



Web-ohjelmistokehittäjän päiväkirja

Sampo Jaakkola

Haaga-Helia ammattikorkeakoulu

Tradenomi tietojenkäsittely

Amk-opinnäytetyö

2025

Tiivistelmä

Tekijä(t) Sampo Jaakkola
Tutkinto Tradenomi tietojenkäsittely
Raportin/Opinnäytetyön nimi Web-ohjelmistokehittäjän päiväkirja
Sivu- ja liitesivumäärä 52 + 0
<p>Tämän päiväkirjamuotoisen opinnäytetyön tarkoituksena on esitellä ohjelmistokehittäjän työtehtäviä ja arkea pienehkön yrityksen palveluksessa, jonka kehittämä ohjelmisto mahdollistaa mobiilisovellusten kehittämisen ja julkaisemisen ilman tarvetta tuottaa omaa ohjelmakoodia. Tekijän työnkuvaan kuuluu ohjelmiston käyttöliittymän kehittäminen ja ylläpito, eli uusien ominaisuuksien ja toimintojen mahdollistaminen käyttöliittymässä sekä käyttöliittymän toiminnallisuuden parantaminen ja virheiden korjaus. Käyttöliittymä on toteutettu JavaScript-ohjelmointikielellä ja sen React- ja Redux-kirjastoja hyödyntämällä.</p> <p>Työn tavoitteena on edesauttaa tekijän ammatillista kehittymistä erityisesti kolmella eri osa-alueella, eli työssä käytettyjen teknologioiden ja työkalujen kehittyneemmässä käyttämisessä, virheettömämmän ohjelmakoodin kirjoittamisessa ja kattavammassa testaamisessa sekä oppimisessa kokeneemmilta kollegoilta. Työssä ei ole analysoitu tekijän kehitystä aiemmin mainitun työnkuvan ulkopuolelta.</p> <p>Työssä käytetty tietoperusta koostuu pääasiassa JavaScriptin, Reactin, Reduxin, HTML:n ja CSS:n virallisesta dokumentaatiosta. Lähteinä käytetään myös kirjaa Eloquent JavaScript 2nd Edition, joitakin IT-alan tunnettuja keskustelupalstoja ja oppaita, kuten Stack Overflow ja GeeksforGeeks, joitakin ulkoisten komponenttikirjastojen dokumentaatioita sekä joitakin yksittäisiä artikkeleita.</p> <p>Työ koostuu johdannosta, lähtötilanteen kuvauksesta, seurantajakson raportoinnista viikkoanalyysiineen ja pohdinnasta. Seurantajakso on toteutettu kahdeksassa viikossa aikavälillä 13.1.-7.3.2025 ja jokaisen viikonpäivän raportointi on tehty päiväkirjamaisina merkintöinä, joissa on ensin asetettu päivälle tavoite ja sen jälkeen arvioitu tavoitteessa onnistumista. Jokaista viikkoa kohden on tehty viikkoanalyysi, jossa esitellään tarkemmin viikon aikana esiintyneitä ongelmia ja niiden ratkaisuita sekä tietoperustaan perehtymistä. Työn päättävässä pohdinnassa tekijä analysoi onnistumistaan ammatillisen kehityksen tavoitteissaan, työn aikana oppimistaan sekä tulevia kehityskohteitaan.</p>
Asiasanat Ohjelmistokehitys, ammatillinen kehittyminen, käyttöliittymä.

Sisällys

1	Johdanto	1
1.1.1	Käsitteet.....	1
1.1.2	Tietoperusta.....	3
2	Lähtötilanteen kuvaus.....	4
2.1	Oman nykyisen työ analysointi	4
2.2	Sidosryhmien esittely	5
2.2.1	Sisäiset sidosryhmät	5
2.2.2	Ulkoiset sidosryhmät.....	6
2.3	Työpaikan vuorovaikutustilanteet	6
3	Seurantajakson raportointi viikkoanalyysineen	8
3.1	Seurantaviikko 1.....	8
3.2	Seurantaviikko 2.....	12
3.3	Seurantaviikko 3.....	17
3.4	Seurantaviikko 4.....	21
3.5	Seurantaviikko 5.....	25
3.6	Seurantaviikko 6.....	30
3.7	Seurantaviikko 7.....	34
3.8	Seurantaviikko 8.....	38
4	Pohdinta	45
4.1	Kehityskohteet ja tulokset.....	45
4.2	Yhteenveto.....	48
	Lähteet.....	49

1 Johdanto

Toteutin päiväkirjan seurantajakson aikavälillä 13.1.-7.3.2025. Työskentelin viikon jokaisena arkipäivänä toimistollamme Helsingin Kaisaniemessä. Yrityksen pääasiallinen tuote on ohjelmisto, jonka avulla asiakkaat voivat itse luoda mobiilisovelluksia ilman tarvetta tuottaa omaa ohjelmakoodia. Oma toimenkuvani on ohjelmiston käyttöliittymän kehittäminen ja ylläpito. Käytännössä tämä tarkoittaa uusien ominaisuuksien ja toimintojen mahdollistamista käyttöliittymässä sekä käyttöliittymän toiminnallisuuden parantamista ja siinä esiintyvien virheiden korjaamista. Työssäni tarvittavaa osaamista ovat muun muassa JavaScript-ohjelmointikielen, sen React- ja Redux-kirjastojen, Visual Studio Code-ohjelmointityökalun ja GitLab-versiohallintatyökalujen sujuva käyttö.

Ammatillisen kehittymisen tavoitteitani ovat työssä käytettävien työkalujen ja teknologioiden syvämpi omaksuminen ja tehokkaampi soveltaminen, virheettömämmän ohjelmakoodin kirjoittaminen ja lopputuloksen varmentaminen sekä muihin kollegoihin tukeutuminen ja näiden kokemuksesta oppiminen. Kehittymiseni tavoitteet rajoittuvat seurantajakson aikana vain käyttöliittymän kehittämistä ja ylläpitämistä edistävään osaamiseen. Taulukossa 1 on esitelty tavoitteet, niihin liittyvät tietoperustan luvut, seurantaviikot ja tulokset.

Taulukko 1. Peittomatriisi päiväkirjaopinnoista tekstin sisäisistä kytköksistä

Oman ammatillisen kehittymisen tavoitteet	Tietoperustan luku raportissa	Seurantaviikko	Oman ammatillisen kehittymisen tulokset
Työssä käytettävien työkalujen ja teknologioiden omaksuminen ja tehokkaampi soveltaminen	2.1	Viikot 1, 2, 3, 4, 5, 6, 7, 8.	Luvut 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 4.
Virheettömämmän ohjelmakoodin kirjoittaminen ja lopputuloksen validointi	2.1	Viikot 1, 2, 6, 8.	Luvut 3.1, 3.2, 3.6, 3.8, 4.
Muihin kollegoihin tukeutuminen ja näiden kokemuksesta oppiminen.		Viikot 1, 3, 5, 6.	Luvut 3.1, 3.3, 3.5, 3.6, 4.

1.1.1 Käsitteet

API – eli Application Programming Interface on joukko sääntöjä tai protokollia, joiden avulla ohjelmit voivat kommunikoida toistensa kanssa vaihtaakseen tietoja, ominaisuuksia ja toiminnallisuksia (IBM s.a. a)

Back end – termi, joka kuvaa ohjelmiston käyttäjälle näkymätöntä osaa. Käyttäjä ei välttämättä huomaa olevansa vuorovaikutuksessa tämän ohjelmiston osan kanssa, vaikka suurin osa esimerkiksi tallennetusta tiedosta säilytetään täällä. (Cambride University Press & Assessment 2025a, COMPUTING).

Front end – termi, joka kuvaa ohjelmiston käyttäjälle näkyvää osaa ja jota kautta käyttäjä on vuorovaikutuksessa ohjelmiston kanssa (Cambride University Press & Assessment 2025b, COMPUTING).

Full stack – termi, joka kuvaa sekä back end, että front end osaamista. Full stack kehittäjät osaavat toimia molemmissa ympäristöissä. (MongoDB, Inc. 2025, What is full stack fevelopment?).

GitLab – versionhallinnan työkalu, jossa voidaan hallinnoida työntekijöiden muutoksia ohjelmiston ohjelmakoodiin. Ongelmatilanteessa ohjelmisto voidaan esimerkiksi palauttaa aiempaan toimivaan versioon. (TechTarget 2025).

JavaScript – ohjelmointikieli, jota käytetään hyvin laajalti verkkosivujen toteuttamisessa. Verkkosivujen edistyneemmät toiminnot, kuten interaktiivisuus, animointi jne. on yleensä toteutettu JavaScriptillä (Mozilla s.a. a).

Jira – työkalu, jossa voidaan suunnitella, kirjata ja hallita esimerkiksi ohjelmiston ylläpitoon ja kehitykseen liittyviä tehtäviä (Atlassian 2025, What is Jira?). Yrityksessämme jokaisella kehitysryhmällä on oma osionsa, joihin kaikki työntekijät voivat luoda tehtäviä.

Muutospyyntö – GitLabissa “merge request”, toimistollamme kommunikointiin on vakiintunut GitLabin kilpailijan GitHubin termi ”pull request” ja lyhenne PR. Pyyynnössä voidaan tarkastella, analysoida ja verrata työntekijän tai työntekijöiden ohjelmakoodiin tekemiä muutoksia (GitLab 2025). Ryhmässämme pyynnön muutosten yhdistäminen olemassa olevaan ohjelmakoodiin vaatii yleensä kahden toisen työntekijän koodikatselmuksen ja hyväksynnän.

React – JavaScript ohjelmointikielen kirjasto, jolla toteutetaan käyttöliittymiä. Reactilla toteutetut käyttöliittymät koostuvat yleensä ns. komponenteista, joita voi parhaassa tapauksessa käyttää uudelleen käyttöliittymän useassa eri osassa. (Refsnes Data 2025a).

Slack – viestintätyökalu, jossa esimerkiksi yrityksen sisäistä viestintää on yleensä jaoteltu useisiin eri kanaviin, esimerkiksi ryhmittäin (Slack Technologies, LLC 2025). Yrityksemme pääasiallinen viestintäkanava.

TypeScript – JavaScriptiä hyödyntävä ohjelmointikieli, jossa muuttujat voi tyypittää, eli ilmoittaa minkälaisia arvoja muuttuja voi sisältää (esim. merkkijono, numero, lista, objekti jne.). Toisin kuin

JavaScript, TypeScript tarkistaa minkä tyyppin arvoja muuttujaan asetetaan ja ilmoittaa eroavaisuuksista, mikä saattaa vähentää ohjelmistoon päätyvien virheiden todennäköisyyttä. (Microsoft 2025).

1.1.2 Tietoperusta

Tämän työn tietoperustana käytetyt lähteet koostuvat pääasiassa JavaScriptin, Reactin, Reduxin, HTML:n ja CSS:n virallisista dokumentaatioista. Lähteenä on myös käytetty kirjaa Eloquent Javascript 2nd Edition, joitakin IT-alan tunnettuja keskustelupalstoja ja oppaita kuten Stack Overflow ja GeeksforGeeks, joitakin ulkoisten komponenttikirjastojen dokumentaatioita sekä joitakin yksittäisiä artikkeleita.

2 Lähtötilanteen kuvaus

Kokemukseni ohjelmistokehittäjänä on vielä varsin vaatimatonta. Tietotaitoni perustana toimii toistaiseksi vain yksi Helsingin Yliopiston MOOC-kurssi Java-ohjelmoinnin perusteista, Haaga-Heliassa käymäni ohjelmoinnin kurssit ja toukokuussa 2024 alkanut työharjoittelu, joka jatkuu toistaiseksi voimassa olevana työsuhteena saman työnantajan palveluksessa.

Omaan kuitenkin mielestäni kohtalaisen hyvät ongelmanratkaisutaidot, olen joskus liiankin perusteellinen ja omaksun nopeasti uusia asioita. Toimin myös joissakin asioissa varsin johdonmukaisesti, mutta tässä haluaisin kehittyä niin, että johdonmukaisuus ilmenisi vielä laajemmin työnteosani. Nämä ominaisuudet ovat toistaiseksi palvelleet minua hyvin, sillä tunnen saavani konkreettisia asioita aikaiseksi ja pidän oman työni jälkeä vähäisen kokemukseni huomioon ottaen varsin hyvänä.

2.1 Oman nykyisen työ analysointi

Omat päivittäiset työtehtäväni käsittävät käytännössä ohjelmistossamme esiintyvien virheiden korjaamista, ohjelmiston käyttöliittymän parannuksia, uusien ominaisuuksien mahdollistamista ja oman sekä muiden kollegojen tuottaman koodin lukemista. Pääasiassa tehtäviä seurataan Jirassa, johon kaikki yrityksessä työskentelevät henkilöt voivat ilmoittaa virheitä ja toivottuja toteutettavia ominaisuuksia. Varsin usein kuitenkin esimerkiksi kiireisiä virheilmoituksia ilmoitetaan henkilökohtaisesti Slackissa tai kasvotusten.

Ohjelmistossa esiintyvien virheiden korjaus tarkoittaa kohdallani yleensä virheen paikallistamista käyttöliittymästä ja ohjelmakoodista, vaihtoehdoisen ratkaisun harkitsemista sekä toteuttamista, mahdollisten sivuvaikutusten kartoittamista muualla käyttöliittymässä ja toteutetun ratkaisun testaamista. Yksi omista kehityskohteistani pyrkii ennaltaehkäisemään itse tuottamaani virheellistä ohjelmakoodia. Valitsin kehityskohteen siksi, että JavaScript ei valitettavasti ole erityisen avulias ohjelmointikieli virheiden ennaltaehkäisyn kannalta, jolloin ohjelmoijan vastuu kasvaa ja huolellisuus sekä koodin tuottamisessa, että sen testaamisessa on ensiarvoisen tärkeää (Haverbeke 2015, 140).

Käyttöliittymäparannusten ja uusien ominaisuuksien toteuttamisen käytäntö vaihtelee enemmän tehtävän mukaan, mutta useimmiten aloitan tekemällä tarvittavat muutokset suoraan olemassa olevaan ohjelmakoodiin. Jos kyseessä on täysin uusi ominaisuus, niin luon uuden komponentin ensin. Muutokset tulisi saada mahdollisimman nopeasti testattavaksi ja tuotantoon, jonka jälkeen uutta ohjelmakoodia voi siistiä ja mahdollisesti eriyttää pienempiin komponentteihin.

Kaikki ohjelmakoodiin tehdyt muutokset viedään GitLabiin muutospyyntönä, jonka ainakin kaksi muuta ryhmän jäsentä tulee lukea ja tarkistaa. Omiin työtehtäviini kuuluu siten myös muiden tekemien muutosten läpikäymistä ja testaamista käytännössä.

Käyttöliittymämme on toteutettu JavaScript-ohjelmointikieltä ja sen React-kirjastoa käyttäen, joten niiden taitaminen on ensiarvoisen tärkeää. Lisäksi käyttöliittymän koodi tukeutuu laajalti Redux-tilanhallintakirjastoon, joten myös sen osaaminen on tärkeää. Ohjelmakoodiin perustuvan osaamisen tueksi tarvitaan käyttämiemme työkalujen osaamista, joista tärkeimpiä ovat Visual Studio Code ohjelmointityökalu sekä GitLab versionhallintatyökalu ja siihen liittyvät Git-komennot. Näihin taitoihin olen saanut perustan opintojeni kautta, mutta suurin osa osaamisestani on kertynyt työharjoittelun ja sen jälkeen jatkuneen työsuhteen aikana. Edellä mainittujen teknologioiden osaaminen sekä tuon osaamisen syventäminen ja laajentaminen on kenties tärkein kehityskohteeni. Ohjelmoinnin rikkaus piilee sen tekniikoiden laajuudessa ja monimuotoisuudessa, eikä pidä tyytyä pitämään joi-tain tiettyjä tutumpia keinoja tai ohjelmoinnin hyviä tapoja ainoina oikeina ratkaisuin (Haverbeke 2015, 3).

Oma osaamiseni ei ole mielestäni enää täysin aloittelijan tasolla, sillä kykenen toimimaan itsenäisesti ja löytämään mielekkäitä ja hyödyllisiä tehtäviä oma-aloitteisesti. Tukeudun kuitenkin vielä edelleen mielelläni ryhmävetäjäni ja ryhmäni jäsenten apuun, erityisesti jos tunnen käyttäväni liikaa aikaa jonkin tietyn ongelman ratkaisemiseen. Olen saanut tekemisestäni myös kiitosta, joten koen osaamiseni olevan vaatimuksiin nähden hyvällä tasolla. En koe itseäni kuitenkaan vielä erityisen kokeneeksi ja haluaisin kehittyä erityisesti johdonmukaisuudessa, järjestelmällisyydessä sekä avun pyytämisessä.

2.2 Sidosryhmien esittely

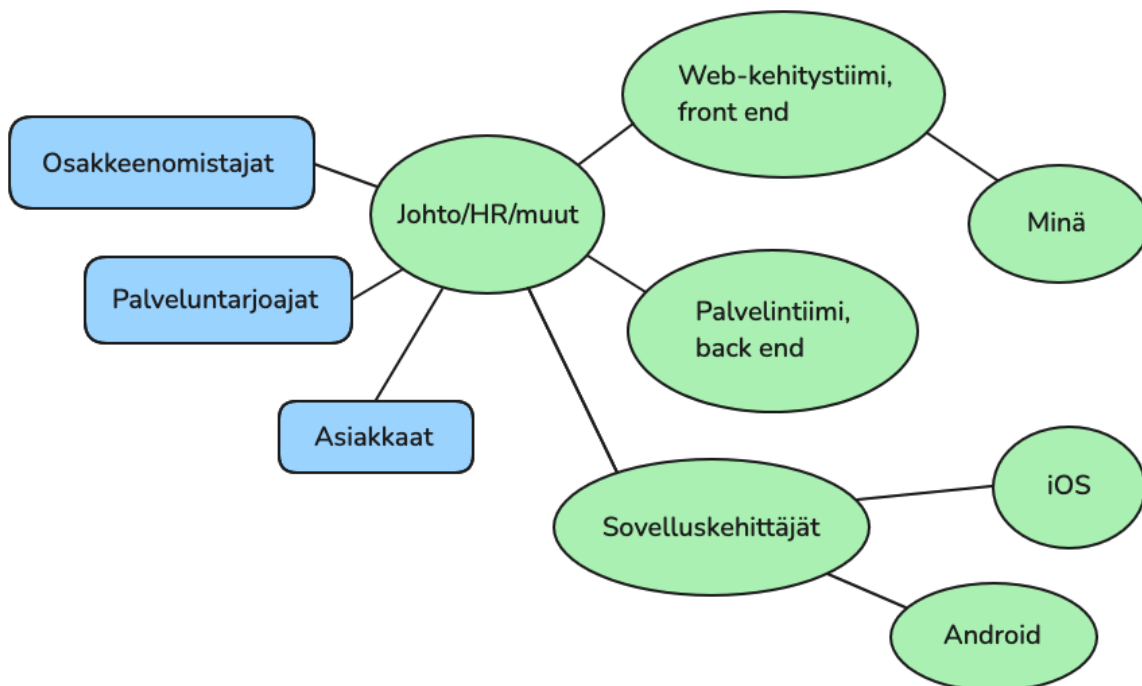
2.2.1 Sisäiset sidosryhmät

Seurantajakson aikana yrityksemme sisäisiin sidosryhmiin kuului johto, web-kehitystiimi tai front end, johon itse kuulun, palvelintiimi tai back end ja sovelluskehittäjät. Johto koostui toimitusjohtajasta, hallituksen puheenjohtajasta, teknologiajohtajasta ja tuotepäälliköstä. Web-kehitystiimiin kuului lisäksi ryhmän vetäjä, kaksi kollegaa ja yksi harjoittelija. Palvelintiimi käsitti ryhmän vetäjän sekä kaksi kehittäjää ja sovelluskehittäjiä työskenteli seurantajaksoni aikana kaksi, yksi Android- ja yksi iOS-kehittäjä. Yrityksen palveluksessa työskenteli myös yksi HR-henkilö sekä yksi henkilö graafisessa suunnittelussa/markkinoinnissa.

2.2.2 Ulkoiset sidosryhmät

Yrityksen ulkoiisiin sidosryhmiin kuuluivat asiakkaat, osakkeenomistajat ja palveluntarjoajat. Tämän työn kannalta ei ole oleellista eritellä asiakkaitamme tarkemmin, mutta voitaneen mainita niihin kuuluvan useita tunnettuja viihdealan tahoja ja televisioituja tapahtumia sekä ohjelmia. Palveluntarjoajiin kuuluvat esimerkiksi Google Cloud ja Firebase, joissa hallinnoidaan ja säilytetään ohjelmistomme tietovarastoja. Työntekijöiden sähköpostitilejä ja yhteisiä tiedostoja hallinnoidaan ja säilytetään myös Googlen palveluissa.

Oman työni kannalta tärkein sidosryhmä oli ja on edelleen asiakkaat. Asiakkaat maksavat välillisesti myös oman palkkani, joten heidän tyytyväisenä pitämisenä on kaikkein tärkeintä. Tämän jälkeen seuraa johto, joiden ohjeiden ja vaatimusten mukaan minun tulee työni suorittaa. Lopulta on myös hyvin tärkeää koordinoita kaikkien eri ryhmien työntekijöiden välillä, miten työtehtäviä kannattaa suorittaa, jotta voimme kaikki yhdessä päästä parempiin lopputuloksiin. Kuvassa 1 on pyritty havainnollistamaan sidosryhmiä ja niiden välisiä yhteyksiä.



Kuva 1. Työpaikan sisäiset ja ulkoiset sidosryhmät. Ulkoiset sinisiä suorakulmioita ja sisäiset vihreitä soikioita.

2.3 Työpaikan vuorovaikutustilanteet

Toimistollamme tapahtuu kohtalaisen paljon vuorovaikutustilanteita toimiston pienen koon takia. Omalla kohdallani vielä erityisen paljon, koska yritän turvautua kollegoitteni apuun mahdollisimman

usein. Yritän tietysti ensin ratkaista ongelmat itse, mutta pyrin välttämään tilanteita, joissa en yksinkertaisesti pääse enää etenemään. Työharjoitteluni aikana en mielestäni pyytänyt apua tarpeeksi nopeasti. Toimistollamme käydään myös päivittäin lyhyt kokous, jossa kaikki ohjelmiston eri osien ylläpidon ja kehityksen parissa työskentelevät henkilöt käyvät läpi tehtävät, joiden parissa he aikovat tuona päivänä työskennellä. Kokouksen muoto ei mielestäni seuraa tarkasti mitään tiettyä ohjelmistokehityksen mallia, mikä sopii itselleni varsin hyvin. Käytän mieluummin aikaani työtehtävieni suorittamiseen. Muut vuorovaikutustilanteet ovat yleensä virheraportointeja toimiston muilta työntekijöiltä, tehtyjen muutosten tai ominaisuuksien käymistä läpi yhdessä tai työhön liittymätöntä keskustelua kollegojen kanssa.

Minulla itselläni ei juuri koskaan muodostu vuorovaikutustilanteita asiakkaisiimme tai muihin ulkoihin sidosryhmiimme. En välttämättä yleensä edes tiedä, kenen toivomaa ominaisuutta kehitän tai kenen ilmoittamaa virhettä korjaan. En ole myöskään joutunut vielä olemaan yhteydessä yhteenkään osakkeenomistajaamme tai palveluntarjoajaamme, mutta saattaa olla, että joudun joskus esimerkiksi ilmoittamaan jossakin käyttämässämme palvelussa ilmenevästä ongelmasta.

En enää koe mitään vuorovaikutustilannetta erityisen haasteelliseksi. Aluksi minun oli hieman vaikea ilmaista itseäni selkeästi päivittäisessä kokouksessa, koska ympäristö oli minulle vielä niin vieras. Lisäksi kokous käydään aina englanniksi, ja vaikka osaan mielestäni kieltä erinomaisesti, niin esimerkiksi ammattisanastoni ei ollut vielä tarpeeksi kehittynyttä. Koen hetkittäin vielä samaa epävarmuutta, myös tilanteissa, joissa minun pitää yrittää selittää kirjoittamaani koodia, mutta tiedän kehittyväni ilmaisussani jatkuvasti.

3 Seurantajakson raportointi viikkoanalyysineen

Seurantajakso käsittää kahdeksan viikkoa, joista jokaiselle on varattu oma lukunsa. Jokainen viikko sisältää merkinnät jokaiselle viikon arkipäivälle ja viikon kokoavalle viikkoanalyysille. Päivien merkinnät on pyritty esittämään ensin päivälle asetettujen tavoitteiden ja tavoitteissa onnistumisen muodossa. Viikkoanalyysissä on pyritty esittelemään tarkemmin joitakin viikon aikana toteutettuja ratkaisuita ja niiden käytäntöä.

3.1 Seurantaviikko 1

– Maanantai 13.1.2025

Arvelin jatkavani erään aiemmin kesken jääneen tehtäväni kanssa, mutta sain yllätyksekseni tehtäväkseni pitää huolta uudesta työharjoitteluun tulleesta opiskelijasta. Tämä kävi toki hyvin yksiin työkalujen ja teknologioiden paremman sisäistämisen kanssa, sillä toista opettaessa asiat selkiintyvät myös niiden opastajalle. Pidän kuitenkin itseäni sen verran kokemattomana, että tehtävä jännitti kohtalaisen paljon.

En onnistunut tavoitteissani erityisen hyvin. Perusasioita osasin mielestäni selittää, mutta kun yritin ratkaista muita haastavampia tehtäviä yhdessä harjoittelijan kanssa, niin en saanut asioita aikaan tai ilmaistua itseäni toivomallani tavalla. Siirryin suosiolla helpompaan tehtävään, eli Twitter-logojen ja linkkien korvaamiseen X-palvelun vastaavilla. Tämä tehtävä oli onneksi helpompi ratkaista ja pääsin sen myötä myös esittelemään harjoittelijalle GitLabia sekä muutospyynnön tekemistä.

– Tiistai 14.1.2025

Tavoitteenani on oppia tänään muutosten järjestelmällisempää testaamista, sillä kollegani tarvitsi apua erään ominaisuuden kanssa, jota olin itse työstänyt ennen joulupyhiä. Muutokset koskivat käyttäjien itse lisättävää tietoa, tai custom dataa, jota voidaan tallentaa ohjelmiston eri osa-alueihin avain arvo pareina. Ne ovat voineet olla esimerkiksi käyttäjän itse määrittämiä linkkejä yleisiin ehtoihin tai visailujen kysymyksiä ja vastauksia. Käyttäjien määrittämät avaimet tuli säilyttää siinä muodossa kuin ne oli annettu, mutta ohjelmistomme eri osissa avaimia muutettiin eri muotoihin.

Onnistuin tavoitteissani hyvin, koska löysin ohjelmistosta kaksi paikkaa, joissa ominaisuus ei toiminut oikein. Raportoin ongelmakohdat kollegalleni, joka tuntui olevan kiitollinen antamastani avusta ja hän lupasi korjata ongelmat pikimmiten. Vaikka olin aiemmin omassa ratkaisussani osannut ratkaista avainten käsittelyn lähes kaikista ohjelmiston osa-alueista, niin päätin myös tutustua kollegani muutospyyntöön, jossa ratkaisu oli kieltämättä yksinkertaisempi.

Sain loppupäiväksi vielä lisää tehtäviä iOS-kehittäjältämme, jota olen auttanut ohjelmistostamme erillisen työkalun kanssa. Kyseistä työkalua käytetään erään Kreikan televisiossa näytettävän tietovisailun kysymysten vastausten näyttämiseen ja käsittelemiseen. Tehtävät olivat onneksi kohtalaisen helppoja ja onnistuin saamaan ne lähes valmiiksi ennen kotiinlähtöäni.

– Keskiviikko 15.1.2025

Tänään tavoitteenani on viimeistellä eiliset muutokset kreikkalaisen tietovisan työkaluun. Osallistujien kuvat eivät vielä näkyneet oikein, ylimääräisiä palvelinpyyntöjä tapahtui edelleen, eräs puto-
tusvalikko ei vielä näyttänyt tai asettanut toivottua oletusmäärää visailuun vastanneista ja tallennusnappi hävisi jostain syystä vaaleaa teemaa käytettäessä. Haluan myös vielä jatkaa hieman Twitter-ikonien ja linkkien parissa, koska olin aiemmin huomannut joitakin kohtia, jotka tuli vielä siistiä ohjelmakoodista.

Tämä päivä oli onnistunut. Sain tehtyä kaikki toivomani muutokset loppuun ja lisäksi sain vielä edelleen testata kollegani tekemiä muutoksia custom datan käsittelyssä. Löysin testatessani kaksi muutoksista johtumatonta virhettä, jotka päätin korjata samalla, koska ne eivät olleet erityisen monimutkaisia. Luin myös uudestaan hänen muutospyyntöään, jossa vaikutuin ratkaisun kauaskantoisuudesta. Aiemmin mainitsemani avainten käsittely ulottuu jo ohjelmiston osiin, mihin käyttäjä ei voi vielä lisätä itse määriteltyä ylimääräistä tietoa. Tämä hyvin todennäköisesti vähentää tehtävän työn määrää tulevaisuudessa.

– Torstai 16.1.2025

Keskiviikkona ennen kotiinlähtöäni sain kuulla, että kreikkalaisen tietovisan työkalun julkaisussa oli virheitä. Virheet johtuivat muutoksistani puuttuvista TypeScript-typityksistä, jotka otan tavoitteekseni lisätä heti aamulla. Sain ne toimimaan oikein, mutta en ollut täysin tyytyväinen lopputulokseen, joten pyysin myös ryhmävetäjäni katsomaan muutokset läpi, että julkaisu varmasti onnistuisi. Aion myös keskittyä testaamiseen, koska olin aiemmin luvannut kollegalleni, että auttaisin häntä testaamaan edelleen custom datan käsittelyyn kohdistuneita muutoksia.

Onnistuin saamaan tietovisan työkalun julkaistavaan kuntoon ja testasin custom data ominaisuutta niin laajalti kuin vain osasin. Kyseistä ominaisuutta voi hyödyntää lähes jokaisessa ohjelmistoon liittyvässä työkalussa, joten testaaminen oli kohtalaisen työlästä. En kuitenkaan tällä kertaa löytänyt enää korjattavaa, mutta kun keskustelin muutoksista ryhmävetäjäni kanssa, niin ilmeni, että hänen mielestään avainten käsittelyä ei kannattaisikaan toteuttaa ohjelmiston osa-alueissa, missä

ominaisuutta ei vielä voi käyttää. Hän esitti varsin päteviä perusteluja kannalleen, joten päädyin olemaan hänen kanssaan aiheesta samaa mieltä.

– Perjantai 17.1.2025

Olosuhteiden myötä tänäänkin tavoitteeni liittyivät testaamiseen ja puhtaampaan ohjelmakoodiin. Olin nimittäin jo jonkin aikaa odottanut yhden back end kehittäjäme palaavan töihin, jotta hän voisi korjata erään kohtalaisen ison ongelman kilpailujen hakemisessa palvelimelta. Jostain syystä kaikki tietyn tilin luomat kilpailut eivät välttämättä löytyneetkään palvelinpyynnöllä, mikä haittasi huomattavasti kreikkalaisen tietovisan työkalun toimintaa. Aion myös opastaa harjoittelijaamme luokkakomponentin muuttamisessa toiminnalliseen komponenttiin.

Ilokseni sain huomata kaikkien kreikkalaisen tietovisan kilpailujen löytyvän myös erillisestä työkalusta pääohjelmiston lisäksi ja aiemmin korjaamani sivutuksen toimivan edelleen. Sivutus ei aiemmin toiminut ulkoisen työkalun kilpailujen listauksessa, koska viimeiseltä sivulta ei pääsyt siirtymään takaisin. Tämän ongelman olin korjannut ennen opinnäytetyön aloittamista. Ilmeisesti harjoittelijamme sai myös tehtävänsä suoritettua ainakin ryhmävetäjäni mukaan, joka auttoi häntä loppupäivän. Ohjelmakoodin ajanmukaistaminen on sinänsä tärkeä tehtävä, jolle ei valitettavasti tunnu aina löytyvän riittävästi aikaa, joten sitä oli hyödyllistä opettaa. Aloitin itse harjoittelijana samankaltaisista tehtävistä.

– Viikkoanalyysi 1

Kaiken kaikkiaan viikko oli monipuolinen. Vaikka en ehtinyt erityisen paljon keskittyä toivomaani tehtävään, eli sovelluksen julkaisuosioon liittyviin käyttöliittymäparannuksiin, niin sain tehdä monipuolisia ja opettavaisia työtehtäviä. Harjoittelijan opastaminen oli hyödyllistä, koska opin paremmin sanoittamaan tekemääni ja syitä siihen, miksi toimin tietyllä tavalla. Samalla kun kerroin ohjelmakoodista ja mitä siinä tapahtuu, niin asiat tuntuivat selkeämmiltä myös itselleni. Toisaalta minulla on vaikeuksia keskittyä työntekoon, jos joku toinen seuraa vieressä.

Ylimääräisten palvelinpyyntöjen yhteydessä ryhmävetäjäni opettama tieto siitä, että komponentin sulkeminen StrictMode-sulkeisiin pakottaa sen osat virkistymään toisen kerran kehitysympäristössä oli mielestäni varsin arvokasta (Meta Platforms, Inc s.a. a). Jatkossa jos saan tehtäväkseni selvittää, että miksi esimerkiksi palvelinpyyntöjä tapahtuu tarpeettoman usein, niin tiedän heti, mistä kannattaa aloittaa.

Ongelmat kreikkalaisen tietovisan työkalun kanssa auttoivat ymmärtämään paremmin TypeScriptin ominaisuuksia. TypeScript on yleistymässä, ja sitä olisi hyödyllistä opetella lisää muutenkin. Omassa ohjelmistossamme sitä ei juuri tarvita, sillä lähes kaikki ohjelmakoodi on toteutettu JavaScriptillä ja Reactilla. Kyseinen alusta on erään yrityksen ulkopuolisen konsultin kehittämä. Tämän tehtävän myötä pääsin myös tällä viikolla perehtymään tietoperustaani, tarkemmin React-dokumentointiin. Listan käyttäminen tilamuuttujana tuli sitä kautta tutuksi, vaikka se ei lopulliseen toteutukseeni päätynytkään. Opin tarkemmin, että listana olevaa tilamuuttujaa tulee muuttaa asettamalla sille arvoksi uuden listan (Meta Platforms, Inc s.a. b). Lisäksi sain pudotusvalikon näyttämään sekä vaihtamaan oikean arvon tutustumalla option elementin W3Schoolsin HTML-dokumentointiin (Refsnes Data 2025b). Tallennusnapin katoaminen johtui jostain syystä tyylitiedostoon jääneestä asetuksesta, jossa napin taustaväriksi asetettiin valkoinen vaalean teeman ollessa aktiivinen.

Virheet, jotka löysin custom datan muutosten testaamisen aikana oli helppo korjata lisäämällä varmistus poistettujen tai valmiiksi puuttuvien arvojen varalta. Ristiriita ryhmävetäjäni ja toisen kollegani näkemysten välillä kyseisen tiedon käsittelyn toteuttamisesta etukäteen paikoissa, jossa koko ominaisuutta ei voi vielä käyttää oli varsin herkullinen. Ymmärrän kollegani näkemystä siinä mielessä, että toteutus saattaa vähentää työmäärää tulevaisuudessa, mutta olen silti päätenyt olemaan samaa mieltä ryhmävetäjäni kanssa. Ohjelmakoodi saattaa johtaa harhaan vielä asiaan perehtymättömiä, jos ohjelmakoodissa vaikuttaa siltä, että ominaisuus olisikin käytettävissä jossain, missä näin ei todellisuudessa olekaan. Se saattaakin siis lisätä työmäärää mahdollisten ongelmatilanteiden ratkaisemisen kohdalla. Lisäksi ohjelmakoodiin päätyy roikkuvaa ohjelmakoodia, joka ei pahimmassa tapauksessa tule koskaan käyttöön ja palvelinpyyntöihin saattaa myös päätyä ylimääräistä tietoa, joka joudutaan kuitenkin palvelinpuolella käsittelemään, vaikka sille ei ole toimittava vastinetta.

Kuvien sivulle lataamisen yhteydessä opin tietoperustaani tukeutuen lisää HTML:n img-elementin ominaisuuksista. Tätä kautta kävi ilmi, että kuvan mitat saa varsin helposti selville elementin onLoad-tapahtumalla (GeeksforGeeks 2024a). Haluaisin kovasti saada toteutukseni nopeasti valmiiksi ja muiden nähtäville, koska mielestäni ratkaisuni oli varsin yksinkertainen. Tätä tehtävää tehdessäni olen nimittäin nähnyt erään kollegani ratkaisun, missä ilmeisesti kuvan mitat haetaan käyttäen kohtalaisen monimutkaista funktiota, jossa kuvasta tehdään niin sanottu kangas, jonka kautta saadaan kankaan mitat. Lisäksi ajattelin tehdä ohjelmakoodiin tarkistuksen joka kuvan kohdalle, että jos ne eivät vastaa määriteltyjä kuvasuhteen raja-arvoja, kuten esimerkiksi 9/16, niin kuvia ei voi ladata palvelimelle, ennen kuin ne ovat valmiiksi oikeassa kuvasuhteessa, tai ne on rajattu käyttäen toista hieman muokattua komponenttia. Tällä hetkellä kuvan rajaaminen johtaa suoraan

kuvan lataamiseen, eli joudun hieman soveltamaan komponentin toimintaa siten, että kuva rajataan, mutta rajaaminen ei johda suoraan kuvan lataamiseen palvelimelle.

Jo tällä ensimmäisellä seurantaviikolla olen mielestäni kehittynyt paljon eräässä tärkeimmistä tavoitteistani, eli avun pyytämisessä. Aina kun minulla oli siihen mahdollisuus tai tarve, niin kysyin rohkeasti apua. On mielestäni itsestään selvää, että vaikka parhaiten oppii itse tekemällä, niin optimaalisimpaan tulokseen pääsee vain ottamalla oppia kokeneempien tekijöiden neuvoista. Toinen erittäin tärkeä oivallus liittyy kuvaelementin ominaisuuksien avaamiseen. Ei kannata olettaa, että kokeneemmat kollegat ovat ottaneet jo kaikesta selvää, vaan ensin kannattaa itse yrittää tutkia aihetta, eikä vain päätyä mukailemaan aikaisempia ratkaisuita.

3.2 Seurantaviikko 2

– Maanantai 20.1.2025

Tänään tavoitteeni on edistyä koodin paremmassa ymmärtämisessä ja tuottamisessa kuvien palvelimelle lataamisen kautta. Lisäksi aion saada vietyä onnistuneesti GitLabiin saavutettavuuden tyyliasetuksiin tekemäni muutokset. Perjantaina vieminen epäonnistui jostakin syystä. Aion myös tavoitella tehtävieni nopeampaa valmistumista pyytämällä apua ryhmävetäjältäni.

Onnistuin suunnittelemani tavoitteiden osalta varsin hyvin, sillä sain edistettyä kuvien lataamista siihen pisteeseen, että voin laajentaa ratkaisuni Google Playn osiosta Apple Storeen ladattaviin mainoskuviin. Sain siis jonkin verran apua, mutta uskoisin, että olisin löytänyt ratkaisut itsekini. Koska olen kuitenkin työskennellyt tämän tehtävän parissa jo kohtalaisen kauan, niin koen pientä painetta saada parannukset valmiiksi. GitLabin ongelman sain ratkaistua itse etsimällä vastausta verkosta.

– Tiistai 21.1.2025

Päätin yrittää keskittyä tehtävissäni tänään virheettömämmän koodin tuottamiseen. Haverbeken (2015, 3) mukaan ohjelmien hallinnassa pitäminen on ohjelmoinnin pääasiallinen ongelma. Siksi mielestäni ohjelmien oikea toiminta on ensiarvoisen tärkeää.

Koska kuvien ohjelmistoon ja palvelimelle lataamiseen käytettävä komponentti on varsin laajalti käytössä ohjelmistomme eri osa-alueissa, niin päätin eriyttää tekemäni muutokset kahteen uuteen komponenttiin. Työstämäni parannukset tulevat siis voimaan vain sovelluksen julkaisuun eri sovelluskaupoissa kuuluvaan ohjelmistomme osaan.

Lisäksi löysin mahdollisesti yhden uuden ongelman samasta osasta, joka ei voi johtua omista muutoksistani, koska löysin sen tuotantoympäristössä. Nyt tiedän etsiä kyseistä virhettä ohjelmakoodissa työstäessäni omaa koodiani ja mahdollisesti myös korjata sen. Valitettavasti jouduin myös korjaamaan erään ongelman vanhasta koodistani, joka liittyi saavutettavuuden tyyliasetukseen. Se ei tosin ollut vielä ehtinyt edes testiympäristöön, mutta se tulisi saattaa vähintään sinne heti seuraavana aamuna, koska yksi iOS-kehittäjäamme joutuu tuolloin esittelemään ominaisuutta johdolle. Onnistuin mielestäni tavoitteissani hyvin. Ratkaisuni eristäminen kuvien lataamisessa vähentää huomattavasti ongelmien esiintymisen todennäköisyyttä ohjelmiston muissa osa-alueissa. Löysin mahdollisesti uuden ongelman, jonka voin korjata, sekä tein korjauksen toiseen kriittisessä vaiheessa. Harmittaa vain, että en ollut aiemmin huomannut tuota viimeistä ongelmaa.

– Keskiviikko 22.1.2025

Päätin jatkaa tänään virheettömämmän koodin parissa, sillä kävimme aamulla läpi yhdessä ryhmävetäjäni kanssa hieman kerääntyneitä muutospyyntöjä. Tässä oli mielestäni oiva tilaisuus oppia kokeneemmalta, että miten välttää ongelmakohtia omassa koodissani. Sain myös toteuttaakseni kreikkalaisen asiakkaan toivomia muutoksia tietokilpailun työkaluun ja erityisesti yksi tehtävä sivusi aihetta hyvin.

Aamun yhteisen koodikatselmuksen ansiosta opin lukemaan paremmin koodia, kun ryhmävetäjäni näytti koodin etenemistä eräässä artikkelin tallentamiseen liittyvässä funktiossa. Artikkelit ovat olennainen osa ohjelmistomme kautta luotuja sovelluksia, sillä suurin osa sovellusten sisällöistä on näitä artikkeleita. Sain häneltä myös joitakin hyviä vinkkejä koodin siistinä pitämiseen, mikä mielestäni vähentää mahdollisten virheiden esiintymistä luettavamman koodin ansiosta. Onnistuin myös varsin vaivattomasti etsimään ongelman tietovisatyökalussa ja lisäämään ongelmakohtaan varmistuksen, jota ilman työkalun profiilisivu kaatui kokonaan.

– Torstai 23.1.2025

Olin ajatellut keskittyä tänään vaihteeksi ohjelmakoodin parempaan ymmärtämiseen ja tuottamiseen, mutta päädyin kuitenkin ratkomaan erästä kuvien lataamiseen liittyvää ongelmaa. Minulla oli tietty aavistus virheen syystä, mutta vastaanotin mielelläni ryhmävetäjäni tarjouksen avusta. Hänen ansiostaan saimme ongelman nopeammin paikallistettua ja korjattua, minkä lisäksi sain taas seurata ja omaksua kokeneemman kollegan toimintatapoja.

Sain keskittyä jonkin aikaa myös parannuksiin, joita back end ryhmävetäjä oli pyytänyt ryhmälämmemme. Otin tehtävän itselleni, koska minulla oli jo valmiina joitakin parannuksia samaan ohjelmiston osa-alueeseen, mutta ne olivat jääneet roikkumaan, koska ominaisuus ei ole vielä tuotannossa. Näihinkin tehtäviin liittyi paljon virheiden eliminointia, sillä koko ominaisuus on varsin varhaisessa vaiheessa.

Vaikka ryhmävetäjäni auttoi ratkaisun kanssa, niin ongelma kuvien lataamisessa oli juuri siellä, missä aavistelin sen olevan. Ryhmävetäjäni mainitsi myös tästä kiittäessäni häntä avusta ongelman ratkaisemisen kanssa, joten tämän ongelman osalta tunsin onnistuneeni silti varsin kiitettävästi. Samoin päivän toisessa tehtävässäni onnistuin ratkaisemaan useita ongelmia, joiden perusteella voin hyvin todennäköisesti saattaa ominaisuuteen liittyvät parannukset loppuun.

– Perjantai 24.1.2025

Päätin ehdottaa ryhmävetäjälleni, että toisimme torstaina ratkaisemamme ongelmaa koskevat muutokset tuotantoon. Perjantaisin tätä ei yleensä tehdä, koska jos muutokset tuottavat sivuvaikutuksia ohjelmiston toimintaan, niin kukaan ei ole viikonloppuisin korjaamassa niitä. En kuitenkaan ajatellut keskittyä tänään sen enempää virheettömämmän koodin edistämiseen, sillä korjaus oli käytännössä vain pieni muutos yhteen riviin ohjelmakoodissa. Ajattelin sen sijaan pyrkiä edistämään koodin ymmärrystä ja teknologioiden tehokkaampaa käyttöä perehtymällä syvemmin avoimiin muutospyyntöihin.

Näissä pyynnöissä oli tänäänkin joitakin minulle tuntemattomampia funktioita, joihin lähdin tutustumaan paremmin tietoperustani kautta. Mielestäni tämä ei eroa juurikaan esimerkiksi vieraan kielen opiskelusta, jolloin yleensä tuntematon sana pyritään kääntämään omalle kielelle. Onnistuin siis syventämään tietämystäni kahden eri listan käyttöön liittyvän funktion toiminnasta ja käyttämisestä.

– Viikkoanalyysi 2

Suurin osa viikosta kului edelleen sovelluksen julkaisun parannusten parissa, erityisesti kuvien lataamisessa ohjelmistoon ja palvelimelle. Tehtävä on ollut varsin antoisa, mutta haluaisin tosiaan saada sen pian valmiiksi. Tätä valmistumista on toki hieman hidastanut muut kiireellisemmät tehtävät, joita olen saanut tai ottanut hoitaakseni.

Alkuviikon GitLabiin liittyvä ongelma ratkesi tutustumalla Stack Overflown keskustelupalstaan. Palstan korkeimmassa vastauksessa ehdotettiin useita "git config pull.ff" lauseita, joista tässä tapauksessa käytin lausetta "git config pull.ff only" (Stack Overflow 2020). Varsin usein kohtaamani

ongelman ratkaisemiseen olen aina ennen pyytänyt ryhmävetäjäni apua, mutta hän oli kiireinen, niin päätin yrittää itse. Aikaisemmin en uskaltanut, koska pelkäsin hävittäväni tekemiäni muutoksia tai mahdollisesti sekoittavani "main" haaramme. Ajan kanssa Git-komennot ja logiikka ovat käyneet tutummaksi, joten osasin pitää muutokseni ja haaran turvassa sekä vietyä muutokseni turvallisesti eteenpäin.

Kreikassa näytettävän tietovisailun työkalun ongelma oli onneksi aika yksinkertainen. Tietovisa näytetään jo tulevana viikonloppuna, joten kaikki siihen liittyvät työtehtävät ovat olleet jo jonkin aikaa tärkein prioriteettini. Työkalussa käsitellään visailuun vastanneiden yleisön profiileita, joihin liitetty osallistujien tietoja ja profiilikuva. Ongelmana oli varmistuksen puuttuminen osallistujan kaupungin tiedosta, mutta se sattumalta liittyi myös profiilikuviin. Nimittäin jos osallistujalta puuttui kuva, niin tietoja ei voinut saada auki, koska se vaati kuvan klikkaamista. Työkalussa oli mahdollisuus näyttää kaikkien profiilien tiedot jo listanäkymässä erään valintaruudun valitsemalla, mutta jos tämä valintaruutu oli valittuna, mutta kuvaa ei löytynyt, niin ohjelma kaatui. Puuttuvaan kaupunkitietoon oli helppo lisätä varmistus ja puuttuvan kuvan tilalle toin oletuskuvan, jota pystyi klikkaamaan.

Toisaalla löytämäni virhe pääohjelmiston sovellusten julkaisuun liitettävien kuvien lataamisessa oli oikeastaan kohtalaisen vakava. Käyttäjän näkökulmasta kuvat hävisivät kokonaan, vaikka todellisuudessa kyse oli virheestä palvelinvastauksen lähettämän tiedon käsittelyssä. Kuvat siis tallentuivat palvelimelle, mutta käyttäjällä ei voinut olla siitä mitään tietoa, paitsi tarkistamalla selaimen tarkastelunäkymän verkkoliikenteestä palvelimen vastaus, josta kuva olisi löytynyt. Tutkimme siis ongelmaa yhdessä ryhmävetäjäni kanssa, mutta aavistukseni osui täysin oikeaan. Työskennellessämme ongelman parissa silmiini osui lokien kautta joitakin objektien avaimia, jotka olivat esimerkiksi muodossa "somePhone_12screenshot". Tästä päätin, että ongelma voisi liittyä aiempiin muutoksiin custom datan käsittelyssä ja erääseen tiettyyn funktioon, joka kääntää avaimia tarvittavaan muotoon. Löysimmekin kyseisestä funktiosta säännöllisen lausekkeen, joka poisti kirjainten edessä olevan alaviivan ja muutti kirjaimen isoksi, mutta lauseke ei osannut ottaa huomioon numeroita (kuva 2).

Overview 0 Commits 1 Pipelines 0 Changes 1

Compare main and latest version

client/src/utills/api.js

```

@@ -171,7 +171,8 @@ export const selectiveCamelizeKeys = (data, preservedProps = []) => {
 171 171     return camelizedCache.get(str)
 172 172 }
 173 173
 174 - const camelized = str.replace(/_([a-z])/g, (_, letter) =>
 174 + /** APPs => PUBLISH APPS entities have numbers on some object properties i.e.: screenshotsIphone65 */
 175 + const camelized = str.replace(/_([a-z0-9])/g, (_, letter) =>
 175 176     letter.toUpperCase()
 176 177 )
 177 178     camelizedCache.set(str, camelized)

```

Kuva 2. Korjaus säännölliseen lausekkeeseen, joka ottaa korjauksen jälkeen huomioon myös numerot. Kuva GitLabin muutospyynnöstä.

Saman antoisan tehtävän ratkaisuihin kuuluu muitakin. Käyttöliittymän kautta on siis nyt mahdollista lisätä useita kuvia kerralla ja kuvia lisättyään käyttäjällä on mahdollisuus ladata nämä kuvat palvelimelle tai lisätä muitakin kuvia. Tekemäni muutoksien ansiosta mahdollisuus lisätä muita kuvia ei enää rajoitu vain yhteen kuvaan, eivätkä uusien tai uuden kuvan lisääminen poista jo ladattuja. Yhden kuvan lisäämisen rajauksen poistaminen selvisi tutkimalla Browse-painiketta painettaessa kutsuttavan funktion palauttavan input-elementin ominaisuuksia Mozillan dokumentaatiosta, jonka perusteella opin lisätä kyseiselle elementille multiple totuusarvon, joka mahdollistaa useamman tiedoston valinnan (Mozilla s.a. b). Jostain syystä uusia kuvia lisättäessä jo olemassa olevat kuvat oli unohdettu, jolloin uudet kuvat korvaavat vanhat. Tämä oli helppo korjata käyttäen levityssyntaksia, jolloin listan vanhat arvot säilytetään tuodessa uusia arvoja listaan (Mozilla s.a. c.). Lisäksi perjantaina pyytäessäni ryhmävetäjäni katsomaan mielestäni varsin paisunutta ohjelmakoodiani, niin ilmeni, että myöskään hänelle kuvasuhteiden avaaminen edellisessä viikkoanalyysissäni esittelemäni kautta ei ollut entuudestaan tuttua. Mieltäni lämmitti erityisesti hänen kommenttinsa, jossa hän totesi, ettei mahdollisia virheitä muualla ohjelmakoodissa kannata pelätä, jos olemassa olevan komponentin toimintaa parannetaan. Virheisiin täytyy tietysti osata varautua ja niitä tulee ennaltaehkäistä mahdollisimman tehokkaasti. Luulen kyllä, että siihen tarvitaan vielä jonkin verran työtä, mutta ilokseni sovimme jatkavamme aiheen parissa yhdessä ensi viikolla.

Keskiviikkona päätin hieman edistää back end ryhmävetäjän pyytämiä muutoksia, joista osa oli kohtuullisen helppo toteuttaa ja joitakin virheitä helppo ratkaista, mutta joista yksi vaati hieman enemmän työtä. Ohjelmistoomme on tulossa ominaisuus, jossa käyttäjien on mahdollista luoda omia aliverkkotunnuksia ja linkkejä. Tällä hetkellä käyttäjä voi luoda korkeintaan kaksi aliverkkotunnusta, mutta niitä ei voi muokata mitenkään. Back end ryhmävetäjä toivoi mahdollisuutta muokata

aliverkkotunnuksen nimeä ja loppupäätettä, mitä varten päätin luoda uuden komponentin. Sain kuitenkin kaipaamiani lisäohjeita ja siirryin muihin kiireellisempiin tehtäviin, mutta sitä ennen sain ryhmävetäjältäni taas mietittävää uusiin komponentteihin liittyen, koska aliverkkotunnuksen luomiseen käytettävä komponentti tarjoaa kohtalaisen vaivattomilla muutoksilla mahdollisuuden myös muokkaamiseen.

Loppuviikon koodikatselmuksen listaan liittyvien metodien toimintaa etsin tietoperustani kautta ja kysymällä ryhmävetäjältäni. Ensinnäkin minulle uusi tuttavuus `Array.of`-metodi tuottaa uuden listan sille argumentteina annetuista arvoista, riippumatta niiden määrästä tai tyypeistä (Mozilla s.a. d). Muistan tarvinneeni vastaavaa ratkaisua aiemmin tällä viikolla, mutta päätyneeni tuolloin johonkin muuhun keinoon. Aion kokeilla kyseistä metodia heti, kun sille tulee tilaisuus. Toinen minulle tumpi listaan liittyvä `flat`-metodi latistaa rakenteellisen listan, mutta en ollut ennen ryhmävetäjäni tekemää muutospyyntöä nähnyt sille parametrinä annettavaa numeroarvoa, joka määrittelee, kuinka monta tasoa latistetaan.

Viikon tärkein oppini liittyy uusien komponenttien luomiseen. Edellisen viikon päiväkirjamerkinnoissä olen maininnut uusien komponenttien luomisesta ja niiden virheiden vähentävästä vaikutuksesta, mutta totuudella on toinenkin puoli. Jatkuvasti uusien komponenttien luonti paisuttaa ohjelmakoodin määrää, mikä johtaa väistämättä muihin ongelmiin ja saattaa jopa edesauttaa uusien ongelmien ilmenemistä. Tavoitteena on tietysti kaiken kaikkiaan ohjelmakoodin optimointi mahdollisuuksien mukaan ja uusien komponenttien luonti näiden mahdollisuuksien puitteissa. Yritän siis jatkossa virheitä pelkäämättä parantaa olemassa olevaa koodia ja mahdollisimman perusteellisesti selvittää tekemieni muutosten seuraukset.

3.3 Seurantaviikko 3

– Maanantai 27.1.2025

Aion tänään aloittaa kuvien lataamiseen tekemieni muutosten testaamisella muualla ohjelmistossa. Niin kuin olen aiemmin maininnut, niin kyseinen komponentti on varsin laajalti käytössä ohjelmistomme eri osissa. Yllättävän harvoin kyseessä on kuitenkin useamman kuvan lataamisesta kerralla, mihin suurin osa muutoksistani on tehty. Siistimme myös vähän koodiani yhdessä ryhmävetäjäni kanssa.

Koska useamman kuvan lataamiseen käytettävä komponentti oli loppujen lopuksi yllättävän vähässä käytössä, niin toin tekemäni muutokset takaisin alkuperäisiin komponentteihin ja poistin tekemäni uudet. Tämä vähentää turhan koodin määrää ja koodin sekavuutta, joten olen tyytyväinen ratkaisuuni. Saimme muutenkin koodia siistimmäksi, ja koen oppineeni lisää koodin kirjoittamisen hyvistä käytännöistä.

– Tiistai 28.1.2025

Joudun taas tänään jättämään kuvien lataamisen muutokset vähemmälle huomiolle, sillä sain joitakin kiireellisempiä tehtäviä kreikkalaisen tietovisan työkalun parissa. Tavoitteeni ohjelmakoodin paremmasta ymmärtämisestä ei tästä muuttunut, sillä saan syventää tietämystäni esimerkiksi JavaScriptin numeroiden pyöristämisessä ja edelleen listojen käsittelyssä.

Tavoitteissani onnistuin hyvin, koska opin ymmärtämään paremmin lukujen pyöristämistä ja reduce-metodin käyttämistä listoissa. Kyseinen metodi on ollut pitkään listallani yksittäisistä asioista, joita haluaisin oppia käyttämään paremmin. Vielä sen käytössä on oppimista, mutta ymmärsin paremmin niin sanotun kerääjän eli accumulatorin toimintaa.

– Keskiviikko 29.1.2025

Toivon tänään saavani vihdoinkin kuvien lataamisen muutokset valmiiksi ja päätin siksi pyytää apua ryhmävetäjältäni. Olisin jo kauan sitten päässyt siirtymään muihin tehtäviin ilman erästä varsin hankalaa ongelmaa, joka johtaa komponentin jatkuvaan virkistämiseen. Ongelma osoittautui kuitenkin sen verran hankalaksi, että ryhmävetäjänikään ei saanut sitä ratkaistuksi. Siksi päätimme molemmat siirtyä työstämään kiireisempiä tehtäviä.

En kokenut onnistuneeni tavoitteissani, koska en saanut edistettyä alkuperäistä tehtävääni, mutta sain onneksi täydentää tietotaitojani muiden tehtävien parissa. Päätin esimerkiksi siistiä kreikkalaisen tietovisan työkalun tilastonäkymän käyttämään useiden div-elementtien sijaan table-elementtiä, jonka pitäisi auttaa pitämään ohjelmakoodi ja ulkoasu siistimpänä.

– Torstai 30.1.2025

Jatkan tänään kreikkalaisen tietovisailun työkalun parannusten parissa. Tavoitteenani on saada parannukset tänään valmiiksi, koska ymmärtääkseni visailu näytetään viikoittain lauantaisin. Koen myös lievää stressiä, sillä minulle on kerääntynyt ehkä jo liikaakin tehtäviä, ja osa on jäänyt roikkumaan keskeneräisinä. Yhden niistä laitoin suoraan Jiraan, jotta joku ryhmäni jäsenistä voisi ottaa sen hoitaakseen. Kyseessä oli tietyn tilin käyttäjien haku tietyillä parametreilla, ja sain tehtävän tiedot tänään back end kehittäjältä, joka oli toteuttanut palvelinpyynnön haulle.

Jouduin tukeutumaan paljolti tietoperustaani saadakseni tietovisailun vastausten tulokset pyöristymään oikein. Viikko sitten lauantai-iltana seuratessamme visailua toimistolla back end ryhmävetäjän ja iOS-kehittäjän kanssa huomasimme, että tulosten yhteenlaskettu summa pyöristyi joskus väärin. Kokonaistulos saattoi joskus jäädä 99 %:iin tai nousta 101 %:iin. Löysin erään ratkaisun Stack Overflowsta, jonka toteutin onnistuneesti. Sain myös näkymän siistimisen valmiiksi, joten olen tyytyväinen päivään, vaikka olisin halunnut saada muutokset nopeammin valmiiksi.

– Perjantai 31.1.2025

Tänään yritän tavoitella parempaa ohjelmakoodin ymmärtämistä ja sitä kautta tehokkaampaa koodin kirjoittamista. Ajattelin siksi antaa eilen tekemäni funktion tekoälyn tarkasteltavaksi, jotta voisin sitä kautta kirjoittaa funktion luettavammaksi ja siistimmäksi. Olen joskus seurannut vierestä, kun ryhmävetäjäni antaa omaa koodiaan tekoälyn käsiteltäväksi. Tätä esimerkkiä seuraten kohteeksi päätyi Googlen Gemini, jonka ehdotus vaikutti lupaavalta. Oman funktioni rivimäärä kutistui noin puoleen alkuperäisestä. En aikonut kuluttaa tähän kuitenkaan koko päivää, joten aion myös lähteä toteuttamaan eilen Jiraan luomaani tehtävää tilin käyttäjien hakemisesta.

Olin toivonut tekoälyn ratkaisun olevan toimiva, mutta testatessani sitä käytännössä huomasin, että prosentit pyöristyivät silti väärin. Yritin hetken aikaa ymmärtää miksi, mutta koska kreikkalainen visailu käydään jo huomenna, niin päätin antaa oman toimivan ratkaisuni jäädä tuotantoon ja siirtyä toiseen tehtävääni, eli tekemään hakuominaisuutta tilin käyttäjille.

Yleensä olen ensin tehnyt tarvittavat muutokset jo olemassa olevaan komponenttiin tai ohjelmakoodiin ja sitten siirtänyt ne tarvittaessa uuteen komponenttiin, mutta päätin tällä kertaa kirjoittaa muutokset jo valmiiksi uuteen komponenttiin. Minulle tämä ratkaisu tuntui tässä tapauksessa luontevalta ja sain mielestäni näin tehtyä selkeämpää ja tehokkaampaa koodia. Pyysin myös rohkeasti apua ryhmävetäjältäni, mikä tietysti edesauttoi tehtävän nopeampaa toteutumista.

– Viikkoanalyysi 3

Tällä viikolla en mielestäni saanut tarpeeksi aikaiseksi, mutta joitakin onnistumisia viikkoon onneksi mahtui. Olisin esimerkiksi halunnut saada sovelluksen julkaisuun tekemäni parannukset sekä back end ryhmävetäjän pyytämät ominaisuudet linkkien luomisessa ja aliverkkotunnusten muokkaamisessa vähintään testiympäristöön kokeiltavaksi. Minun pitää kuitenkin olla itselleni myös jonkin verran armollinen, sillä sain viikon aikana hoitaakseni kiireisempiä tehtäviä, kuten esimerkiksi kreikkalaisen tietovisailun työkalun parannukset ja käyttäjän etsiminen tietyillä hakuarvoilla tilin käyttäjien listauksesta.

Sovelluksen julkaisun muutokset ja erityisesti sovelluskaupoille toimitettavien kuvakaappausten kuvien lataamisen parannukset ovat lähes valmiit. Ainoana ongelmana on kuvien lataamiseen käytettävän komponentin epävakaus. Yksittäisen kuvan kohdalle lisäämäni tarkistus kuvasuhteiden tai kuvan mittojen sopivuudesta on toimiva, mutta tiedon välittäminen jonkin tietyn kuvan tai kuvien epäsojivuudesta emokomponenttiin tuntuu ylitsepääsemättömältä esteeltä. Olen yrittänyt ryhmävetäjäni kanssa välittää tiedon emokomponenttiin hyväksytyistä ja/tai epäkelvoista kuvista niin listana, kuin objektina, mutta kaikki yrittämämme ratkaisut johtivat lapsikomponentin jatkuvaan virkistämiseen ja ohjelman jumiutumiseen tämän seurauksena. Emme valitettavasti ehtineet palata ongelman pariin alkuviikon jälkeen, mutta minulla on kuitenkin idea ratkaisusta, jota toivon pääseväni kokeilemaan ensi viikolla.

Kreikkalaisen tietovisailun tehtävät tällä viikolla olivat onneksi oppimisen kannalta hedelmällisempiä. Pyydetty muutokset koskivat vain vastausten lukumäärän poistamista näkymästä, sillä asiakkaalle riittivät vain prosentit ja prosenttien pyöristämiset oikein. Kävin kuitenkin läpi Mozillan table-elementin dokumentaatiota toteutustani varten, sillä vaikka sitä ei minulta erikseen pyydetty, niin koin useiden div-elementtien käyttämisen varsin kömpelönä, joten päätin toteuttaa tietojen näyttämisen käyttämällä table-elementtiä. Mozillan dokumentaatiosta minun täytyi lähinnä tarkistaa, miten table-elementti rakentuu, eli mitä elementtejä kuuluu sen otsakkeeseen ja mitä itse tauluun (Mozilla s.a. e). Prosenttien väärin pyöristymiseen löysin apua Stack Overflow keskustelupalstalta, josta löytämässäni ratkaisussa alkuperäisiin pyöristämättömiin kokonaisarvoihin lisättiin 1 niihin arvoihin, joissa desimaaliosuus oli suurin, kunnes päästin täyteen sataan prosenttiin (Stack Overflow 2019). Huomasin eron vasta myöhemmin, mutta jostain syystä itse toteutin ratkaisun niin, että arvoihin, jotka olivat lähimpänä seuraavaa kokonaisarvoa, lisättiin puuttuva desimaaliarvo. Ehkä itselleni lukuja oli helpompi käsitellä näin päin, mutta toteutus oli kuitenkin lähes sama.

Pääsin ilokseni saman tehtävän parissa tutustumaan ohjelmakoodissamme usein kohtaamaani listoja käsittelevään reduce-metodiin. Kyseinen metodi on näyttänyt minulle usein jopa pelottavan monimutkaiselta, mutta nyt opin ymmärtämään sitä vähän paremmin. Metodin kerääjä, eli accumulator käyttäytyy mielestäni hyvin samalla tavalla kuin esimerkiksi for-lausekkeissa käytettävä erillinen muuttuja, jota kasvatetaan lauseketta suorittaessa. Soveltamissani ratkaisuiissa käytin sitä juuri tähän tapaan, jossa sen alkuperäinen arvo oli 0 ja siihen lisättiin listan numeroarvoja ja lopuksi palautettiin koko listan sisältämien arvojen summa. Tämä oli tietysti ehkä yksinkertaisin metodin käyttötapaus, mutta sen pohjalta toivon osaavani lähteä jatkossa yrittämään monimutkaisempia ratkaisuita. (Mozilla s.a. f).

Jouduin edelleen palaamaan saman tehtävän pariin uudestaan loppuviikosta, sillä työkalun rakentaminen ja julkaisu ei onnistunut taaskaan kahden puuttuvan tyyppityksen takia. Olin tästä varsin

harmissani, koska olin aiemmin luvannut kiinnittää niihin paremmin huomiota. Puolustukseksi kyse ei tällä kertaa ollut sittenkään suoraan tyyppityksistä, vaan valmiiden metodien hyväksymistä parametrien tyypeistä. Aiemmin päiväkirjamerkinnoissani mainitsemani kokeilu tekoälyn kanssa auttoi ensimmäisen ongelman ratkaisemisessa. Valitettavasti en ollut säästänyt keskusteluani Geminin kanssa, koska sen tuottama ratkaisu ei toiminut, mutta muistin sen käyttäneen listan järjestämiseen sort-metodia aluksi käyttämäni toSorted-metodin sijaan, joka tuottaa uuden listan (Refsnes Data 2025c). Toiseen ongelmista sain apua ryhmävetäjältäni, jonka kanssa tutustuimme Stack Overflow keskustelupalstaan objektien kopioimisesta. Palstan ylimmässä vastauksessa neuvottiin käyttämään rakenteellista kloonamista structuredClone-metodia käyttäen (Stack Overflow 2022). Kyseinen metodi luo alkuperäisestä objektista niin sanotun syvän kopion, jonka ominaisuudet eivät jaa samoja viitteitä alkuperäisen kanssa. Tällä voidaan varmistua siitä, että kummankaan muuttuessa toinen objekti ei muutu samalla. (Mozilla s.a. g). Omassa ratkaisussani olin luonut listasta objektin käyttämällä levityssyntaksia (Mozilla s.a. c), mutta näiden korjausten ansiosta rakentaminen ja julkaisu meni läpi ongelmitta.

Omasta mielestäni viikon parhain saavutukseni on toteuttamani uusi komponentti tilin käyttäjien etsimiseen. Vaikka prosenttien pyöristäminen viimeinkin oikein tuntuu hyvältä, erityisesti siksi, että tekoäly ei siinä onnistunut, niin kirjoittamani funktio oli mielestäni koodin kannalta heikkolaatuista, vaikkakin se toimii oikein. Nyt luomani komponentti on hyvin pelkistetty, sillä sille ei tarvitse välittää yhtään ominaisuutta emokomponentista, vaan se saa kaiken tarvitsemansa Redux-tilanhallintakirjaston kautta. Juuri tähän haluaisin jatkossa pyrkiä, eli tekemään mahdollisimman ymmärrettävää ja yksinkertaisesti toimivaa koodia. On vastuutonta olettaa, että kukaan muu ei joutuisi tuottamaani koodia lukemaan. Mitä vähemmän jätän varaa arvailulle tai selvittämiselle, sitä vähemmän kollegoitani kuluu aikaa omien tehtäviensä tekemiseen, jos ne liittyvät tuottamaani koodiin. Lisäksi jos joudun palaamaan omaan koodiini, niin oman muistin varaan luottaminen ei kannata, pääsen huomattavasti helpommalla myös itse, jos ymmärrän heti lukemaani.

3.4 Seurantaviikko 4

– Maanantai 3.2.2025

Tänään aion keskittyä sekä parempaan ohjelmakoodin sisäistämiseen, että virheettömämmän koodin tuottamiseen. Haluan saattaa käyttäjien etsimiseen tekemäni toiminnallisuuden muutospyynnöksi, jotta voin pyytää jonkin kollegoistani testaamaan ominaisuutta mahdollisten virheiden löytämisen tai parannusehdotusten toivossa.

Toivoin saavani teknologiajohtajamme testaamaan muutoksia, koska hän oli toiminnallisuutta toivonut. Hän oli kuitenkin kiireinen, joten pyysin ryhmävetäjääni testaajaksi. Ryhmävetäjälläni oli monia arvokkaita neuvoja, jotka olivat suurimmaksi osaksi ulkoasuun kohdistuvia pieniä parannuksia,

mutta virheitä emme löytäneet. Olisin kuitenkin mielelläni löytänyt mahdolliset sellaiset tässä vaiheessa, ennen kuin joku asiakas olisi itse löytänyt jonkin virheen. Pidän kuitenkin päivää onnistuneena, varsinkin kun sain korjattua erään toisen virheen muualla ohjelmakoodissa. Kyse oli taas puuttuvasta varmistuksesta, jolloin ohjelmistomme osa ei toiminut oikein.

– Tiistai 4.2.2025

Tavoitteenani on tänään yrittää saada ratkaistua ongelma kuvien lataamisessa ja sitä kautta syventää ohjelmakoodin ymmärtämistä sekä edistää koodin tehokkaampaa tuottamista. Olen miettinyt erästä ratkaisua, jonka avulla kuvasuhteiltaan tai mitoiltaan sopimattomat kuvat voisi saada selvitettyä niille asetetun CSS-luokan ja erään JavaScriptin valmiin metodin avulla. Aion myös hieman viimeistellä käyttäjien hakemisen uutta ominaisuutta, koska sain ryhmävetäjältäni siihen joitakin vinkkejä.

Sain kuin sainkin kuvien lataamisen toimimaan haluamallani tavalla, vaikka ratkaisu ei löytynytäkään edellä mainitsemaani kautta. En loppujen lopuksi saanutkaan virheellisistä kuvista luotettavaa tietoa käyttämällä edellä mainittua keinoa, koska metodi palautti tosiaikaisen HTML-kokoelman. Kysyin myös Googlen Gemini-tekoälyltä apua ja selailin Stack Overflowta, mutta en oikein päässyt eteenpäin niiden ehdottamissa ratkaisuissa. Ratkaisin ongelman lopulta tavalla, josta en ole ylpeä, mutta ainakin se toimii.

– Keskiviikko 5.2.2025

Haluan tänään saattaa päätökseen tekemäni parannukset sovelluksen julkaisuun. Olin myös saanut eilen joitakin parannusehdotuksia ja toiveita käyttäjän hakuun tekemääni ominaisuuteen, jotka aion myös toteuttaa. Vielä lisäksi ehkä hieman kunnianhimoisena tavoitteenani on yrittää työstää aliverkkotunnusten ja linkkien muokkausta ja luomista, joka on odottanut jo pitkään huomiotani.

Sain julkaisun parannukset viimeistelyä siihen kuntoon, että ne vietiin jo tänään tuotantoon, samoin kuin pienet muutokset käyttäjän hakemisessa. Joitakin asioita, kuten esimerkiksi alkuperäisten kuvien lataaminen takaisin palvelimelta jäi vielä selvitettäväkseni, mutta olin muuten varsin tyytyväinen julkaisun muutosten tilaan. Samoin käyttäjän hakuun, vaikka kyseessä oli aika yksinkertainen toiminto käyttöliittymän osalta. Sain myös edistettyä viimeistä tavoitettani, joten olen varsin tyytyväinen päivään.

– Torstai 6.2.2025

Jatkan tänään aliverkkotunnusten ja linkkien luomisen sekä muokkauksen parissa. Tavoitteenani on saattaa parannukset ja korjaukset loppuun, jotta ehtisin kysyä kollegaltani yhden julkaisukuvien lataamisessa toivotun ominaisuuden toteuttamisessa. Tiedän hänen työskennelleen sen parissa versionhallinnan jäljiltä koodiin jääneiden merkintöjen ansiosta, mutta sain siihen myös varmistuksen ryhmävetäjältäni.

En onnistunut saamaan tehtävääni päätökseen, mutta sain edistettyä sitä kohtalaisen paljon. Tunsin jatkuvasti löytäväni uusia ratkaistavia ongelmia, jotka oli mielestäni käsiteltävä, jotta olisin voinut olla tyytyväinen työni jälkeen. Ominaisuuden alkuun saattaneen kollegani jäädessä joululomille olin saanut muutokset hoitaakseni, mutta koska kyseinen ominaisuus ei ollut tulossa vielä hetkeen tuotantoon, niin olin työskennellyt kiireisempien tehtävien parissa.

– Perjantai 7.2.2025

Tänään pyrin viimeistelemään eilen kesken jääneen tehtäväni. En aseta tälle päivälle muita tavoitteita, koska haluan saada muutokseni mahdollisimman nopeasti valmiiksi, jotta voin jatkaa jo tuotannossa olevien ominaisuuksien kehittämistä ja korjausta.

Sain muutokset lähes valmiiksi, mutta yksi ongelma linkkien luomiseen käytettävän valikon sulkeutumisessa ennen aikojaan jäi ratkaisematta. Yritin monia eri ratkaisuita, mutta en ehtinyt toteuttaa toimivaa sellaista työaikani puitteissa. Tutustuin kuitenkin työaikani ulkopuolella erääseen mahdolliseen vaihtoehtoon, jota odotan pääseväni kokeilemaan ensi viikon alussa.

– Viikkoanalyysi 4

Maanantaina ratkaisemani ongelma sai yhden ohjelmistomme osa-alueen epäkuuntoon. Kyseessä oli kilpailuiden osallistujien äänestämiseen lisättävät ulkoiset äänilähteet. Ymmärtääkseni kyseessä voi olla esimerkiksi jokin toinen, ohjelmistostamme irrallinen verkkosivu. Kyseinen ominaisuus on kohtalaisen uusi, eikä itselleni erityisen tuttu, mutta komponentti, jonka kautta tuo linkittäminen tapahtuu, kaatui ilman osallistujan kohdalle asetettua tyhjää objektia, jos osallistujia ei ollut ehditty vielä ladata.

Viikon paras saavutus oli kuvien lataamisen parannusten valmiiksi saaminen. Niin kuin olen jo aikaisemmin maininnut, niin olen mielestäni työskennellyt tehtävän parissa jo liiankin kauan. Joitakin pieniä parannuksia jäi vielä tehtäväkseni, mutta luulisin saavani ne tehtyä ensi viikolla. On kuitenkin helpotus päästä tekemään jotain muuta välillä.

Suurimmaksi ongelmaksi kuvien lataamisessa muodostui kelpaamattomien kuvien näyttäminen selkeästi käyttäjälle. Lähes kaikki yritykseni välittää tieto lapsikomponentin kelpaamattomista kuvista emokomponenttiin päätyivät lapsikomponentin jatkuvaan virkistämiseen, joka siis kaatoi ohjelmiston kokonaan. Viikon alkupuolella keksimäni ratkaisu, jota yritin tiistaina toteuttaa, käytti HTML-dokumenttien valmista `getElementsByClass`-metodia, joka siis etsi dokumentista kaikki elementit, joilla oli tietty tyyli luokka (Refsnes Data 2025d). Tässä tapauksessa etsittävä tyyli luokka oli itse nimelläni "invalid-picture", joka asetettiin jokaisen kelpaamattoman kuvan tyyli luokaksi lapsikomponentissa. Päätin siis yrittää etsiä kyseiset elementit emokomponentissa, mutta metodin palauttama lista poikkesi normaalista listasta siten, että se oli reaaliaikainen. Toisin sanoen, vaikka kuvat edelleen näyttivät ruudulla tyyli luokkani merkitsemiltä, niin tuo tieto oli tallessa vain kyseisessä virkistyksessä. Tosi aikainen dokumentti ilmensi jo niin sanottua normaalitilannetta, jossa kuville tuota tyyli luokkaa ei ollut asetettu. Yritin jonkin aikaa ratkaista ongelman asettamalla dokumenttiin tapahtumakuuntelijan, joka voidaan ohjelmoida etsimään tiettyjä tapahtumia tai muutoksia HTML-dokumentissa, kuten esimerkiksi hiiren painalluksia tai tyyli luokkien muutoksia, mutta en saanut ratkaisuja toimimaan oikein (Mozilla s.a. h). Loppujen lopuksi asetin tiedon lapsikomponentissa kelpaamattomista kuvista emokomponentissa sijaitsevaan tavalliseen objektimuuttujaan, jotta vielä korkeammalla tasolla sijaitsevat `useEffect`-koukut eivät olettaisi lapsikomponenttien tilan muuttuneen, millä taas vältyttäisiin komponentin ylimääräiseltä virkistämiseltä.

Linkkien ja aliverkkotunnusten luomiseen käytettävä ohjelmiston osa-alue oli jäänyt vähemmälle huomiolle jo pidemmäksi aikaa. Sain kyseisen ominaisuuden vastuulleni ennen joulupyhiä, jolloin kävimme sen tilaa läpi ominaisuuden kehittäneen kollegani kanssa. Ominaisuus oli vasta varhaisessa kehitysvaiheessa, joten sen parissa ilmeni monia huomiota vaativia tehtäviä. Muun muassa linkkien luomiseen käytettävä valikko ei ollut käytettävissä pienemillä ruuduilla, joilla sulkemis- ja tallentamispainikkeet jäivät piiloon, aliverkkotunnusten nimeä tai niihin linkitettyä sovellusta ei voinut muokata, pudotusvalikoiden arvot eivät näyttäneet valittuja, jotkut tekstikentät eivät toimineet oikein ja niin edelleen. Tärkein parannus on varmaan kuitenkin ollut linkkien rakentajan muuttujien arvojen säilöminen Redux-tilanhallintakirjastossa komponentin paikallisten `useState`-muuttujien sijaan. Kyseinen muutos vähentää koodin sekavuutta ja rivimäärää, kun useiden paikallisten tilamuttujien sijaan enää ainoastaan yhtä tilaa tarvitse käsitellä (Abramov ym. 2025, Single Source of Truth).

Edellisen tehtäväni myötä sain myös Googlen Gemini-kielimallin kanssa kerrata säännöllisten lausekkeiden perusteita (kuva 3).

✦ The regular expression `const pathAllowedChars = /^[a-zA-Z0-9\-_]+$/` defines a character set allowed in a path component (like a directory or filename). Let's break it down:

- `^`: This symbol means "start of string". The entire string being tested *must* begin matching at this point.
- `[a-zA-Z0-9\-_]`: This is a *character class*. It defines a set of allowed characters. It includes:
 - `a-z`: All lowercase letters.
 - `A-Z`: All uppercase letters.
 - `0-9`: All digits.
 - `\-`: The hyphen (minus sign). It's escaped with a backslash `\` because the hyphen has special meaning within a character class (it's used to define ranges, like `a-z`).
 - `_`: The underscore character.
- `+`: This quantifier means "one or more" occurrences of the preceding character or group. In this case, it means "one or more" of the characters defined in the character class.
- `$`: This symbol means "end of string". The entire string being tested *must* end matching at this point.

In summary:

This regular expression matches strings that consist *only* of one or more alphanumeric characters (letters and numbers), hyphens, or underscores. It does *not* allow any other characters, such as spaces, forward slashes, periods, or any other symbols.

Kuva 3. Keskustelu säännöllisestä lausekkeesta Googlen Gemini-kielimallin kanssa. Kyseinen lauseke tarkistaa käyttäjän syöttämän linkin polun sopivuutta annettuihin raja-arvoihin.

Sain perjantaina yhdeltä kollegaltani joitakin vinkkejä kuvien lataamiseen palvelimelta niiden alkuperäisessä koossa. Kyseessä on viimeisiä tehtäviä sovelluksen julkaisukuvien lataamisessa, mutta koska halusin saattaa isommat muutokset jo tuotantoon, niin päätin ratkaista tämän ongelman jälkikäteen. En tällä viikolla ehtinyt kokeilla vinkkejä, mutta nyt minulla on parempi käsitys ongelman luonteesta. Toivon sen auttavan ensi viikolla minua käsittelemään tehtävää tehokkaammin, jotta saisin kaikki minulta toivotut muutokset kunnialla toteutettua. Aion myös yrittää palata lyhyesti kelpaamattomien kuvien näyttämiseen tekemääni ratkaisuun, koska en ollut siihen ihan täysin tyytyväinen ja sain siihen viikonloppuna joitakin uusia ideoita.

3.5 Seurantaviikko 5

– Maanantai 10.2.2025

Tarkoitukseni on tänään ratkaista ongelma linkkien luomiseen käytettävän valikon sulkeutumisessa ennen aikojaan erään toisen valikon ollessa auki. Ongelma esiintyy luomani toiminnon yhteydessä, joka sulkee linkkivalikon, kun sen ulkopuolella painetaan hiiren painiketta. Kun toista avointa valikkoa yritetään käyttää, niin alkuperäinen linkkivalikko sulkeutuu.

Olin kuitenkin saanut ryhmävetäjältäni viestin ennen töihin tuloani, jossa hän pyysi minua katsomaan tarkemmin erään kollegani muutospyyntöjä. Luin siis ensi töikseni kyseisiä pyyntöjä erityisellä tarkkuudella läpi koko aamun. Koska pidän itseäni edelleen hyvin kokemattomana, niin en ole aiemmin uskaltanut kommentoida muiden tekemisiä erityisen aktiivisesti. Löysin kuitenkin kollegani tekemistä muutoksista joitakin kohtia, mistä pyysin tarkistusta. Lisäksi sain ratkaistua linkkivalikon ennen aikaisen sulkeutumisen ongelman, joten päivä oli varsin onnistunut.

– Tiistai 11.2.2025

Aion tänään kokeilla käytännössä viikonloppuna kehittämäni vaihtoehtoisia ratkaisua virheellisten kuvien näyttämisestä käyttäjälle. Lisäksi sovelluksen julkaisussa käytettävien ikonien sallitut kuvasuhteet eivät vielä täysin vastaa Applen App Storen ja Googlen Play Storen vaatimuksia. Yritän myös ehtiä tutkimaan, että voisinko toteuttaa kreikkalaisen tietovisan järjestäjien pyytämää ominaisuutta, jonka kautta kaikki tietyn kyselyn vastaukset voisi viedä erilliseen tiedostoon tai tiedostoihin.

Valitettavasti miettimäni ratkaisu ei edistänyt ensimmäistä tavoitettani. Onneksi sain edistettyä muita tehtäviäni jonkin verran. Ikonien ja erään toisen kuvan kuvasuhteiden vaatimukset oli helppo päivittää ja lisäsin niille myös vastaavan visuaalisen ilmeen, kuin monien kuvien lataamisen yhteydessä mahdollisesti hylätyille.

Ilmoitin iOS-kehittäjällemme, että kreikkalaisen asiakkaan toivoma ominaisuus pitäisi olla mahdollista, koska lähes vastaavia oli jo olemassa, mutta ne toimivat vähän kömpelösti. Sain kuitenkin tarkennusta tehtävään, sillä kävi ilmi, että asiakas toivoi tietojen viemistä CSV-tiedostoon. Tämän pitäisi helpottaa tehtävän mahdollista toteuttamista.

– Keskiviikko 12.2.2025

Tavoitteenani on tänään toteuttaa kuvien lataaminen palvelimeltamme niiden alkuperäisessä koossa. Tämän lisäksi kuulin tämän aamupäivän kokouksessa, että kreikkalaisen asiakkaan toimiva ominaisuus kyselyiden vastauksista CSV-tiedostona on varsin kiireinen, joten yritän edistää myös sitä.

Onnistuin saamaan jo ohjelmistoomme ladatut kuvat ladattua niiden alkuperäisessä koossa. Jotkin tietyt kuvat tosin skaalautuivat aina maksimissaan 2560 pikselin leveyteen tai korkeuteen, jos kumpikaan ylitti tuon arvon. Jouduin varmistamaan asian back end ryhmävetäjältä, joka kertoi kuvien skaalautuvan kyseisellä tavalla palvelimella. Sain myös toteutettua toiminnaltaan raskaahkon esiasteen kyselyiden vastausten viemisestä CSV-tiedostoon.

– Torstai 13.2.2025

Tänään tavoitteenani on yrittää edistää iOS-kehittäjäimme kysymää toimintoa kyselyiden vastausten viemistä erilliseen CSV-tiedostoon. Yritän myös ratkaista ongelman samassa näkymässä tapahtuvasta loppumattomasta komponentin virkistämisestä, jos kyselyiden vastauksia on runsaasti.

Viimeistelin tietojen viemiseen käytettävän toiminnon ja se tuntui toimivan lähes oikein. Kreikkalaisen tietovisan kyselyissä olevista vastauksista kaikki eivät kuitenkaan ilmeisesti päätyneet CSV-tiedostoon, koska erään kyselyn vastausten taulunäkymässä ilmoitettu vastausten määrä 1 827 kutistui noin 1 600 riviin. Jouduin myös poistamaan itse kirjoittamani funktion, joka tarkisti, että onko vastausten joukossa kuvia tai tiedostoja, joita käyttäjä voi ladata. Kyseisen funktion käyttäminen tuntui johtavan virkistysongelmaan.

– Perjantai 14.2.2025

Ryhmävetäjältäni saamien vinkkien perusteella päätin tänään yrittää parantaa eilen toteuttamiani muutoksia kyselyiden vastausten hakemisessa. Hän ehdotti ratkaisua, jossa useamman haun sijaan hakisin kaikki kyselyiden vastaukset kerralla perustuen jo ensimmäisen haun yhteydessä ilmoitettuun vastausten kokonaismäärään. Eilen viimeistelemäni ratkaisu haki tulokset sivu kerrallaan, joka tuntui hyvin aikaa vievältä ja muutenkin raskaalta, koska sivu myös vaihtui joka haun kohdalla ja sivun käyttäjien kuvat ladattiin myös jokaisen sivunvaihdon yhteydessä.

Sain apua ryhmävetäjältäni, jonka kanssa teimme yhdessä paranneltua toteutusta. Tällä kertaa tosin sain itse toimia koodin kirjoittajana, hänen antaessaan ohjeita ja neuvoja. Mielestäni opin näin paljon hyödyllistä tietoa esimerkiksi tehokkaammista toimintatavoista ja tietyistä JavaScriptin valmiista metodeista.

– Viikkoanalyysi 5

Olin varsin huojentunut saadessani tällä viikolla useita työstämiäni muutoksia valmiiksi. Linkkien luomiseen käytettävä valikko tarvitsee lisää huomiota, erityisesti ulkoasun osalta, mutta sain ilokseni huomata harjoittelijamme saaneen siitä tehtävän itselleen.

Viikon alussa ratkaisemaani linkin valikon sulkeutumisen keksin yrittää keinoa, jolla voisin yrittää estää sulkemisen toteuttavan funktion toiminnan, jos painallus tapahtuu jonkin tietyn HTML-elementin sisältämässä elementissä. Löysin lupaavalta vaikuttavan ratkaisun Stack Overflowsta, jossa opastettiin käyttämään elementin sisältämää `contains`-metodia (Stack Overflow 2013), joka vertailee kahta elementtiä ja palauttaa totuusarvon, mikäli verrattava kohde sisältää metodille parametrinä annetun elementin (Mozilla s.a. i). Verrattavan kohteen, eli toisen valikon luokkanimi oli helppo selvittää, ja jos painallus tapahtui tämän valikon lapsielementissä, niin alkuperäinen linkki-valikko ei sulkeutunut.

Ikoniin ja erään toisen yksittäisen kuvan lataamiseen käytettävä komponentti ei ollut sama, jota olin muokannut useamman kuvan lataamisen muutosten toteuttamisen aikana. Palasin tähän aiempaan tehtävääni yhtenäistämään muutoksia ja vielä tarkistamaan kuvasuhteiden ohjeistusta, sekä ratkaisemaan muita mahdollisesti ilmaantuvia ongelmia. Komponentti oli varsin laajassa käytössä, joten koska en halunnut rikkoa sitä muissa kuvien lataamisen toiminnallisuuksissa, niin päätin välittää sille totuusarvon, joka mahdollisti tekemiäni muutosten visualisoinnin ainoastaan komponentin ollessa kutsuttuna sovelluksen julkaisun osiosta. Mielestäni ratkaisuni vähentää mahdollisten virheiden esiintymistä huomattavasti, vaikka olisin mieluummin siistinyt koodia enemmän. Onneksi ryhmävetäjäni oli samaa mieltä kanssani, sillä hänen laskujensa mukaan olisin mahdollisesti joutunut ottamaan huomioon 39 muuta komponenttia, jos olisin lähtenyt sitä enemmän siistimään.

Kuvien lataamiselle alkuperäisessä koossa oli jo olemassa totuusarvo `originalSize`, joka välitettiin kuvien lataamisen suorittavalle palvelinpyynnölle. Jostain syystä tämän totuusarvon välittäminen ei riittänyt, vaan palvelinpyynnön toteuttamiseen oli unohtunut rivi, joka esti oikeaan ehtolausekkeeseen päätyminen. Poistin myös tuon rivin ja sain ongelman ratkaistua ilman tietoperustaani tukeutumista (kuva 4). Useimmiten toivon tosin vielä ohjelmointiurani alkupuolella kohtaavani ongelmia, jotka ratkaistakseni minun täytyy tutkia dokumentaatioita, keskustelupalstoja sisäisiä ohjeita ja niin edelleen, koska mielestäni opin siten parhaiten.

```

client/src/utils/image.js
@@ -57,12 +57,7 @@ export const serveImageSrc = ({
 57 57     { u: credentials?.id }[accessParam]
 58 58
 59 59     let endpoint
 60 -   if (
 61 -     imageAccess?.includes('public') ||
 62 -     !imageAccess ||
 63 -     !accessKey ||
 64 -     !credentials?.password
 65 -   ) {
 60 +   if (imageAccess?.includes('public') || !accessKey || !credentials?.password) {
 66 61     endpoint = getImageUrl({ imageKey })
 67 62   } else {
 68 63     const queryString = makeQueryString({
 69 64       [accessParam]: accessKey,
 70 65       ...(originalSize && { image_size: 'original', image_format: 'png' })

```

Kuva 4. Koska muuttuja `imageAccess` oli aina määrittelemätön, niin palvelinhakuun ei päätynyt kyselyä kuvan alkuperäisestä koosta. Kuva GitLabiin tekemästäni muutospyynnöstä.

Kyselyiden vastausten sivun virkistämisen ongelma oli itse tekemieni muutosten aiheuttama. Olin ryhmävetäjänä pyynnöstä yrittänyt estää "Export all images and files" painikkeen toimimisen, mikäli näkyvissä vastauksissa ei ollut ladattavia kuvia tai tiedostoja. Käytin tarkistamiseen omaa funktiota, joka tarkisti listoja niiden valmista some-metodia käyttäen. `Array.prototype.some()` tarkistaa, että toteutuuko sille parametrinä annettu funktio jossain listan sisältämistä arvoista (Mozilla s.a. j). Tässä tapauksessa tarkistukseen riitti kyselyn vastauksen sisältämien kuva- tai tiedostolistojen pituuden olemassaolo. Annoin tälläkin kertaa tekemäni funktion Geminin optimoitavaksi, mutta sen muokkaama ehdotus ei taaskaan toiminut oikein. Jouduin myös valitettavasti luopumaan ratkaisustani, koska suurissa vastausmäärissä se johti komponentin jatkuvaan virkistämiseen, koska sitä kutsuttiin aina komponentin päivittymisen yhteydessä ja se asetti paikalliseen tilamuuttujaan totuusarvon. Ongelma olisi korjaantunut kokeilemälläni komponentin päivittämisellä luokkakomponentista toiminnalliseen, mutta tämä ratkaisu oli mielestäni varsin ylimitoitettu suhteellisen pienen ongelman ratkaisemiseksi. Erityisesti siksi, että päivittäminen toi valitettavasti joitakin toivottomia seurauksia komponentin toimintaan. Aion kuitenkin palata ratkaisuun, koska mielestäni koodin nykyaikaistaminen on hyödyllistä ja ratkaisuni oli hyvin lähellä toivottua toimintaa.

Loppuviikon aikana työstämäni tehtävä kyselyiden vastausten CSV-tiedostoon viemisessä osoitautui osaamiseni kehityksen kannalta hyödyllisimmäksi. Ryhmävetäjänä kanssa tekemäni parikoodauksen ansiosta opin käyttämään JavaScriptin omaa Fetch API:a, joka tarjoaa rajapinnan HTTP-

pyynnöille ja pyyntöjen vastausten käsittelylle (Mozilla s.a. k). Teimme täysin uuden palvelinpyynnön kaikkien kyselyn vastausten hakemista varten. Valitettavasti kokeillessani palvelinpyyntöä käytännössä ilmeni, että se ei ratkaissut aiemmin torstain 13.2. päiväkirjamerkinnässä käsittelemääni tiedon katoamisen ongelmaa. Lisäksi rivimäärä kutistui entisestään, alle 900:aan 1 827:stä. Kysyin aiheesta back end ryhmävetäjältä, joka osasi kertoa maksimisivumäärän palvelinpyynnössämme olevan 1 000. Hän suostui nostamaan sen 10 000:een, mutta en ehtinyt enää perjantaina kokeilla uutta pyyntöä. Sain kuitenkin selvitettyä ongelman tiedon katoamisessa. Kyse ei sinänsä ollut virheestä, vaan kyselyyn osallistuneiden käyttäjien lähettämästä täysin tyhjästä vastauslomakkeesta. Muutin ohjelmakoodia siten, että näissä tapauksissa CSV-tiedostoon lisätään uusi sarake "Email", johon tallennetaan vastaajan sähköposti. Näin rivejä ei katoa, vaikka käyttäjä olisi osallistunut kyselyyn, mutta ei vastannut yhteenkään kysymykseen.

3.6 Seurantaviikko 6

– Maanantai 17.2.2025

Tänään aion testata tekemiäni muutoksia kyselyiden vastausten viemisessä CSV-tiedostoon. En usko siihen kuluvan koko päivää, joten aion myös edistää yhtä tai useampaa tehtävää, jotka tuotepäällikkömme oli raportoinut Jiraan.

Testasin useampaa kyselyä, joissa vastauksia oli yli tuhat, ja kaikissa näissä CSV-tiedosto tuli luoduksi ilman ongelmia ja vastausten määrää vastaavalla rivimäärällä. Olin tähän tulokseen varsin tyytyväinen, erityisesti siksi, että asiakkaamme oli jo ehtinyt kysyä ominaisuutta iOS-kehittäjälämmme. Hänen mukaansa asiakkaamme oli tyytyväinen, joten koen onnistuneeni ominaisuuden toteuttamisessa hyvin. Tosin hän mainitsi myös, että joillakin asiakkaillamme saattaa olla kyselyitä, joihin vastauksia saattaa tulla enemmän kuin 10 000, joten saatan vielä palata tehtävän pariin tulevaisuudessa.

Toisen tavoitteeni tehtävät olivat käyttöliittymäparannuksia ja virheiden ratkaisemista vielä julkaisemattomassa ominaisuudessa. Ominaisuus mahdollistaa tapahtumien luomisen ja on mielestäni kohtalaisen monimutkainen, koska se käsittää myös tapahtumapaikkojen ja -aikojen luomista sekä hallintaa. Päivämäärien ja aikojen käsittely ohjelmoinnissa voi joskus ainakin itselleni olla hankalaa. Päätin juuri siksi yrittää ratkaista näitä ongelmia, ja ehdinkin jo korjata monta niistä.

– Tiistai 18.2.2025

Ajattelin aluksi jatkaa tapahtumien rakentajassa ilmenneiden ongelmien kanssa, mutta sain päivittäisessä kokouksessamme kuulla ominaisuuden kehittäneen kollegani työstävän joitakin ratkaisuita samoihin ongelmiin. Päätin siis vain tehdä kohtalaisen helpot käyttöliittymän parannukset loppuun

ja yrittää ratkaista tilanteen, jossa kyselyiden vastauksia olisikin enemmän kuin tällä hetkellä yhdelle haulle korkein mahdollinen 10 000 kappaletta.

Vaikka olin asettanut tapahtumien rakentajan tehtävät itselleni ja olin myös siirtänyt tehtävät kategoriaan, jonka mukaan työskentelen jo niiden parissa, niin en lainkaan pahastunut kollegani lähtiessä selvittämään niitä. Olihan hän kuitenkin aloittanut ominaisuuden ja ainakin omaa mieltäni rauhoittaa, jos saan itse tuottamani koodin toimimaan paremmin. Saatoin siis vain loppuun yhden käyttöliittymäparannuksen ja vapautin loput tehtävät itseltäni sekä siirsin ne takaisin avoimiksi. Säästin kuitenkin muutamia tekemiäni virheiden ratkaisuita erilliseen Git-haaraan varmuuden vuoksi.

Tein myös loppuun aiemmin aloittamani kysymysten vastausten näyttävän komponentin päivittämisen toiminnalliseksi komponentiksi, estin kaikkien kuvien ja tiedostojen lataamisen vastauksista, jos niitä ei ollut sekä toteutin ratkaisun vastausten haulle, jos niitä oli enemmän kuin korkein haettava sivumäärä. Mielestäni päivä oli varsin onnistunut.

– Keskiviikko 19.2.2025

Sain eilen vinkkejä ryhmävetäjältäni päivittämäni komponentin jatkosiistimiseen ja ajattelin toteuttaa ne tänään. Aion myös jatkaa eilen aloittamaani tehtävää käyttäjän tarkempia tietoja näyttävästä valikosta käyttäjien listauksessa.

Onnistuin vähentämään paikallisten tilamuuttujien määrää luokkakomponentin alkuperäisestä yhdeksästä vain kahteen. Lisäksi päätin käyttää kehittämäni kaikki kyselyn vastaukset kerralla hakevaa funktiota vastausten sisältämien kuvien ja tiedostojen haussa, joka yksinkertaisti toimintoa huomattavasti.

Sain myös edistettyä kehittämäni yksittäisen käyttäjän tiedot näyttävän valikon niin pitkälle, että minun tarvitsee oikeastaan enää siistiä koodiani. Onnistuin tänään mielestäni tavoitteissani hyvin.

– Torstai 20.2.2025

Aioin tänään keskittyä käyttäjän tietoja näyttävän komponentin koodin siistimiseen, mutta sain toimitus- ja teknologiajohtajilta toiveita sovelluksen julkaisun kuvien lataamisessa. Koodin siistiminen on toki tärkeää, mutta kuvien käsittelyä vaativien toimintojen toteuttaminen voi olla haastavaa ja aikaa vievää. Tämän lisäksi eräs asiakkaamme saattaa tarvita ominaisuutta jo ensi viikolla, joten päätin keskittyä tähän tehtävään.

Olin seurannut molempien sovelluskauppojen antamia ohjeistuksia kuvien koosta tarkkaan ja oikein, mutta myönnän olleeni ehkä liian ankara käyttäjälle. Sain kuitenkin mielestäni annettua käyttäjälle enemmän tietoa ja vaihtoehtoja, mikäli kuvat eivät täyttäneet niille annettuja ehtoja.

– Perjantai 21.2.2025

Tänään jatkan tehtävääni kuvien lataamisessa. Aion myös keskustella aiheesta juuri aloittaneen visuaalisen suunnittelijamme ja back end ryhmävetäjän kanssa. Juuri häneltä opin aiemmin, että kuvia skaalataan palvelimelle ladattaessa joskus joka tapauksessa. Lisäksi hänellä saattaa olla toteuttaa mahdollinen ratkaisu kaikkien kuvien skaalaamiseen määrättyyn kokoon niiden päätyessä palvelimellemme.

Molemmilla kollegoillani oli antaa arvokkaita näkökantoja kuvien lataamisen käyttöliittymän parantamiseen. Päätimme kuitenkin, että jatkan muutosteni työstämistä, koska sitä kautta käyttäjällä saattaa olla parempi mahdollisuus vaikuttaa lopputulokseen. Kuvan kokoa muutettaessa palvelinpuolella on vaikeampi määrittää, mitä kuvasta jää pois, kun taas käyttöliittymässä käyttämämme kirjaston pitäisi pystyä mahdollistamaan kuvan tietyn osan rajaaminen. Mielestäni onnistuin edistämään tehtävääni hyvin ja olen vähän varmempi lopputuloksen toimivuudesta.

Sain hoidettavakseni myös erään kiireisen tehtävän, jossa piti ratkaista kilpailuiden luomisessa esiintyvä ongelma. Onnistuin ratkaisemaan sen, ja saimme korjauksen tuotantoon ennen lauantaita, jolloin kyseisen ohjelmiston osan oli toimittava moitteetta.

– Viikkoanalyysi 6

Tapahtumarakentajan tehtävissä sain syventyä päivämäärien käsittelyyn ohjelmakoodissa. Vaikka oma panokseni niissä jäi loppujen lopuksi varsin vähäiseksi, niin ennen kuin luovutin tehtävät kollegani hoidettavaksi, sain mielestäni arvokasta kokemusta. Mielestäni joissakin ohjelmistomme osissa käyttämämme Moment.js-kirjasto osoittautui varsin hyödylliseksi ja ohjelmakoodin luettavuuden kannalta sujuvammaksi, kuin JavaScriptin valmis Date-ohjelmointirajapinta. Erityisesti päivämäärien lisääminen Momentin add komennolla (Moment.js s.a.) oli mielestäni huomattavasti miellyttävämpää, kuin Date-objekteja käsittelemällä (Stack Overflow 2012). Tulevaisuudessa voi tosin olla hyödyllisempää tarkkailla JavaScriptin omaa Temporal-ohjelmointirajapintaa, jolloin ei ehkä tarvitsisi tukeutua vanhentuneisiin rajapintoihin tai ulkoisiin kirjastoihin (Mozilla s.a. I).

Vaikka sitä ei erikseen minulta pyydetty, niin päätin kuitenkin palata kyselyiden vastausten CSV-tiedoston viemiseen vielä viikon alussa. Tilanteita, joissa kyselyiden vastauksia olisi yli 10 000 ei

ehkä tule ihan heti eteen, mutta ajattelin siihen varautumisen vaativan kohtalaisen vähän aikaa ja vaivaa. Tiesin myös, että mikäli palvelimeltamme haetaan vähemmän tietoa, kuin siellä on antaa, niin palvelin antaa vastauksena avaimen seuraavan erän tai jäljelle jääneen tiedon hakemiselle. Tein yksinkertaisen while-lausekkeen, joka toisti hakua niin kauan, että kyseistä avainta ei enää löytynyt. Sekä alkuperäinen, että toistolausekkeen toteuttamat haut tehtiin async-funktion await avainsanaa käyttämällä, jotta kaikki haut odottavat edellisen valmistumista (Mozilla s.a. m).

Aiemmin mainitsemani vähäiset muutokset tapahtumarakentajaan korjasivat päivämäärän valintaan käytettävän kalenterin jäämisen osittain muiden sivuston elementtien alle, jolloin joitakin päiviä ei voinut valita ollenkaan. Muutokset oli helppo tehdä, mutta niiden ansiosta kalenterin tyyliasetuksiin, kuten kalenterin sijoittumiseen sivulla, on nyt helpompi vaikuttaa. Jostain syystä kyseiselle komponentille ei voinut aiemmin välittää className-muuttujaa, jonka kautta tyyliasetuksia on helppompi käsitellä (Meta Platforms, Inc s.a. c).

Päivittäessäni kyselyiden vastauksia listaavaa komponenttia toiminnalliseksi en huomannut keskittyä koodin siistimiseen. Yritin vain pitää koodin mahdollisimman samanlaisena, kuin se oli ollut alun perin luokkakomponentissa. Päivittäessäni huomasin kyllä, että joitakin paikallisia tilamuuttujia ei käytetty ollenkaan, joten poistin ne samalla. Ryhmävetäjäni ehdotusten perusteella sain kuitenkin supistettua tarvittavia tilamuuttujia edelleen. Itse kysymysten vastaukset voi viedä kahteen eri tiedostomuotoon, CSV ja PDF, ja lisäksi kaikki vastausten sisältämät vastaajien vastaukseen lataamat tiedostot ja kuvat voi ladata yhdellä hiiren painalluksella. Jokaista toimintoa varten oli oma tilamuuttujansa sekä tiedolle käynnissä olevasta latauksesta, että latauksen yhteydessä ilmoitettavalle viestille. Koska yksikään toiminto ei voinut olla saman aikaisesti käynnissä, niin asetin latauksen tiedon yhteen totuusmuuttujaan ja latausviestin yhteen objektimuuttujaan, jossa avain oli tiedoston muoto ja arvo oli näytettävä viesti. Lisäksi ratkaisuni kaikkien vastausten hakemisesta kerralla teki tarpeettomaksi ainoan jäljelle jääneen useEffect-koukun, joka haki ja vaihtoi jokaisen 25 rivin sivun kerrallaan. Jatkossa yritän itse kiinnittää huomiota vastaaviin tilanteisiin, joissa koodin päivittämisen lisäksi sitä voi yrittää siistiä.

Viikon puolivälissä otin hoitaakseni tehtävän, joka oli ollut avoimena Jirassa jo kohtalaisen pitkään. Kyseessä oli käyttäjien listauksessa näytettävä valikko, joka näyttää laajemmat käyttäjän tiedot käyttäjän riviä painettaessa. Tehtävän ohjeistus suositteli käyttämään saman tyyppistä valikkoa, kuin yksittäisen kyselyn vastauksen tietoja näytettäessä. Koska kyseinen valikko oli käynyt minulle varsin tutuksi aiempaa tehtävää suorittaessani, niin tehtävä oli kohtalaisen helppo. Sain ominaisuuden valmiiksi, mutta valitettavasti en ehtinyt yksinkertaistaa ja siistiä koodiani. Olen joutunut jättämään muitakin vastaavia tehtäviä suoritettavaksi myöhemmin, mistä olen hieman harmissani. Mielestäni koodin siistiminen on ensinnäkin hyvin tärkeää, sekä myös hyvin opettavaista. Pidän tästä

syystä nykyään tarkemmin kirjaa siistittävästä muutoksistani, mutta toivon pystyväni järjestämään niille myös aikaa.

Kiireisempänä tehtävänä sain palata parantamaan sovellukseen liittyvien kuvien lataamista. Erityisesti yksittäisten kuvien ja ikonien lataamisessa ohjelmistoomme käyttäjälle ei annettu tarpeeksi tietoa, miksi kuva ei kelpaa tai keinoja tilanteen korjaamiseen. Korjasin virheilmoitusta siten, että käyttäjälle näytetään virheilmoituksen yhteydessä oikea kuvakoko, joka aiemmin näytettiin ainoastaan ohjeistuksessa ennen kuvien latausikkunaa. Tutustuin myös kuvien rajaamiseen käyttämämme ulkoisen komponentin, tai oikeammin kirjaston ohjeistukseen tarkemmin selvittääkseni, että voiko samaa kirjastoa käyttää kuvien koon muuttamiseen. Onnistuin saamaan osittain toimivan ratkaisun, vaikka sitä ei ohjeistuksessa suositeltu käyttämään (Hervieu 2025, Props: cropSize). Katsoimme ohjeistusta yhdessä ryhmävetäjäni kanssa, joten en ollut onneksi päätöksessäni täysin yksin. Olin aiemmin ehdottanut erään toisen kirjaston käyttämistä kuvien koon muuttamiseen, mutta tiedän, että useisiin ulkoisiin kirjastoihin tukeutuminen saattaa tuoda epävakautta ohjelmiston toimintaan.

Perjantaina toteuttamani muutos kilpailuiden maksettujen äänien määrän rajoittamisesta samasta IP-osoitteesta tiettyyn määrään oli sinänsä yksinkertainen, mutta sain siinäkin arvokasta oppia. Määrälle oli voitava antaa arvoksi myös 0, joka ei jäänyt käyttöliittymässämme voimaan. Löysin kohtalaisen nopeasti syyn. Kun kilpailun arvoja päivitettiin ja haettiin uudestaan, niin kyseisen muuttujan arvoa tarkistettaessa ei otettu huomioon, että 0 arvolla tarkistus olettaa arvon puuttuvan. Osasin korjata koodia niin, että myös 0 on hyväksyttävä arvo, mutta kysyin ryhmävetäjältäni parempaa keinoa toteutukselleni. Hän opetti minulle Nullish coalescing-operaattorin käyttöä, joka palauttaa operaattorin oikealla puolella olevan arvon vasemman puolen arvon ollessa tyhjä tai määrittämätön ja muutoin palauttaa vasemmanpuoleisen arvon (Mozilla s.a. n).

3.7 Seurantaviikko 7

– Maanantai 24.2.2025

En ole vielä täysin tyytyväinen kuvan koon muuttamiseen käyttämämme kirjaston kautta. Kuva muuttuu oikeaan kokoon, mutta kuvan rajaamista näyttävä alue katoaa joskus, eikä kuvaa välttämättä näy ollenkaan käyttäjän yrittäessä muuttaa kuvan kokoa. Keskityn tänään korjaamaan toiminnon niin, että kuva ja kuvaa rajaava alue näkyvät oikein.

Sain heti ryhmävetäjältäni apua tehtäväni suorittamiseen. Keskityimme tosin enemmän komponentin epävakaiseen toimintaan, koska jo aiemmin kohtaamamme ongelma komponentin jatkuvassa virkistämisessä tuntui esiintyvän edelleen liian usein. Yritimme korjata ongelmaa melkein koko päivän, mutta päädyimme pitämään komponentin tilassa, johon olin sen jo aiemmin saattanut. Saan

todennäköisesti tehtäväkseni kirjoittaa toiminnoille uudet komponentit, jotka koskevat vain soveluksen julkaisuun liittyvien kuvien lataamista.

– Tiistai 25.2.2025

Keskityn tänään ratkaisemaan tietyn kuvatyypin lataamisessa esiintyvän ongelman, jossa kuvaa rajaava kirjasto olettaa kuvan olevan muotokuvamuodossa, vaikka sen leveys on isompi kuin korkeus. Olen aiemmin tehnyt tarkistuksen, jossa kuvan kuvasuhteista voidaan päätellä sen muoto. Toivon ratkaisun löytyvän nopeasti, jotta voin jatkaa toimintojen viemisellä erilliseen komponenttiin.

Valitettavasti jouduin käyttämään ongelman ratkaisemiseen enemmän aikaa, kuin olin toivonut. Sain kuitenkin mielestäni ominaisuuden julkaistavaan tilaan, joten aloitin kehittämään uutta siistimpää, ja toivottavasti myös vakaampaa versiota komponentista. En ehtinyt päästä erityisen pitkälle tässä tavoitteessani, mutta olen toiveikas uuden komponentin selkeämmästä ja vakaammasta toiminnasta.

– Keskiviikko 26.2.2025

Keskityn tänään edistämään kuvien lataamiseen tekemieni muutosten siistimistä. Tehtävä on sen verran laaja, että en usko ehtiväni tänään tekemään mitään muuta.

Aluksi minua jopa hieman ahdisti tehtävän laajuus ja koin vaikeaksi päättää mistä aloittaa. Päätin kuitenkin keskittyä aina pieneen osaan kerrallaan, ja olinkin saanut siihen ryhmävetäjältäni avuksi pienen paperille hahmotellun suunnitelman. Onnistuin saamaan itseni sellaiseen tilaan, että kaikki tuntui sujuvan luonnostaan. Sain tehtyä suurimman osan siistimisestä valmiiksi, ja olen siksi hyvin tyytyväinen päivään.

– Torstai 27.2.2025

Olen saanut kuvien lataamiseen tekemäni muutokset lähes siistittyä, mutta yritän saattaa ne tänään päätökseen. Lisäksi toimitusjohtajamme ilmoitti ominaisuuteen liittyvistä ongelmista, joten yritän ratkaista ne samalla.

Sain koodin siistimisen lähes valmiiksi, mutta en saanut hyvää ratkaisua annettuja raja-arvoja pienempien kuvien rajaamiseen. Isompien kuvien kanssa ei ole samaa ongelmaa. Lisäksi en vielä kään päässyt eroon komponentin jatkuvasta virkistämisestä, joka esiintyi virheellisten kuvien tiedon asettamisessa lista- tai objektimuotoisiin tilamuuttujiin.

- Perjantai 28.2.2025

Jatkan kuvien rajaamiseen, tai oikeastaan kuvan koon muuttamiseen sopivan ratkaisun etsimistä. Lisäksi kun juttelin aiheesta iOS-kehittäjäme kanssa, niin hän toisti toiveen eräästä jo aiemmin toivotusta ominaisuudesta. Aion yrittää ehtiä keskittyä molempiin tehtäviin, vaikka koen jo lievää stressiä tehtävien suorittamisesta.

Joudun todennäköisesti vain estämään liian pienten kuvien rajaamisen, koska vaikka sain sen toimimaan osittain, niin en saanut mielestäni tarpeeksi toimivaa ratkaisua, jotta ominaisuutta olisi järkevää säilyttää. Aloitin myös toisen tavoitteeni suorittamisen, jonka parissa sain onneksi kokea joi-takin onnistumisiakin.

- Viikkoanalyysi 7

En saanut edelleenkään kuvien lataamiseen tarkoitettua komponenttia toimimaan vakaammin, edes ryhmävetäjäni avustuksella. Päätimme yrittää eriyttää komponentin toiminnan täysin uusiin siistitympiin versioihin. Tätä ennen yritin kuitenkin saada selkeämpää ja toimivampaa ratkaisua kuvan koon tai rajauksen muuttamiseen sen ollessa väärän kokoinen. Eräs ratkaisuihini mahdollisti myös liian pienten kuvien lataamisen, jolloin kuvan rajaaja otti kuvaan ylimääräistä kuvan rajojen ulkopuolelta. Tämä ei kuitenkaan ollut iOS-kehittäjäme mielestä erityisen suotavaa, joten jätin toiminnon tilaan, jossa rajaaminen ei onnistu kuvan ollessa raja-arvojen alapuolella.

Ratkaisin myös ongelman kuvan rajaajan oletuksesta muotokuvamuotoisesta kuvasta. Kyseessä oli itse tekemäni virhe, jossa en ollut osannut ottaa huomioon, että tietyn kuvatyypin raja-arvot sallivat vain maisemamuotoisen kuvan. Tällöin tavoiteltava kuvasuhde tuli käännettyä turhaan esimerkiksi muotoon 9/16, vaikka se oli jo valmiiksi oikein muodossa 16/9.

Näiden korjausten jälkeen pääsin aloittamaan kuvien lataamiseen käytettävien komponenttien päivittämistä. Aluksi ajattelin saavani apua tehtävään ryhmävetäjältäni, mutta hänellä oli muita kiireisempiä tehtäviä hoidettavanaan, joten tämä tehtävä jäi minulle. Tiesin tehtävän olevan laaja ja kohtalaisen työläs, mutta yllätyin iloisesti, kuinka nopeasti edistyin tehtävässä. Pääasiassa lähdin eriyttämään komponentteja loogisempiin osiin. Haverbeken (2015, 176) mukaan ohjelman jakaminen tällä tavalla auttaa koodia tuntemattomia löytämään etsimänsä ja helpottaa ohjelmoijaa pitämään toisiinsa liittyvät osat yhdessä. Pyrin myös karsimaan turhaa koodia ja tehostamaan olemassa olevia toimintoja käyttämällä vähemmän koodirivejä saman asian tekemiseen. Kun aiemmin esimerkiksi yksi komponentti näytti sekä latauspainikkeen, että ladattujen kuvien näkymät, niin nyt

kaikki kolme on eriytetty kolmeen eri komponenttiin. Liikaa pirstaloitumista tulee toki myös varoa, mutta yritin seurata yleistä ohjetta, jonka mukaan yhden komponentin tulisi toteuttaa yhtä asiaa.

Valitettavasti edes komponenttien päivittäminen ei poistanut jatkuvan virkistymisen ongelmaa. Muutoin kohtalaisen onnistuneen päivittämisen jälkeen yritin asettaa tiedon virheellisistä kuvista taas sekä ylemmällä tasolla emokomponenteissa, että erikseen Reduxin tilanhallintakirjastossa, mutta tiedon asettaminen lista- tai tilamuuttujaan missä tahansa näistä tuntui aina saavan komponentit virkistymään jatkuvasti. Ryhmävetäjäni lupasi yrittää vielä ratkaista ongelman kiireellisemmät tehtävänsä suoritettuaan.

Toimitusjohtajamme ilmoittaessa minulle joistakin ongelmista edelleen kuvien lataamisessa aloin jo kokea lievää epätoivoa tehtävän suorittamisesta. Aluksi luulin ongelmien johtuvan väärän kokoisista kuvista, mutta kuvien rajaaja ei aina pysynytkään sille annetuissa arvoissa kuvia rajattaessa. En saanut kehitettyä täysin itseäni miellyttävää toteutusta ominaisuudelle, mutta koska en halunnut jäädä jumiin tähän yhteen tehtävään, niin päätin edistää toista.

Koska olin kuullut samasta ominaisuudesta jo kolmelta eri henkilöltä, niin ajattelin sen olevan varsin haluttu toiminnallisuus. Kyseessä oli kuvien vetäminen suoraan ne näyttävään kenttään, jolloin erillistä latausvalikkoa ei välttämättä tarvitsisi näyttää. Olin kysynyt ominaisuudesta aiemmin ryhmävetäjältäni, kun minulta ensimmäisen kerran sitä toivottiin, mutta hänen mukaansa se kuulosti varsin vaikealta toteuttaa. Hän spekuloi kentän olevan vaikea löytää hiirellä, jos siinä olisi valmiiksi paljon kuvia samalla, kun käyttäjä yrittää vetää lisää kuvia kenttään. Tämä ei onneksi osoittautunut ongelmalliseksi, vaan oikea kenttä löytyi aina kun hiiren vei sen päälle, vaikka siinä olisi ollut jo valmiiksi useita kuvia. Jouduin lisäämään kolmeen kohtaan div-elementin (kuva 5) ja asettamaan sille useRef-koukulla luodun viitteen, jonka kautta esimerkiksi elementin paikan jo koon tietoihin päästään käsiksi (Meta Platforms, Inc s.a. d). Tämän jälkeen lisäsin elementeille veto- ja pudotusfunktioita, jotka esimerkiksi estivät normaalin toiminnan pudotettaessa kuva selaimen ja suorittivat tiettyjä toimintoja pudotettaessa kuva kenttään (Refsnes Data 2025e). Normaalisti kuvan pudottaminen selaimen saa sen avautumaan uudella välilehdellä, mutta nämä lisäykset johtivat kuvien käsittelemiseen ja näyttämiseen niille tarkoitetussa kentässä.

```

274 +     <div
275 +         onDrop={e => onImageDrop(e)}
276 +         onDragOver={onDragOver}
277 +         onDragEnter={e => e.preventDefault()}
278 +         onDragLeave={onDragLeave}
279 +         ref={dropRef}
280 +     >

```

Kuva 5. Uuden div-elementin määrittely. Kuva GitLabiin tekemästani muutospyyntöstä.

Vaikka en saanut ominaisuutta valmiiksi, niin kuvien vetäminen suoraan ne näyttävään kenttään oli viikon paras saavutukseni. Olen tietysti tyytyväinen myös aluksi hankalalta tuntuneen koodin siistimisen ja päivittämisen onnistumiseen, mutta on mukava saada toteutettua monien toivoma toiminnallisuus. Huolimatta siitä, että tehtävät kuvien lataamisessa aiheuttavat jo kohtalaista stressiä, niin odotan innolla pääseväni takaisin niiden pariin ensi viikolla. Olen aika vakuuttunut siitä, että saan kuvien vetämisen toimimaan ja iloinen siitä erityisesti siksi, että ryhmävetäjäni piti tehtävää vaikeana. Vaikeuksia saattaa tosin vielä ilmaantua ja olettaisin näin tapahtuvan viimeistään, kun minun pitää ratkaista suoraan ladattujen kuvien koon tarkistaminen. Mielessäni on jo joitakin ratkaisuja, mutta nähtäväksi jää ovatko ne toteutettavissa.

3.8 Seurantaviikko 8

– Maanantai 3.3.2025

Keskityn tänään toiminnallisuuteen, jossa kuvia voi ladata vetämällä ne suoraan kuvakenttään. Sain aloitettua ominaisuutta viime viikon lopulla ja olen varsin toiveikas sen onnistuneesta toteuttamisesta. Toivon saavani edistettyä sitä siihen pisteeseen, että voin ruveta kokeilemaan ratkaisuitani kuvien kuvasuhteiden tarkistamisesta, kun ne on ladattu sivulle vetämällä ja pudottamalla.

Ryhmävetäjäni kertoi tänään auttavansa minua komponentin jatkuvan virkistämisen kanssa, mikä lievensi huomattavasti tehtävieni suorittamisesta kokemaani stressiä. Sain siis keskittyä täysin kuvien vetämiseen ja pudottamiseen, mikä johti hyviin tuloksiin. Kuvien lataaminen edellä mainitulla tavalla toimii hyvin, vaikka niiden kuvasuhteita ei vielä tarkisteta. Mielestäni tästä on kuitenkin hyvä jatkaa tarkistamisen toteuttamiseen ja muuhun viimeistelyyn.

– Tiistai 4.3.2025

Yritän tänään saada valmiiksi kuvien lataamiseen tekemäni parannukset. Lupasin toimitusjohtajallemme saada tehtävän valmiiksi tämän viikon alussa, joten koen jo kohtalaisen kovaa stressiä muutosten loppuun saattamisesta. En aio tänään keskittyä mihinkään muuhun.

Sain edistettyä tehtävää hyvin, mutta en saanut sitä vieläkään täysin valmiiksi. Olen kuitenkin tyytyväinen, koska kuvien vetäminen suoraan kuvakenttään toimii lähes moitteettomasti ja sain toteutettua ratkaisun, jonka ansiosta kelpaamattomien kuvien lataaminen avaa latausvalikon, jossa niitä voi mahdollisesti yrittää vielä rajata. Koon muuttaminen raja-arvoja pienemmille kuville ei ole kuitenkaan ratkaisussani mahdollista, mitä pelkään toimitus- ja teknologiajohtajien edelleen toivovan mahdolliseksi.

– Keskiviikko 5.3.2025

Pyrin edelleen saamaan valmiiksi kuvien latauksen parannukset. Tuntuu vain, että keksin tai saan koko ajan uutta parannettavaa tai korjattavaa. Yritän nyt vain keskittyä saamaan keskeneräiset tehtävät valmiiksi, enkä ottaa muita tehtäviä.

Löysin kuitenkin heti aamulla kuvista virheen, joka esti julkaistavan sovelluksen ikonien näkymisen oikein. Sain ratkaistua ongelman, mutta siihen kului lähes koko aamupäivä. Ratkaisu ei myöskään ollut paras mahdollinen, mutta siirryin silti muihin osa-alueisiin, jotka eivät olleet yhtä hyvässä tilassa. Toteutin ratkaisun yksittäisen kuvan ja ikonien poistamiselle, jota ei ennen jostain syystä ollut laisinkaan. Tein myös kuvien rajaajaan napin, jota painamalla rajaus skaalautuu oikeisiin mittoihin ja lisäsin vielä erillisen virheilmoituksen, jos kuva on raja-arvoja pienempi.

– Torstai 6.3.2025

Viimeistelen tänään kuvien lataamisen parannukset. Luulen saavani ne valmiiksi, koska jäljellä on vain enää jotain pieniä käyttöliittymäparannuksia ja joitakin pieniä virheitä korjattavana. Jos minulla jää aikaa, niin otan ehkä jonkin uuden tehtävän hoitaakseni Jirasta.

Annoin harjoittelijallemme erään tehtävän, mikä minulle oli annettu jo jonkin aikaa sitten. Kyseessä ei ollut mikään erityisen vaativa tehtävä, mutta en ollut löytänyt sen tekemiseen vielä itse aikaa. Sain myös viimeistelyt omassa tehtävässäni sen verran valmiiksi, että jäljellä on enää koodikatselmus, jonka toivon voivani tehdä yhdessä ryhmävetäjäni kanssa. Pyysin teknologiajohtajamme katsomaan muutoksia läpi kanssani, eikä hänen mielestään eilen tekemälleni automaattiselle rajaukselle ollut tarvetta, niin poistin ominaisuuden. Yksi pienehkö virhe jäi mieltäni vaivaamaan, mutta aion katsoa sitä huomenna yhdessä ryhmävetäjäni kanssa.

– Perjantai 7.3.2025

Eilen ennen kotiinlähtöäni sain kuulla tuotepäälliköltämme varsin pysäyttävästä virheestä, joka haittaa yhden tärkeän asiakkaamme sovelluksen toimintaa. Keskityn tänään etsimään ja korjaamaan kyseisen virheen ja sen jälkeen aion korjata myös eilen minua vaivaamaan jääneen pienen virheen sovelluksen julkaisuun liitettävien sovelluksen ikonien lataamisessa.

Menin kuitenkin ensin ryhmävetäjäni luo keskustelemaan päivän tehtävistä ja sain kuulla, että myös hän oli saanut kiireisen ongelman ratkaistavakseen. Aluksi ajattelimme saaneemme mahdollisesti saman tehtävän hoitaaksemme, mutta ilmeni, että kyse oli kahdesta eri ongelmasta, jotka molemmat haittasivat erityisesti samaa asiakasta. Päädyimme kuitenkin ratkomaan yhdessä niitä, ja vaikka enimmäkseen seurasin vierestä ryhmävetäjäni toimintaa, niin sain mielestäni erittäin arvokkaita oppeja virheiden paikallistamiseen ja ratkaisemiseen.

Edelliset ongelmat ratkesivat onneksi varsin nopeasti ja sain keskittyä ikonien lataamisessa esiintyvään epävakaiseen toimintaan. Onnistuin korjaamaan toiminnan vakaaksi ja tein joitakin muita pieniä parannuksia, joiden myötä tunsin vihdoinkin saaneeni sovelluksen julkaisun kuvien lataamisen parannukset valmiiksi.

– Viikkoanalyysi 8

Viimeisen seurantaviikon paras saavutus oli sovelluksen julkaisun kuvien lataamiseen tekemiäni parannusten saattaminen valmiiksi. Luulen joutuvani vielä palaamaan tehtävään, sillä annan muutenkin vielä toimitus-, teknologiajohtajan sekä iOS-kehittäjän testattavaksi ja oletan heillä olevan vielä joitakin parannusehdotuksia tai mahdollisesti löydettyjä virheitä korjattavana. Lisäksi kuvien lataamiseen vetämällä ja pudottamalla toteuttamani uusi koodi kaipaa siistimistä. Olen kuitenkin huojentunut päästessäni vihdoinkin tekemään muitakin tehtäviä.

Jouduin jonkin verran miettimään ratkaisua vetämällä ja pudottamalla ladattujen kuvien koon tarkistamiseen. Kuvien kuvakoon selvittämiseen tiesin yhden ratkaisun olevan entuudestaan aiemmin oppimani `img-elementtien onload-rajapinnan` käyttäminen (GeeksforGeeks 2024a). Minun täytyi vain keksiä ratkaisulle joku järkevä toteutus, koska edellä mainitun rajapinnan käyttö vaati kuvien liittämistä HTML-dokumenttiin kuvaelementteinä. Päätin luoda kuvakoon tarkistamiseen erillisen komponentin, jossa kuvat vietiin yksi kerrallaan listasta `map`-metodia käyttäen piilotettuun kuvaelementtiin, jolloin jokaiselle kuvalle tehtiin elementin `onload-rajapinnan` kautta välittämäni funktion tekemäni tarkistukset kuvan mitoista ja kuvasuhteesta. Mikäli yksikin kuvista osoittautui mittasuhteiltaan kelpaamattomaksi, niin komponentin toiminta lopetettiin asettamalla sen emokomponentissa sijaitsevaan totuusarvolliseen tilamuuttujaan kielteinen arvo ja kaikki kuvat vietiin jo olemassa

olevaan kuvien lataamisvalikkoon, jossa oli myös valmis toteutus virheilmoituksille sekä toiminoille tilanteen korjaamiseksi.

Sovelluksen julkaisuun lisättävien ikonien sivuston näkymästä katoamiseen löysin syyn ryhmävetäjäni vinkin perusteella. Hän oli huomannut, että yhtä kuvien lisäämiseen käytettävän lomakkeen arvoja muuttavaa funktiota kutsutaan liian usein. Arvelin sen kutsujana olevan erään tekemäni `useEffect`-koukun `onDone` callback-funktio. Callback-funktio annetaan argumenttina jollekin toiselle funktiolle ja sitä kutsutaan tämän toisen funktion suorituksen aikana (Mozilla s.a. o). Oletukseni osoittautui oikeaksi, joten lisäsin koukkuun ehtolauseen, jonka ansiosta koukku suoritetaan vain, jos ladattavien kuvien listassa on arvoja ja tilamuuttujan `loadDroppedImages` arvo on tosi (kuva 6). Tämä esti `onDone`-funktion tarpeettoman kutsumisen ja ikonien katoamisen uusia kuvia pudotettaessa ladattavaksi. Jostain syystä koukun toiminnot suoritettiin aiemmin useita kertoja komponentti näytettäessä, vaikka sille annettiin seurattavaksi muutokset vain totuusarvossa `loadDroppedImages`.

```

102   useEffect(() => {
103     if (imagesToLoad.length && loadDroppedImages) {
104       const ident = 'app-publish-images-promotion-form'
105       Promise.all([
106         ...imagesToLoad.map(async image => {
107           const imageData = await createFormData(image)
108           return addImage({
109             ident,
110             payload: imageData,
111             brandId
112           })
113         })
114       ])
115         .then(res => {
116           setIsLoading(false)
117           const mappedResponse = res.map(result => {
118             return Object.values(result?.response?.entities?.photos)[0]
119           })
120
121           imagesAdded(mappedResponse, ident, brandId)
122
123           if (onDone) {
124             onDone(fieldIdent, mappedResponse, input, parentKey)
125           }
126           setIsLoading(false)
127         })
128         .catch(_error => {
129           setIsLoading(false)
130         })
131       setLoadDroppedImages(false)
132     }
133   }, [imagesToLoad, loadDroppedImages])

```

Kuva 6. Luomani useEffect-koukku vedettyjen ja pudotettujen kuvien lataamiseen. Koukku sisältää myös myöhemmin analyysissä esittelemäni onDone-funktion. Kuvassa on Visual Studio-koodieditorissa avattu ohjelmakoodiamme sisältävä tiedosto.

Harjoittelijallemme antamani tehtävä oli varsin yksinkertainen, mutta se vaati jonkin verran työtä. Tehtävä ei myöskään ollut kiireinen, minkä takia sen tekeminen oli itseltäni jäänyt. Ohjelmistossamme varsin laajalti käytettävä komponentti salli kahden tyyliominaisuuden muuttamisen, mutta kyseiset ominaisuudet vaikuttivat vain kahdessa paikassa. Tehtävänä olikin piilottaa tai ottaa kyseiset asetukset pois käytöstä kaikkialla muualla. Kollegani tekemä komponentti sisälsi onneksi jo valmiin työkalun tähän, nimittäin komponentille välitettiin listana käytöstä poistettavien ominaisuuksien nimiä, tai ehkä tarkemmin avaimia, joita ei ominaisuuksien valikossa tullut näyttää.

Sain perjantaina ratkaistavakseni virheen, jonka toimitusjohtajamme oli löytänyt. Ongelma oli varsin pysäyttävä, mutta sen ratkaisu oli onneksi aika helppo. Sovelluksen ala- tai sivupalkkinavigaatioita ei voinut ongelman takia määrittää, koska arvoja asettava listan map-metodi sai käsiteltäväkseen määrittämättömän listan. Lisäsin koodiin varmistuksen edellä mainitun tilanteen varalta (kuva 7).

client/src/bundles/admin/reducers/app-builder.js	
↑	@@ -757,7 +757,7 @@ function onSetBottomNavigationItem(state, action) {
757	757 ...screen,
758	758 bottomNav: {
759	759 ...screen.bottomNav,
760	- navList: screen.bottomNav.navList.map((item, idx) => {
760	+ navList: screen.bottomNav?.navList?.map((item, idx) => {
761	761 if (idx === navItemId) {
762	762 return navigationItem
763	763 } else {
↓	@@ -968,7 +968,7 @@ function onSetSideNavigationItem(state, action) {
↑	
968	968 ...screen,
969	969 menuNav: {
970	970 ...screen.menuNav,
971	- navList: screen.menuNav.navList.map((item, idx) => {
971	+ navList: screen.menuNav?.navList?.map((item, idx) => {
972	972 if (idx === navItemId) {
973	973 return navigationItem
974	974 } else {
↓	@@ -2248,6 +2248,7 @@ function onChangeScreenState(state, payload) {
↑	
2248	2248 }
2249	2249 } else {
2250	2250 let newNavList = screen?.[navType]?.navList
2251	+ if (!newNavList) return screen
2251	2252 let newNavItem = Object.assign(newNavList[index], {
2252	2253 style: style
2253	2254 })
↓	

Kuva 7. Kuvassa lisäämäni varmistukset navigaatioiden käsittelemiseen, joita ilman ohjelmistomme lakkasi toimimasta. Kuva otettu GitLabiin tekemästäni muutokset sisältävästä yhdistämissyönteistä.

Tätä ongelmaa ratkaistessani löysin toisen virheen, joka esiintyi, jos navigaatiolle halusi määrittää tyyliaasetuksia, kuten esimerkiksi navigaatiopainikkeen taustaväriä ja niin edelleen. Jos sovelluksella oli useampia näyttöjä, eikä muissa näytöissä ollut määritelty yhtään ala- tai sivupalkin navigaatiota, niin ohjelma kaatui. Lisäsin samaan tiedostoon navigaatioiden tyyliaasetuksia käsittelevään funktioon kohdan, jossa näyttö palautetaan sellaisenaan, jos navigaatiolistaa ei ole.

4 Pohdinta

4.1 Kehityskohteet ja tulokset

Eniten koen kehittyneeni teknologioiden ja työkalujen sujuvammassa käytössä. Esimerkiksi aiemmin hieman etäiseltä tuntunut listan map-metodi on nykyään minulla lähes päivittäisessä käytössä. Ymmärrän myös paremmin esimerkiksi map- ja forEach-metodien välisen eron, nimittäin map-metodi palauttaa uuden listan, kun taas forEach ei palauta mitään, mutta käy vain listan arvoja läpi (GeeksforGeeks 2024b).

Objektien käyttö on nykyään jopa mieluisaa ja erityisesti objektin levityssyntaksi on tullut tutuksi. Sama syntaksi toimii myös esimerkiksi listaan ja merkkijonoon, mitä en tiennyt ennen seurantajaksoni alkamista (Mozilla s.a. c). Tiedon järjestyminen objektissa ja listassa on tärkeä ymmärtää, sillä joskus tiedot täytyy voida näyttää objektin järjestyksestä poikkeavalla tavalla, jolloin voi joutua käyttämään esimerkiksi listaa. Objektin avainten, arvojen tai molempien muuttaminen listaksi Object.keys, Object.values ja Object.entries funktioilla saattaa olla eduksi tällaisissa tilanteissa (Mozilla s.a. p, Description).

Muuttujien tyhjiin, määrittämättömiin ja muihin arvoihin varautuminen on nykyään minulle selvempää. Osaan myös paremmin hahmottaa, milloin varautuminen varmistuksella on tarpeellista, tai milloin se voi olla jopa haitallista. Esimerkki jälkimmäisestä tapauksesta voisi olla Nullish coalescing eli ??-operaattorin käyttö, joka ei välttämättä toimi oikein, jos tyhjälle arvolle on asetettu varmistus esimerkiksi tyhjäksi objektiksi ja tämä arvo annetaan ??-operaattorin vasemmalle puolelle. Nullish coalescing operator palauttaa sen oikealla puolella olevan arvon vasemman ollessa tyhjä tai määrittämätön ja muussa tapauksessa palauttaa sen vasemmalla puolella olevan arvon (Mozilla s.a. n). Esimerkissäni operaattori palauttaisikin siis vasemmanpuoleisen tyhjän objektin, vaikka ohjelmiston oikean toiminnan kannalta haluttaisiinkin palauttaa oikeanpuoleinen arvo.

Olen saanut tutustua HTML-elementtien sisäänrakennettuihin funktioihin, joista kerryttämäni osaaaminen tuntui yllättävän laajalta kollegoihini verrattuna. Esimerkiksi kuvien mittojen selvittäminen tätä kautta vaikutti olevan heille aika vierasta. Saman aiheen parissa työskennellessäni opin kohtalaisen paljon canvas-rajapinnasta, joka vaikutti yllättävän monipuoliselta työkalulta kuvien tai muiden visuaalisten elementtien käsittelyyn. Ilmeisesti canvasin pikselitason lähestymistapa on omiaan sovelluksen kuvien julkaisuun tekemässäni ominaisuudessa, jossa kuva piti piirtää pikseli pikseliltä käyttäjän rajaamaan alueeseen ja oikeaan kokoon (Haverbeke 2015, 295).

Osaan nyt jonkin verran paremmin tyypittää muuttujia TypeScriptissä, mutta sillä ei voi vielä väittää TypeScriptin osaamista kovinkaan laajaksi. TypeScript kuuluu tulevaisuudessa yhdeksi tärkeäksi

kehityskohteekseni, vaikka sitä en joutuisi nykyisessä työpaikassani enää erityisen paljon käyttämään. Yleisesti sen käyttö on kuitenkin lisääntynyt tasaisesti. Lisäksi haluaisin oppia paremmin käyttämään Pythonia, jonka käyttö on lisääntynyt vielä enemmän, myös web-kehityksessä. (JetBrains 2024). Pythonia olen käyttänyt viimeksi Haaga-Helian kurssilla, mutta esimerkiksi yrityksemme back end on toteutettu sitä käyttäen. En enää oikeastaan usko mahdollisuuksiini full stack kehittäjänä, mutta haluaisin joskus työskennellä back endin parissa, joka on oikeastaan aina kiinnostanut minua enemmän.

Yllättävän usein löysin itseni tilanteesta, jossa kollegoideni mielestä minulta pyydetty ominaisuus tai toiminto oli vaikeaa tai mahdotonta toteuttaa. Lähdin kuitenkin itse yrittämään ratkaisua ja useimmiten koin onnistumisia. Seurauksena sain paljon kaivattua itseluottamusta, joten jatkossa aion jatkaa samaa toimintamallia. Arvostan totta kai kokeneempia kollegoitani, mutta kaikkea heidän sanomaansa ei kannata ottaa absoluuttisena totuutena, vaan asioista kannattaa myös itse ottaa selvää.

Yleisesti koen kehittyneeni tiedon haussa ja omaksumisessa. Aiemmin esimerkiksi JavaScriptin ja Reactin virallisten dokumentaatioiden ymmärtäminen oli paljon hankalampaa. Nyt asiat selkiintyvät helpommin, vaikka kehitettävää on tässäkin edelleen. Kynnys tiedon hakemiselle kuitenkin madaltuu sitä mukaa, mitä helpommin käsitän dokumentaatiosta lukemani asiat. Olen myös käyttänyt tekoälyä tai kielimalleja tiedon hakemiseen enemmän kuin koskaan aikaisemmin. Vaikka ne eivät aina kyenneet antamaan toimivaa ratkaisua, niin niiden avulla minun oli helpompi soveltaa tietoa käytäntöön niiden tarjoamien esimerkkien kautta. Tiedon lähteestä riippumatta koodin lukeminen ja ymmärtäminen on kuitenkin koodin kirjoittamisen lisäksi erottamaton osa ohjelmoinnin opettelua (Haverbeke 2015, 7).

Lukemani koodin paremmalle ymmärtämiselle on muutenkin tarvetta, sillä uuden tiedon omaksumisen lisäksi tätä taitoa tarvitaan kollegoideni koodin katselmuksissa. Tässä kehittyäkseni minun täytyy edelleen ottaa selvää tuntemattomammista metodeista, funktioista ja operaattoreista, vaikka se saattaisi tuntua raskaalta satojen koodirivien seasta. Paljolti tämän opinnäytetyön tekemisen ansiosta minulla on tosin jäänyt tiedon hakemiseen tietty rutiini, joka helpottaa esimerkiksi näitäkin tilanteita. Lisäksi minun täytyy muistaa testata kollegoideni tekemiä muutoksia käytännössä, sillä kaikkia virheitä ei voi löytää pelkästään koodia lukemalla.

Joitakin kertoja olen saanut yhdeltä kollegaltani palautetta koodin kommentoinnista. Hän on toivonut minun tekevän niitä tilanteissa, joissa ei ole täysin selvää, miksi olen päätenyt johonkin tiettyyn ratkaisuun. Myönnän tekeväni kommentointia aivan liian vähän, mutta ehkä tähän on vaikuttanut aikoinaan eräältä alalla työskentelevältä ystävältäni kuulemma kommentti, jonka mukaan sitä ei tarvitse tehdä. Olen kuitenkin päätenyt olemaan sitä mieltä, että vaikka kommentointi saattaa lisätä

työmäärää, niin se tekee tuntemattoman koodin lukemisesta vähemmän painostavaa (Haverbeke 2015, 176).

Aiemmin toivottomalta tuntunut Git-versionhallinta on alkanut tuntumaan yhä luontevammalta. Uskallan toimia sen kanssa lähes täysin itsenäisesti, vaikka kyseessä olisi tehtyjen muutosten julkaisu tuotantoon. Saatan joskus vielä pyytää tukea kollegoiltani, mutta vain edellisessä tapauksessa. Omat muutokseni ja yhdistämispyyntöni osaan tehdä ilman muiden apua sekä ratkaista mahdolliset ongelmat niiden parissa.

Kuten olen jo aiemmin todennut, niin kukaan ei pysty kirjoittamaan täysin virheetöntä koodia. Olen kuitenkin jo harjoitteluni ja erityisesti seurantajakson aikana pyrkinyt tuottamaan sitä virheettömämmin. Yhtä lailla, kuten kollegoideni tekemät muutokset, myös omat muutokseni on testattava mahdollisimman kattavasti. Pysin aina selvittämään tuottamani koodin vaikutukset muissa ohjelmiston osissa niin perusteellisesti kuin voin ja ratkaisemaan ongelmat niiden alkulähteissä. Ongelmien ratkaiseminen niiden esiintymispaikassa on yleensä reaktiivista, ja enneminkin takaa toistuvat ja pahenevat ongelmien sarjat (IBM s.a. b). Seurantajakson aikana tietooni ei ole myöskään erityisen usein tullut tilannetta, jossa tuottamani koodi olisi aiheuttanut virheitä ohjelmistomme toiminnassa.

Avun pyytäminen on käynyt itselleni huomattavasti luonnollisemmaksi seurantajakson alettua. Toivon vain, että en tukeudu kollegoihini liikaa, koska ratkaisuita täytyy pystyä tekemään myös itse ja muille täytyy suoda työrauha. Lisäksi oppiminen ottamalla itse selvää asioista ja kokeilemalla käytännössä on varmasti tehokkaampaa, kuin muiden työskentelyn vierestä seuraaminen. On kuitenkin hyvä oppia tiedostamaan raja, milloin työtä ei vain saa enää etenemään tarpeeksi tehokkaasti yksin.

Olen saanut työstäni kiitettävää palautetta erityisesti ryhmänjohtajaltani, mutta myös teknologiajohtajaltamme. Koen täten onnistuneeni kehittämistavoitteissani hyvin, mutta tiedän, että minun pitää kehittyä edelleen. Tärkein kehityskohteeni jatkossa on ottaa vastaan ja suorittaa vaativampia tehtäviä, joissa pääsen todennäköisimmin tutustumaan uusiin asioihin. Suurin osa tehtävistä, joita suoritin seurantajakson aikana oli käyttöliittymään liittyviä parannuksia ja virheiden korjaamista, joissa voi usein suoriutua jo olemassa olevalla osaamisella tai muiden esimerkkiä seuraamalla. Haasteet ja uuden oppiminen kuitenkin kehittävät osaamista kaikkein parhaiten. Haverbeken (2015, 3) mukaan ohjelmoijat, jotka kieltäytyvät tutkimasta uutta seisauttavat, unohtavat ilonsa ja tylsistyvät ammattiinsa.

4.2 Yhteenveto

Seurantajakson aikana opin valtavasti uutta. En tosin koskaan aikaisemmin ole pitänyt yhtä tarkkaa kirjaa minkään osaamiseni kehittymisestä, joten ehkä myös siksi tämän työn tekemisen aikana kerryttämäni osaaminen tuntuu paljolta.

Onnistuin mielestäni asettamaan ammatillisen kehityksen kannalta merkitykselliset ja saavutettavat tavoitteet, joissa jokaisessa onnistuin kehittymään. Toisaalta tiedostan paremmin kohteita, joissa minun tulee vielä erityisesti kehittyä. Pystyin enimmäkseen pitäytymään suunnittelemissani aikataulussa, vaikka työn viimeistely viivästyi hieman. Opin myös asioita, joita en ollut kirjannut tavoitteikseni, kuten esimerkiksi parempaa ammatillista ilmaisu- ja monipuolisempaa tiedonhaku- ja muun muassa tekoälyä käyttäen. Kaiken kaikkiaan tunnen olevani paljon valmiimpi kohtaamaan uusia mahdollisuuksia ja haasteita ohjelmistokehityksen alalla.

Työssäni ei ole mainittu työnantajaani, kollegoitani tai asiakkaitamme nimeltä, joten työssä mainittujen tapahtumien yhdistäminen tiettyihin tahoihin on hankalaa. Esittelemiäni otteita ohjelmakoodistamme ei voi käyttää oikein ilman niiden viitekehystä. Näistä syistä työ on mielestäni myös eettisesti onnistunut.

Olisin voinut panostaa enemmän monipuolisempaan tietoperustaan ja parempaan lähteiden käyttöön. Erityisesti painettuun kirjallisuuteen perehtyminen on jäänyt työn aikana osaltani liian vähäiseksi. Koen tärkeäksi kehittyä myös tässä, koska ohjelmistokehityksen alalla joutuu kehittämään itseään jatkuvasti sekä pysymään ajan tasalla alan trendeistä, jolloin luotettavien tietolähteiden merkitys korostuu.

Juuri alalla tällä hetkellä vallitsevien trendien osalta osaamiseni kehittymisen ajankohtaisuus on tosin hieman ongelmallista, sillä tekoälyn kehittyminen ja sen vaikutus ohjelmistokehityksen työpaikkoihin tuntuu olevan jatkuva puheenaihe. Itse en kuitenkaan ole vielä erityisen huolissani, sillä uskon myös omiin kokemuksiini perustuen, että tekoäly ei vielä vähään aikaan voi yksin korvata ihmiskehittäjän luovuutta ja kekseliäisyyttä.

Lähteet

Abramov, D. et.al. 2025. Redux Fundamentals, Part 2: Concepts and Data Flow. Luettavissa: <https://redux.js.org/tutorials/fundamentals/part-2-concepts-data-flow>. Luettu: 10.2.2025.

Atlassian 2025. Welcome to Jira. Luettavissa: <https://www.atlassian.com/software/jira/guides/getting-started/introduction#what-is-jira-software>. Luettu: 9.5.2025.

Cambridge University Press & Assessment 2025a. Back end. Luettavissa: <https://dictionary.cambridge.org/dictionary/english/back-end>. Luettu: 9.5.2025.

Cambridge University Press & Assessment 2025b. Front end. Luettavissa: <https://dictionary.cambridge.org/dictionary/english/front-end>. Luettu: 9.5.2025.

GeeksforGeeks 2024a. How To Get The Height And Width Of An Image Using ReactJS?. Luettavissa: <https://www.geeksforgeeks.org/how-to-get-the-height-and-width-of-an-image-using-reactjs/>. Luettu: 15.1.2025.

GeeksforGeeks 2024b. Difference between forEach() and map() loop in JavaScript. Luettavissa: <https://www.geeksforgeeks.org/difference-between-foreach-and-map-loop-in-javascript/>. Luettu: 9.5.2025.

GitLab Inc. 2025. Merge requests. Luettavissa: <https://docs.gitlab.com/user/project/merge-requests/>. Luettu: 9.5.2025.

Haverbeke, M. 2015. Eloquent Javascript. 2. painos. No Starch Press, Inc. San Francisco.

Hervieu, V. 2024. react-easy-crop 5.0.8. Luettavissa: <https://github.com/ValentinH/react-easy-crop/blob/main/README.md>. Luettu: 21.2.2025.

IBM s.a. a. What is an API? Luettavissa: <https://www.ibm.com/think/topics/api>. Luettu: 10.5.2025.

IBM s.a. b. What is a root cause analysis? Luettavissa: <https://www.ibm.com/think/topics/root-cause-analysis>. Luettu: 9.5.2025.

JetBrains 2024. Welcome to the State of Developer Ecosystem Report 2024. Luettavissa: <https://www.jetbrains.com/lp/devecosystem-2024/>. Luettu: 9.5.2025.

Meta Platforms, Inc s.a. a. <StrictMode>. Luettavissa: <https://react.dev/reference/react/StrictMode>. Luettu: 15.1.2025.

Meta Platforms, Inc s.a. b. Updating Arrays in State. Luettavissa: <https://react.dev/learn/updating-arrays-in-state>. Luettu 16.1.2025.

Meta Platforms, Inc s.a. c. Adding styles. Luettavissa: <https://react.dev/learn#adding-styles>. Luettu: 18.2.2025.

Meta Platforms, Inc s.a. d. Manipulating the DOM with Refs. Luettavissa: <https://react.dev/learn/manipulating-the-dom-with-refs>. Luettu: 28.2.2025.

Microsoft 2025. TypeScript for JavaScript Programmers. Luettavissa: <https://www.typescript-lang.org/docs/handbook/typescript-in-5-minutes.html>. Luettu: 9.5.2025.

Moment.js s.a. Manipulate. Luettavissa: <https://momentjs.com/docs/#/manipulating/>. Luettu: 17.2.2025.

MongoDB, Inc. 2025. Full Stack Development Explained. Luettavissa: <https://www.mongodb.com/resources/basics/full-stack-development>. Luettu: 9.5.2025.

Mozilla s.a. a. What is JavaScript? Luettavissa: https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Scripting/What_is_JavaScript. Luettu: 9.5.2025.

Mozilla s.a. b. <input type="file">. Luettavissa: <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input/file>. Luettu: 23.1.2025.

Mozilla s.a. c. Spread syntax (...). Luettavissa: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Spread_syntax. Luettu: 30.1.2025.

Mozilla s.a. d. Array.of(). Luettavissa: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/of. Luettu: 24.1.2025.

Mozilla s.a. e. <table>: The Table element. Luettavissa: <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/table>. Luettu: 28.1.2025.

Mozilla s.a. f. Array.prototype.reduce(). Luettavissa: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/reduce. Luettu: 29.1.2025.

Mozilla s.a. g. Deep copy. Luettavissa: https://developer.mozilla.org/en-US/docs/Glossary/Deep_copy. Luettu: 10.5.2025.

Mozilla s.a. h. EventTarget: addEventListener() method. Luettavissa: <https://developer.mozilla.org/en-US/docs/Web/API/EventTarget/addEventListener>. Luettu 4.2.2025.

Mozilla s.a. i. Node: contains() method. Luettavissa: <https://developer.mozilla.org/en-US/docs/Web/API/Node/contains>. Luettu 10.2.2025.

Mozilla s.a. j. Array.prototype.some(). Luettavissa: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/some. Luettu: 12.2.2025.

Mozilla s.a. k. Using the Fetch API. Luettavissa: https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API/Using_Fetch. Luettu: 14.2.2025.

Mozilla s.a. l. Temporal. Luettavissa: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Temporal. Luettu: 17.2.2025.

Mozilla s.a. m. async function. Luettavissa: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/async_function. Luettu: 18.2.2025.

Mozilla s.a. n. Nullish coalescing operator (??). Luettavissa: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Nullish_coalescing. Luettu: 21.2.2025.

Mozilla s.a. o. Callback function. Luettavissa: https://developer.mozilla.org/en-US/docs/Glossary/Callback_function. Luettu: 9.3.2025.

Mozilla s.a. p. Object.entries(). Luettavissa: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object/entries. Luettu: 14.5.2025.

Refsnes Data 2025a. What is React? Luettavissa: https://www.w3schools.com/whatis/whatis_react.asp. Luettu: 9.5.2025.

Refsnes Data 2025b. HTML <option> Tag. Luettavissa: https://www.w3schools.com/tags/tag_option.asp. Luettu: 15.1.2025.

Refsnes Data 2025c. JavaScript Sorting Arrays. Luettavissa: https://www.w3schools.com/js/js_array_sort.asp. Luettu: 31.1.2025.

Refsnes Data 2025d. HTML DOM Document getElementsByClassName(). Luettavissa: https://www.w3schools.com/jsref/met_document_getelementsbyclassname.asp. Luettu: 4.2.2025.

Refsnes Data 2025e. HTML Drag and Drop API. Luettavissa: https://www.w3schools.com/html/html5_draganddrop.asp. Luettu: 28.2.2025.

Stack Overflow 2012. How can I add 1 day to current date?. Luettavissa: <https://stackoverflow.com/questions/9989382/how-can-i-add-1-day-to-current-date>. Luettu: 17.2.2025.

Stack Overflow 2013. How to check in Javascript if one element is contained within another. Luettavissa: <https://stackoverflow.com/questions/2234979/how-to-check-in-javascript-if-one-element-is-contained-within-another>. Luettu 10.2.2025.

Stack Overflow 2019. How to make rounded percentages add up to 100%. Luettavissa: <https://stackoverflow.com/questions/13483430/how-to-make-rounded-percentages-add-up-to-100>. Luettu: 30.1.2025.

Stack Overflow 2020. How can I deal with this Git warning? "Pulling without specifying how to reconcile divergent branches is discouraged". Luettavissa: <https://stackoverflow.com/questions/62653114/how-can-i-deal-with-this-git-warning-pulling-without-specifying-how-to-reconci>. Luettu: 20.1.2025.

Stack Overflow 2022. How do I correctly clone a JavaScript object?. Luettavissa: <https://stackoverflow.com/questions/728360/how-do-i-correctly-clone-a-javascript-object>. Luettu: 31.1.2025.

TechTarget 2025. GitLab. Luettavissa: <https://www.techtarget.com/whatis/definition/Git-Lab#:~:text=GitLab%20is%20an%20Open%20Source,large%20DevOps%20and%20DevSecOps%20projects>. Luettu: 9.5.2025.