein kiva päästä asiakkaalle tekemään työtä ja enemmän kiinni asiakkaaseen. Toivottavasti jatkan Asiakas T:n tehtävien tekemistä, sillä silloin tunnen pääseväni syvemmälle osaksi tiimiä ja pääsen tekemään työtä, jota käytetään ja halutaan.

3.8 Seurantaviikko 8

Maanantai 14.12.2020

Sairaana

Tiistai 15.12.2020

Sairaana

Keskiviikko 16.12.2020

Sairaana

Torstai 17.12.2020

Sairas

Perjantai 18.12.2020

Sairaana

Viikkoanalyysi

En saanut töitä tehtyä, kun olin koko viikon sairaana.

3.9 Seurantaviikko 9

Maanantai 21.12.2020

Edellisellä viikolla on tullut laskureista korjausehdotuksia ja alan niitä tekemään tänään.

Olin ymmärtänyt laskurin logiikan väärin. Siellä on pohja-arvo, mikä kerrotaan vuosittain vaihtuvalla kertoimella, ja minä olin mennyt muuttamaan tätä pohja-arvoa, koska luulin siihen menevän tämän pohja-arvon ja kertoimen summan. Nyt laskuri laski väärin koko summan ja tämä minun piti korjata.

Sain myös palautetta, että vuoden 2021 summaa ei voi vielä laskea, mutta siellä lähdekoodissa on semmoinen metodi, että sinne voi lisätä vuosia eteenpäin, vaikka kuinka paljon, mutta laskuri antaa vain tähän kyseiseen vuoteen asti niitä vaihtoehtoja. Eli vuosi 2021 tulee vasta näkyviin, kun vuosi vaihtuu. Korjasin tämän hetkellisesti, jotta he pääsevät tarkastamaan, että vuosi 2021 laskee oikein.

Tiistai 22.12.2020

Tänään jatkan recourcintool konsoli sovellusta ja mikäli tulee Asiakas T:n laskuriongelmaa, korjaan niitä.

Sain tänään tehtyä konsoli sovellusta ja pääsin siinä eteenpäin hienosti. Sain FOR rakenteen hyvin toimimaan ja nyt minulla on kaikista varauksista luokan mukaiset listat, jotka aion seuraavaksi lähteä viemään tietokantaan. Aloin tutkimaan, miten saisin vietyä tehokkaasti monta riviä kerrallaan ja kokeilin tämmöistä merkkijonon rakentajaa, joka ei sitten johtanut mihinkään. Tämä toiminto lisäsi vain yhden rivin ja meni sitten jumiin.

Päädyin ratkaisuun, jossa FOR rakenteessa vien jokaisen listan rivin kantaan yksitellen. Tässä menee kyllä kauan ennen kuin kaikki listan rivit on viety, mutta näin se ainakin toimii.

Keskiviikko 23.12.2020

Asiakas T:ltä tuli vielä korjauspyyntöjä laskureita koskien. Aloitan niistä ja jatkan konsoli sovellusta, jos jää aikaa.

Asiakas sanoi, että yksi laskuri laskee vielä vanhan vuoden arvoilla ja summa on väärä. Aloitin tutkimaan, mistä kyseinen vika johtui. Löysinkin vian, laskurin yhdessä kohdassa katsotaan IF-lausekkeella kumpaa vuotta laskuri käyttää. Lisäsin laskurin kyseiseen rakenteeseen, jotta se käyttää vuoden 2021 arvoja. Toivottavasti se nyt toimii ja ilmoitin asiakkaalle, että se on korjattu ja he voivat testata uudestaan.

Torstai 24.12.2020

Jouluaatto

Perjantai 25.12.2020

Joulupäivä

Viikkoanalyysi

Viikko oli lyhyt, kun keskelle viikkoa sattui joululoma. Sain vietyä konsoli sovellusta vähän eteenpäin ja sitä jatketaan joululoman jälkeen. Konsoli sovellusta oli taas kiva tehdä, kun ongelmia ei ole paljoa tullut. Tutkin ja kokeilin miten saisin yhdellä rivillä vietyä koko tekstin SQL tietokantaan ja päädyin merkkijonon rakentajaan kokeilemaan (SQL String Builder Query in C# with Optional Parameters). Tällä tekniikalla en saanut jokaista riviä vietyä tietokantaan vaan ensimmäisen rivin jälkeen tietokannasta tuli virheviesti.

Päädyin ratkaisuun, että teen FOR rakenteen, joka käy taulukon läpi ja vie yksi kerrallaan rivin tietokantaan. Tämä ratkaisu on aika hidas ja kuormittaa tietokanta palvelinta. Tällä hetkellä vasta kokeilen sovellusta testi palvelimella ja jo siellä kyseisen rakenteen läpi meneminen kestää jo minuutin. Oikeassa ympäristössä on noin 500 000 riviä, jotka pitää yksitellen sinne viedä ja tässä menee jo kymmeniä minutteja. Onneksi tätä sovellusta ei käytetä aktiivisesti vaan yhden kerran korjaamaan väärät rivit tietokannasta, joten epäilen että haittaa jos siinä vähän kestää. Kun olen saanut sovelluksen valmiiksi, pyrin korjaamaan tietokannan illalla tai viikonloppuna jotta kukaan ei tule korjauksen aikana syöttämään tai muuttamaan tietoja tietokannassa.

Laskureista tuli taas palautetta ja ne pyrin korjaamaan. Toivottavasti asiakas on nyt tyytyväinen muutoksiin ja voimme viedä ne tuotantoon joulun jälkeen. Olin vahingossa mennyt muuttamaan pohja arvoa, joka on joka vuosi sama mutta kerrotaan aina vuosikertoimella. Nyt kun olin mennyt muuttamaan arvoa oli oikea jo kerrottu arvo aivan liian suuri eikä laskuri toiminut oikein.

3.10 Seurantaviikko 10

Maanantai 28.12.2020

Lomapäivä

Tiistai 29.12.2020

Lomapäivä

Keskiviikko 30.12.2020

Lomapäivä

Torstai 31.12.2020

Lomapäivä

Perjantai 01.01.2021

Uudenvuoden päivä

Viikkoanalyysi

Joululoma ja uusivuosi.

4 Pohdinta ja päätelmät

Päiväkirjan kirjoittamisen aikana olen omasta mielestäni kehittynyt paljon ohjelmistokehittäjänä. Tiimin työskennellessä etänä ja tiimin uutena jäsenenä oli aika vaikea päästä tiimiin sisään ja oppia tiimin käytännöt ja yhteiset pelisäännöt. Mutta onneksi tiimi myös ymmärsi tämän ja antoi tukensa aina, kun sitä tarvitsin ja oli valmiina auttamaan. Tiimityöskentely oli helppoa sen jälkeen, kun tutustuin paremmin tiimin jäseniin, jolloin avun kysymisen kynnyks madaltui enkä tuntenut itseäni enää hölmöksi pyytäessäni apua yksinkertaisiin kysymyksiin. Etänä työskentelyssä on kyllä etunsa, mutta itse haluaisin enemmän olla toimistolla, mutta pitäisin etätö mahdollisuuden vapaana, esim. jos olisi 2–3 työpäivää toimistolla ja loput etänä.

Kirjoittamisen aikana minulla oli paljon CSS3-tyylimuutoksia, joita en oikein osannut päiväkirjan alussa kovinkaan hyvin. Mitä enemmän näitä tehtäviä tuli, niin huomasin, että eivät nämä olekaan kovin hankalia toteuttaa ja joka kerta löysin helpommin tiedostosta tai luokasta sen mitä muutetaan ja vielä senkin, minkä komennon tarvitsen, jotta lopputulos olisi oikea.

Kirjoituksen aikana voin todeta, että olen tullut paremmaksi HTML5, CCS3 ja Episerverin käyttäjäksi. Nyt tavallaan vähän jo toivon, että saisin tehdä näitä käyttöliittymän puolen töitä enkä enää liiku epämukavuusalueellani ja tulen paremmaksi joka tehtävän jälkeen.

Mitä tulee ketterään ohjelmistokehitykseen, olen huomannut, että tämä ei ihan vastannut odotuksiani, miten ketterässä tiimissä toimitaan. En tarkoita, että huonoon suuntaan vaan sitä, että voi tehdä töitä ilman, että kukaan sitä sen enempään seuraa ja voin luottaa siihen, kun tiimissä sovitaan, että tämä asia tehdään, se tulee myös tehdyksi. Varsinkin näin etänä työskentelyssä ketterän toimintatavat korostuvat vieläkin enemmän. Mutta minuun se ei vaikuta mielestäni huonolla tavalla. Pystyn tekemään töitä samalla tavalla kuin toimistollakin ja jos on asiaa työkaverille, niin hän on vain viestin tai soiton päässä.

Ketterä työtapana on paperilla aika erilainen kuin käytännössä. Meidän tiimimme toimii ketterästi, mutta ei ihan kokonaan scrum-tyylillä. Muutamia vapauksia olemme soveltaneet, jotta tiimi pysyisi toimintakykyisenä ja asiakkaat saisivat parasta mahdollista palvelua sekä heidän verkkosivunsa kehittyisivät.

Olen oppinut päiväkirjan kirjoittamisen aikana paljon. Koulussa java oli minun vahvin kieleni, mutta nyt kirjoittamisen aikana vahvimmaksi kieleksi on varmasti noussut C#. Sitä olen käyttänyt paljon ja se on aika pitkälle samanlaista kuin java. Olio-ohjelmoinnissa ei

ole tapahtunut suuria kehityksen askelia, kun en ole saanut siitä isompaa tehtävää, jossa tarvittaisiin logiikkaa ja back-end tietämystä. Joitain käyttöliittymän puolen tehtäviä sain kirjoituksen aikana tehdä, mutta ne olivat suhteellisen nopeita eikä tuottanut siten päänvaivaa tai hurjaa kehitystä.

Jatkokehitys mahdollisuuksia on todella paljon, olen nyt vasta raapaissut pintaa asiakasprojektien osalta, ja suuntia on vaikka, kuinka monta. Voisin alkaa kehittämään käyttöliittymä osaamistani, jotta voisin tehdä asiakkaille vieläkin hienompia tyylejä ja ulkoasuja. Tai sitten syventää osaamistani vielä enemmän varsinkin Back-End puolella. .NET alustalla voisin vielä paremmin ja tehokkaammin viedä ja hakea tietokannasta tietoa ja käsitellä niitä oikein ja turvallisesti.

Mutta taidan kuitenkin jatkaa jo olemassa olevalla kehityspolulla eli Azure pilvipalveluiden kanssa. Haluan oppia niiden käyttöä ja olemassaoloa ja yhdistää ohjelmistokehityksen moderniin pilveen. Azuressa on paljon opiskeltavaa ja tällä hetkellä opiskelen azure development -sertifikaattia ja se on todella mielenkiintoista. Vaikka tämä vie paljon aikaa en kuitenkaan halua jättää muita kehitysalueita. Käyn silloin tällöin pluralsightissa verkkokursseja liittyen Microsoft teknologioihin ja niiden uusiin päivityksiin liittyen ASP .NET ja Visual Studioon.

Kokonaisuudessaan matka on ollut todella kiva ja omasta mielestä olen pärjännyt hyvin. Mitkään työtehtävistä eivät ole olleet sellaisia, että olisi pitänyt luovuttaa tai jättää tekemättä. Työ ohjelmistokehittäjänä on todella monipuolista, kun opeteltavaa on niin paljon, ja oma motivaatio oppia on suuri. Välillä itsekkin tiedostan, että olen haukannut liian suuren palan ja joudun vähän jarruttelemaan. Esim. itsenäiset opiskelut, saatan alkaa opiskelemaan montaa asiaa samaan aikaan ja sitten opin molemmista vähän sinnepäin asioita, kun voisi keskittyä siihen yhteen ja oppia sen kunnolla.

Tällä ajanjaksolla en ole pelkästään oppinut koodaamaan, vaan myös paljon tiimin sisäisistä työskentelytavoista. Tiimin kommunikaatio määrää aika paljon sen mitä tehdään ja missä. Kun projekteja on paljon, on vaikea pysyä perässä, missä mitäkin tehdään ja mikä on kenelläkin kesken. Tiimin dynaaminen rakenne mahdollistaa sen, että tiimi toimii hyvin silloinkin, kun on kiire. Jos jossain tarvitaan enemmän työvoimaa, niin sinne sitä siirretään ja jos ei ole paljon hommia niin voidaan tehdä töitä muihin projekteihin. Jokainen on koko ajan kärryllä, missä vaiheessa projektit ja niiden eri tehtävät ovat. Ketterä ohjelmistokehitys myös turvaa sen, että ei ole montaa tuntia palavereita päivän aikana, vaan sprintin alussa sovitaan tehtävät ja niiden sisältö. Sen jälkeen on myös aikaa tehdä niitä tehtäviä. Jos jokin muuttuu, se siirretään seuraavaan sprinttiin.

Lähteet

Computed Properties and Watchers. Luettavissa: <https://vuejs.org/v2/guide/computed.html>

Anniina Saapunki 2018. Ammattikorkeakoulututkinnon opinnäytetyö. Luettavissa: https://www.theseus.fi/bitstream/handle/10024/155429/anniina_saapunki.pdf?sequence=1&isAllowed=y Luettu: 05.11.2020

Christofer Schmitt 2009. CSS Cookbook 3rd Edition. O'Reilly Media, Inc.

Shelley Powers 2015. JavaScript Cookbook 2nd Edition. O'Reilly Media, Inc.

Mohammed Ezzat. Learning C# 8 and .NET Core 3.0 Luettavissa: <https://www.packtpub.com/product/learning-c-8-and-net-core-3-0-video/9781789617542>

Input Purposes for User Interface Components. Luettavissa: <https://www.w3.org/TR/WCAG21/#input-purposes>

SQL String Builder Query in C# with Optional Parameters. Luettavissa: <https://stackoverflow.com/questions/35757991/sql-string-builder-query-in-c-sharp-with-optional-parameters>

HTML Tutorial. Luettavissa: <https://www.w3schools.com/html/default.asp>

Scott Allen 2019. ASP.NET MVC 5. Luettavissa: <https://app.pluralsight.com/paths/skill/mvc5>

Custom HTML on error element #2170. Luettavissa: <https://github.com/jquery-validation/jquery-validation/issues/2170>

CSS @media Rule. Luettavissa: https://www.w3schools.com/cssref/css3_pr_media-query.asp

Class and Style Bindings. Luettavissa: <https://vuejs.org/v2/guide/class-and-style.html>

Marques Woodson 2019. End-to-end Web Testing with TestCafe: Getting Started. Luettavissa: <https://app.pluralsight.com/library/courses/testcafe-end-to-end-web-testing-getting-started/table-of-contents>

Understanding Controllers. Luettavissa: <https://docs.angularjs.org/guide/controller>

Tutorial: Create a simple C# console app in Visual Studio. Luettavissa: <https://docs.microsoft.com/en-us/visualstudio/get-started/csharp/tutorial-console?view=vs-2019>

SqlConnection.ConnectionString Property. Luettavissa: <https://docs.microsoft.com/en-us/dotnet/api/system.data.sqlclient.sqlconnection.connectionstring?view=dotnet-plat-ext-5.0>

Introduction to SQL. Luettavissa: https://www.w3schools.com/sql/sql_intro.asp