

Working at a Digital Services Company (Diary Thesis)

Niko Jääskeläinen



Author(s) Niko Jääskeläinen	
Degree programme Business and Information Technologies	
Report/thesis title Working at a Digital Services Company (Diary Thesis)	Number of pages and appendix pages 70 + 0
<p>This portfolio-like diary thesis documents the daily work and professional development of a student in a small digital services company. The observation period for the diary thesis is 16.09.2019 – 29.11.2019.</p> <p>The diary thesis consists of a small analysis of the current work, history employment at the company, elaboration on company dynamics, interest groups at work, 11 weeks of documented diary entries and a final discussion with conclusions. The diary entries have been documented for each working day and at the end of the week there is an analysis into the week. Week analyses have certain themes to structure them better.</p> <p>The students' tasks include, but are not limited to software development, service design, user interface design, project management and user testing.</p> <p>At the end of the thesis there are discussions and notions on how the student has developed throughout the observation period, most notably in JavaScript development.</p>	
Keywords project management, software development, agile, design, prototyping	

Table of contents

1	Introduction	2
2	Framework.....	3
2.1	Analysis of your current work	3
2.2	Employment	3
2.3	Company dynamics & interaction.....	3
2.4	Client relations.....	4
2.5	Interest groups at work.....	6
3	Diary entries.....	7
3.1	Week 38	7
3.2	Week 39	17
3.3	Week 40	25
3.4	Week 41	30
3.5	Week 42	35
3.6	Week 43	39
3.7	Week 44	43
3.8	Week 45	46
3.9	Week 46	49
3.10	Week 47	53
3.11	Week 48	58
4	Discussion and conclusion	65
	References.....	69

Terms & Abbreviations

API	Application Programming Interface
CSS	Cascading Style Sheet
JS	JavaScript
React Component	Independent, reusable UI pieces
UI	User Interface
PR	Pull Request
CRUD	Create, Read, Update, Delete
GDPR	The General Data Protection Regulation
Figma	A collaborative interfacing tool
APK	Package file format used by the Android operating system
KPI	Key Performance Indicator

1 Introduction

This is a diary thesis focusing on weeks 38 to 48 spent working on a project which will be referred to as Project 1. I will be recording the events of each day and providing an analysis into each week on how things were done, what was done well, what could've been done differently or better and what I learned.

Observation period for the diary thesis is 16.09.2019 to 29.11.2019

Skills required in the work duties are software development, project management, knowledge of agile methodologies, Kanban, design tools, design patterns and social skills.

Difogic Oy is a digital service design company focusing on using user-centric design thinking and technological expertise to create products and services. Difogic has been operating for 8 years and has established its position in the small ICT circles. It is a small company with 6 employees.

The company's CEO is Vlad Bondarenko who hired, trained and with whom I have been working with closely during my employment. Elena Bondarenko is Difogic's Lead Software Engineer who worked on a major scale re-creation of 020202's data warehouse and pipelines. Kirill Chuprov is a Full-Stack Developer who is working remotely from Russia and has occasionally visited Finland, last time in October. Adrian Wilcock is a User Experience Designer who I helped to get employed at Difogic and has been conducting research into company culture and doing business development. Saara Peltomäki was a part-time Service Designer working with a small project relating to a rehabilitation clinic. We had a marketing assistant start in mid-august 2019 to lessen our workload in communications.

My title in the company is Technical Project Manager, my duties are, but not limited to; handling client communications, setting up meetings, digital service wireframing, JavaScript development in React, React Native and other related JS technologies, product development related research and interviews, user testing, UI development, development process structuring and product ownership. In other words, a "generalist" would be more suitable title, as it is called in the tech field.

2 Framework

2.1 Analysis of your current work

This section elaborates on my current work duties and responsibilities.

2.2 Employment

I started as a Technical Project Manager on October 1st in 2018. The position I originally applied for was a Digital Service Apprentice, which soon turned out to be the CTO's plan for the future of the company and its rebranding. The purpose of my hiring was to have a contact person handling the communications and development on the project which was being kicked off on the 3rd of October 2018.

2.3 Company dynamics & interaction

Our current office is located next to the shopping mall, Kamppi, in Sähköotalo. When I started working at the company, we had no own office space and the work was done in the coworking space of BC Papula, which has rebranded to Village Works earlier this year. Coworking space was used due to the fact that 2 out of 4 employees in the company worked remotely. After the company had its 5th employee start in February, we looked into getting our own office space. This was due to the fact that with 3 people present at Sähköotalo we could have "brainstorming" sessions and we started doing business development at the same time, so it'd be good to have our own space where we could discuss freely with the comfort of knowing no one is eavesdropping on our conversations. Also, the fact that Microsoft's Flux was shut down and the people who used it as their working space migrated to Sähköotalo coworking, so it was getting very crowded.

With 3 of our company employees having our own physical office space, we could focus on other things than the current tasks at hand and have general discussions. These discussions would vary from free time interests to more in depth analyses of current trends and business insights. Our roles in the company began to become very clear with more interactions happening. Our CTO would be the brains of everything due to his nature of being extremely educated and whenever looking up on something he had said it was always correct and in line with the current scientific methods, facts etc. The UX Designer would be the person looking at the more empathetic side of things and bring some balance to the cold hard facts while maintaining a logic-based approach. I was assigned as the naysayer in the group of three at first, to bring criticism into the discussions, but as we quickly found out, it was difficult for me to force a counter argument into many of the

things discussed, so that arbitrary role was dropped. After some time, our roles grew more naturally and we've been able to have discussions without them being completely one sided and biased, even though our visions are quite aligned.

In the beginning when we were conducting research into company culture and interviewing many people in the Helsinki tech sector we picked up on methods and practices from other companies and applied the ones we would see fit for our company, namely weekly meetings and briefs. We started having two meets a week, one on Monday and one on Friday. These were general briefs on what we had coming up the coming week and how we'd performed the past week and if there were any improvements to be done or some special insights. Once summer started and the whole Helsinki tech sector quietened down, we did not have weekly meetings. After summer we had time period where there was no need for them.

After summer holidays the workload exploded exponentially, and we'd been quite busy with existing clients and had little time to do business development. The company hired a new person to fulfil the role of creating client relations and finding new interesting projects.

2.4 Client relations

As my original reason for getting hired was someone to handle client relations while having technical knowledge of software development, I've been very active in that role to this very day. Initially my role was getting to know the Project which I was assigned to and to facilitate the communications between the two companies, ours and the client's. At first, I was very confused, because I had little knowledge of the business side and how things are handled in the "real" IT field and to the fact that my studies had mostly focused on development and design. While I had done the business courses in school, most of the terms and methods I came across always sounded familiar, but I did not understand them completely so a lot of my time was spent on reading up on the business side of my work.

In the beginning my client relations consisted of spending a lot of time with the clients since they were preparing their service to be piloted so they were meeting with shareholders and potential customers. I was present in meetings as a "technical consultant" in case there were any questions about the technical challenges and timeframes. Most of the time I gave half answers the like of "It can be done, but we'd need to test it out first. Approximate time to develop would be 1-1.5 weeks", which I quickly found out to be an acceptable answer to a surprisingly large amount of questions. Because I had not had the time to properly get to know the structure and codebase of the

Project yet, it was difficult to estimate times. As time went by and I started doing more software development I got to know the project structure better. This allowed for me to give out better time estimates, but time estimates in software projects are never constants, so my boss told me to exaggerate the times. We could never know if there were going to be some unexpected issues. He suggested for me to “always under promise and overdeliver”.

With my minimal experience and enthusiasm to overperform, I kept the client updated constantly with excessively long emails listing all future development plans, current software issues and general helpdesk type of answers to their questions relating to their own service. It is worth mentioning that for the both of us, the client and I, this was our first software project we worked on. My boss has done multiple software projects and he knows what can happen in them to the core, so he was advising the client to make the right decisions, but they were not listening most of the time. They had to go through the pitfalls and mistakes themselves to learn how things should be done.

As time passed and I got to know the client, the project and the business world better, I learned a lot and gained more confidence which enabled me to form my own thoughts and provide analytical insights into making better choices within the project. I was no longer simply tagging along and answering questions about time estimates. My boss was confident to let me handle the Project on my own, so he had more time to focus on other things our company was working on.

2.5 Interest groups at work

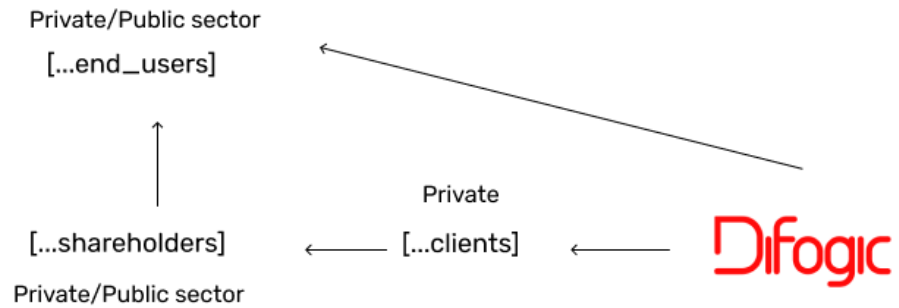


Figure 01. Interest groups at work

Our company does work for and with private companies, and most of the work received is by word of mouth due to the small circles in the IT field, so the end results and quality of work matter.

3 Diary entries

3.1 Week 38

Monday 16 September 2019

Today's objectives are to make sure a testing session for Project 1 goes well technically at Helsinki University, to make a v.1.0.0 build of the app that is being tested and to make sure there are no major bugs with the release.

The day started off with me going over my changes which I'd done for the version 1.0.0 of the Project 1's app. I reviewed my own changes first, making sure there were no overlooked mistakes and excess code. I had just added Google Analytics (Google Analytics 2019) to the app, which tracks some events and actions the user does in the app. When we release the app from the beta testing phase, we could get real usage data of the app and be able to make informed decisions for the next steps. After looking the changes over I made the pull request and added my boss as the reviewer.

After the PR I worked on our UI Kit for the Digital Prototyping Week in Figma. DPW is facilitated by The Shortcut. The Shortcut is a non-profit talent accelerator. Our company is going to have a workshop about rapid prototyping on the third day, so we are preparing our assets for the event.

After some time on the UI Kit I headed out to Helsinki University for the Project 1 session. My role there was mainly to be technical support in case something does not work or if there are any issues with logins and such. Only a few login issues were present, which were related to users having their accounts created for them, so there was some confusion about passwords. Overall the session was not focused on using the app itself, but the Helsinki Uni students were supposed to use the app on the side as a supplementary tool for reaching their pedagogical goals.

The session before this one was organized differently and was much more controlled. The app seemed to provide more value in a controlled environment, so in this session the app was not used as much. What was valuable about this session though, was the fact that the students questioned the app's functionalities, the use cases for them and created dialogue.

After the session we discussed the feedback with the project owner and one other person from the client's side. In the discussion we talked about what the next steps would be. We have a meeting on Wednesday with a Vantaa school representative, so we planned the agenda a bit for that day. Since the consumer version of the app was now waiting to be published, I recommended to use the consumer version for testing in Vantaa, because the teachers are currently overwhelmed with the number of technical tools in the teaching environment. The general atmosphere can be very negative when it comes to new tools and apps, so my reasoning was to ease the users into the service. This way they would not be committed to anything and the usage of the app would bring nothing but benefit in their daily work, because they would be able to only browse content. The project owner argued against and said that the testing should be done with the current "enterprise" version with all functionalities. This way it would be better to pitch the service to the people in charge of IT project acquisitions for the education sector. I recommended to have 2 small kindergarten groups where one would use the consumer version and the other the enterprise version to get different type of feedback on the app for validation of the functionalities.

Tuesday 17 September 2019

Today's objectives are to address multiple emails received on Monday evening. Check their content and if there is anything to address in the released dev build.

The emails from one of the people on the client's side were regarding content differences between the development server and production server, so I explained that they are completely separate from each other for development purposes. Another one was regarding Apple's email about API deprecation, which I had addressed in Monday's email about the new build.

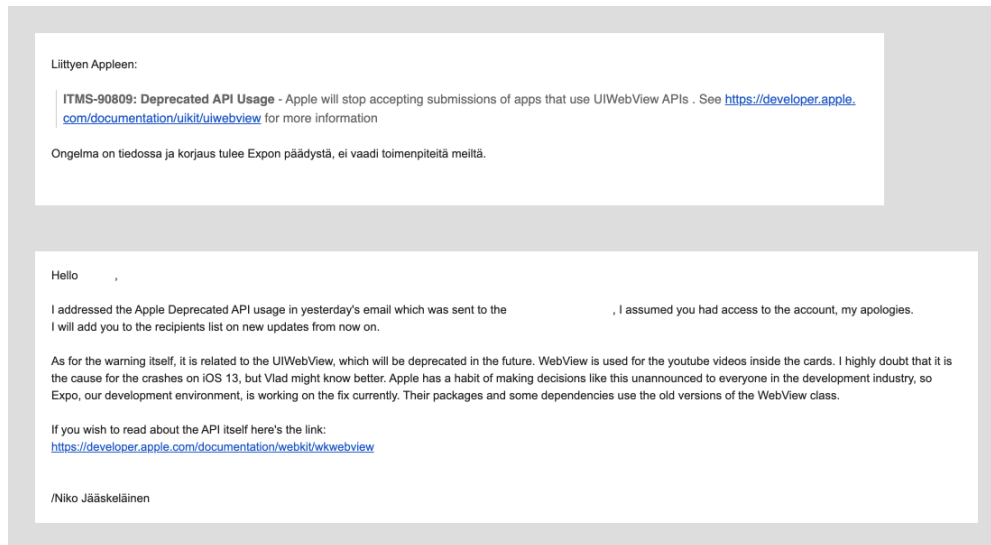


Figure 02. Above: Email snippet from Monday's email. Below: Answer to the client.

After some while I got an answer and apparently the person had just missed the email. Other emails were regarding the Helsinki University users for the service, which required no action from my part.

After the emails I discussed with my boss what to do next with the service and decided on adding the option to edit the user's own followed tags. Currently the only way to choose tags to follow was during onboarding and the user could go through that once. Since the functionality was already there, I decided to create a separate UI component for the tag cloud.

```
Niko Jääskeläinen, 18 hours ago | 1 author (Niko Jääskeläinen)
1 import React, { Component } from 'react'
2 import { Text, StyleSheet, ScrollView, TouchableOpacity } from 'react-native'
3 import { tags } from './tags'
4 import { colors, constants } from './styles'
5
6 const TagItem = props => {
7   return (
8     <TouchableOpacity
9       key={props.tag}
10      onPress={() => props.selectTag(props.tag)}
11      style={props.selected ? styles.tagBlue : styles.tag}>
12      <Text style={props.selected ? { color: 'white' } : { color: colors.blue.deep }} >#{props.tag}</Text>
13    </TouchableOpacity>
14  )
15 }
16
17 class TagCloud extends Component {
18   render () {
19     const { selectTag, selectedTags, style } = this.props
20     return (
21       <ScrollView contentContainerStyle={styles.cloud} style={[styles.cloudStyle, style]}>
22         {tags.map(tag =>
23           <TagItem
24             tag={tag}
25             key={tag}
26             selectTag={selectTag}
27             selected={selectedTags.includes(tag)} />)}
28       </ScrollView>
29     )
30   }
31 }
32
33 > const styles = StyleSheet.create({
34   })
35
36 export default TagCloud
37
```

Figure 03. TagCloud.js component file

After getting some basic functionality done for the profile editing, I noticed that when logging on to the app, the followed tags were not visible in the edit profile screen. I tracked the cause to the backend, where the *followedTags* property was not populated on login. That was a one-line backend fix where I added the user schema's *followedTags* into the select statement on login.

After all the fixes I deployed a new development build for iOS and Android and made sure the app worked without any major bugs or crashes.

Wednesday 18 September 2019

Today's objectives are to fix the failing tests for the pipeline on the web service, attend the meeting with the client and Vantaa stakeholder regarding Project 1 and see what the next steps are. In addition to that my plan is to begin on adding an edit option for the app's task functionality, which would allow the editing of tasks.

I started the day with fixing the tests for the web service. The tests were failing because of one unused variable and deep equal comparison of the test user in a login function. Deep equal was fixed with adding a followed tags parameter to the test user data.

The meeting with the Vantaa stakeholder and the client was in Oodi at 12 and we had reserved 2 hours for it. In the meeting it was discussed about which kindergarten is going to be the next pilot. The kindergarten chosen was the size of 20 employees.

After the meeting I left to the office to start work on a few bugs that were present at the meeting when going through the app. Turned out the *followedTags* were not being displayed properly once the user had gone through the onboarding and they had to refresh the whole app for them to be visible. Seemed that the backend one-line fix did not resolve the issue, so I had to investigate it even further.

In addition to the tags not being visible the search seemed to be broken for some reason. The app would display an empty screen and then not refresh at all. Kirill was tasked to find out why, and he came across an error where at times the aggregation was run before count in the database there were no results. We were going to check this more thoroughly tomorrow.

Thursday 19 September 2019

Today's objectives are to deploy the dev build with fixed search and get the parameter query done.

The day started with multiple emails from the client addressing the broken search. One of the people from the client's side went even to list out the individual search terms which do not work on the app. After replying to one of the emails and CC'ing all of the client's side people letting them know we are working on the fix, we looked at the issue with my boss.

The issue appeared with the "adaptive" search, as we called it, in the categories part of the aggregation function. The categories were using grouping by *\$first* instead of *\$addToSet*. Once that was fixed the backend could be deployed quickly and the app did not require a new build, since the changes were on the backend. The client still emailed a few times to confirm that the search works and even the different search terms used again. After some while they emailed again saying it does not pick up on different descriptions, so we added a few more fields into the search query function. Adding even more fields would've saturated the results and since the explanation of the query to the

client was not a realistic option, we decided to do the bare minimum to keep it working while providing accurate results.

```
301   const getCardSearchQuery = q => ({
302     $and: [
303       {
304         $or: [
305           { title: { $regex: q, $options: 'i' } },
306           { description: { $regex: q, $options: 'i' } },
307           { categories: { $regex: q, $options: 'i' } },
308           { goals_info: { $regex: q, $options: 'i' } },
309           { preview_description: { $regex: q, $options: 'i' } }
310         ]
311       }
312     ]
313   })
```

Figure 04. Search query function using regex

```

338 let cards
339 if (followedTags && followedTags.length) {
340   cards = await Card.aggregate([
341     { $match: query },
342     { $unwind: { path: '$categories', 'preserveNullAndEmptyArrays': true } }, // for empty arrays
343     { $addFields: {
344       matchesNum: {
345         $cond: [{ $in: ['$categories', followedTags] }, 1, 0 ]
346       } },
347     { $group: {
348       _id: '$_id',
349       language: { '$first': '$language' },
350       deck: { '$first': '$deck' },
351       author: { '$first': '$author' },
352       title: { '$first': '$title' },
353       description: { '$first': '$description' },
354       type: { '$first': '$type' },
355       helpers: { '$first': '$helpers' },
356       reviews: { '$first': '$reviews' },
357       score: { '$first': '$score' },
358       tutorials: { '$first': '$tutorials' },
359       resources: { '$first': '$resources' },
360       apps: { '$first': '$apps' },
361       tech: { '$first': '$tech' },
362       activities: { '$first': '$activities' },
363       activities_info: { '$first': '$activities_info' },
364       goals: { '$first': '$goals' },
365       categories: { $addToSet: '$categories' },
366       updated: { '$first': '$updated' },
367       created: { '$first': '$created' },
368       coverUrl: { '$first': '$coverUrl' },
369       thumbnailUrl: { '$first': '$thumbnailUrl' },
370       matches: { $sum: '$matchesNum' }
371     } },
372     { $sort: { matches: -1 } },
373     { $limit: limit },
374     { $skip: skip }
375   ])
376 } else {
377   cards = await Card
378     .find(query)
379     .sort(sortOrder)
380     .skip(skip)
381     .limit(limit)
382     .populate('author')
383     .populate('helpers')
384 }
385

```

Figure 05. Conditional part of the search function with existing followedTags on user

After the search was fixed, I worked on the parameter query by reverse engineering the already existing API's, which apparently involved a lot of abstraction and obfuscation in the codebase. I spent a lot of time to get it to work properly, and it was almost working with basic CRUD functionalities by the end of the day, but my work performance was not the best, so I decided to get back to it tomorrow.

Friday 20 September 2019

Today's objectives are to address the emails about the Google Play app rejection and other issues regarding user deletion and visibility of the app.

After ending the day with GDPR related discussion yesterday, Google Play had rejected the latest android build for some reason. I emailed the person from the client's side who has the developer account which receives the errors and other information from Google Play regarding the new APKs. Turns out the new build was rejected because of misleading metadata. The app's store page had not been updated in a while and the screenshots were old and did not reflect the current state of the app. The app icon did not seem to be compliant with the new specifications either.

I emailed the client regarding the issue and the fixes for it, which would be updating the descriptions, providing up to date screenshots and the new graphic assets. I mentioned that we will do the graphical assets and that the client should do the copywriting, since it's their expertise.

I started on updating the graphical assets of the project and decided to write the descriptions as well, because the client was not responding to the email and I knew they would be emailing me why the new build is not out. Graphical assets were done in Figma with a template that is provided by Google.

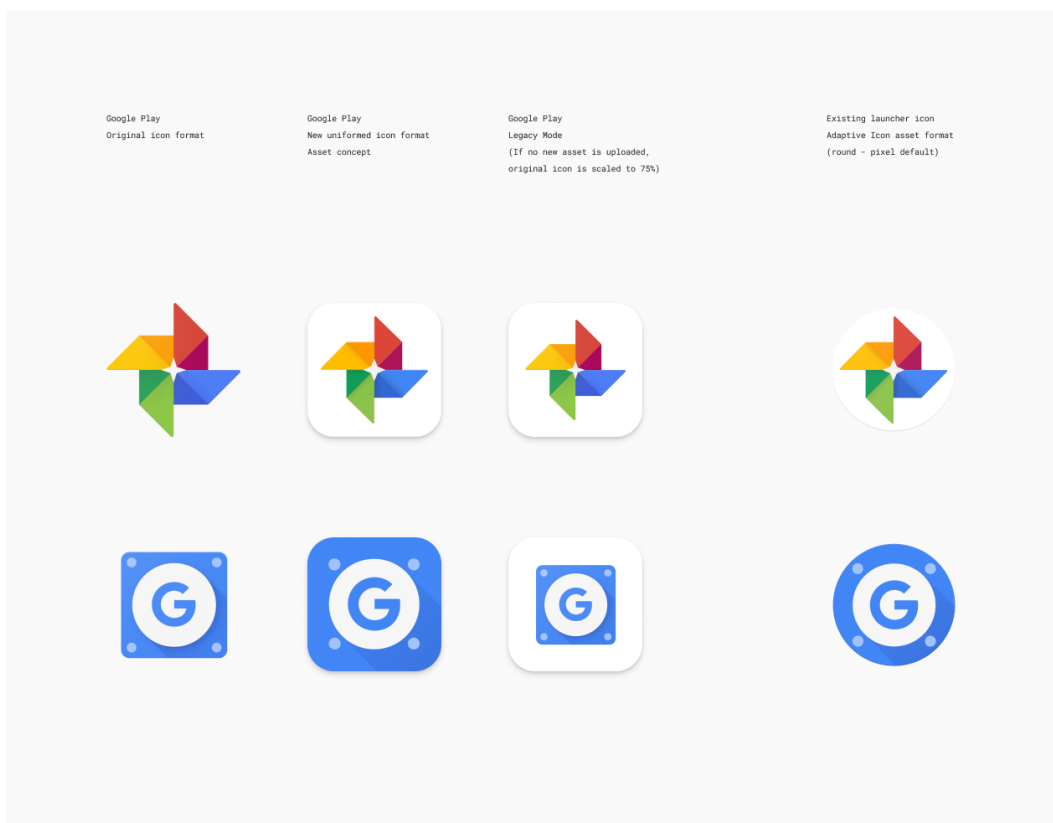


Figure 06. Example part of Google template for creating launcher icons

After the graphical assets were done, I created a new build and deployed it to App Store & Google Play.

I emailed the client and they activated again and came to the realisation that the beta on Android is accessible by anyone even though we had mentioned it multiple times previously. This followed with a lot of emails saying that we need to restrict the access to the service and that the deletion of users needs to be available for the admin panel. From the development side we were against the hard deletion of users in the database because it would compromise the database integrity and cause breakages in the service.

There are a lot of features that the users can use and most of them leave a trace in the service, which would link to non-existing data if the user would be deleted from the database. We had a case of the app crashing half a year ago, when one of the users was hard deleted and they had created a lot of content in the service, so we had experience with hard deleted users. We mentioned that we can handle user deletion in a way which would leave no identifiable traces in the database by nulling all the user data, while keeping the user object in the database to preserve database integrity. The client then responded by saying that they will look into the GDPR and how the user information should be processed in case of deletion, so we can continue with our current tasks at hand. The last response came at the end of the day, so it was time to wrap up the week.

Weekly Analysis

This week's duties seemed to be providing tech support in testing sessions, attending meetings with the client and the various stakeholders and creating new functionalities for the app. I've found that with these meetings I am mostly there as a backup to answer any technical questions and specifics of how something works. My inputs are usually only brushed over and forgotten quite easily.

While the client is more well versed in their business, they quite often seem to stumble when it comes to technical things, which is when they turn to me for help. While I try to explain things very simply without using too many technical terms, they still seem to not understand what I am explaining. Most notably this week it was the email asking about the difference between development version and the production version. I explained that the development version is used for developing the service, testing out new features and it not being stable at all times due to the constant changes and that production version is the stable version that is given out to potential users. It was also not clear that the development and production versions have completely different databases, resulting in confusion as to why some items were not visible on production version while they were in

development. My approach to this has been to explain things as clearly as possible in a few sentences and providing links whenever possible to give out clear differences between things. In this version confusion I provided the client with two URLs to the different versions to make the difference clear.

One other case regarding technical understanding this week was the broken search in the app. The client had gone through the trouble of individually writing down which search terms work and do not work. Luckily, we fixed the issue while they were providing us information about which terms work and do not, so they did not spend too much time on it. We decided with my boss to not explain the search functionality in-depth to them and that we would elaborate on how it works in case they asked.

While the client had some gaps in their technical knowledge, I noticed that I had that as well once I started working on a new API endpoint. It took me some time to figure out how to create a simple getter function for an API. The schema creation part was simple for me, since I could use already existing schemas in the code as a reference, but the functionality was under layers of abstraction which I had never seen before. Abstraction hides the details in the code and allows tackling problems on a “higher” (abstract) level. (Eloquent Javascript, 2018) My progress with API was slow and I did a lot of guessing before getting it right and working. Once I had it working, I understood what works and why.

My development process seems to lack the usage of tests as well. Quite often I make changes, test them out, see that everything works as intended, create a pull request, wait for either my boss or the full-stack developer to review it and then merge it. I forget to run the tests after making changes. This is something I should keep in mind, since it results in more time spent on writing code.

This week also included some copywriting and visual design. Most of the new things this week were the Google Play Store strict requirements and their various different guidelines. The app’s store page is not the most fleshed out at the moment and there are a lot of fields that need filling out. I do not know what information the client wants to put there, so I am going to have leave it for them. But what I do is create the graphical assets for the app store and write some general metadata which should pass the minimum requirements for app approval.

3.2 Week 39

Monday 23 September 2019

Today's tasks are to continue work on the Parameters API, address an email received on Sunday and rehearse for Wednesday's Digital Prototyping Workshop.

We had received an email on Sunday about receiving "Request Validation Error" when saving an item to own collection in the app. The error was caused by trying to add an already saved item into the same collection, saving a duplicate into the collection. The error message was not specific on the frontend, so the client was not able to figure out the reason. We handed the task to the full-stack developer so he could look into it, since preventing duplicates should've already been implemented in the backend.

During the day we practiced our presentation for the Digital Prototyping Workshop at The Shortcut, which was being held on Wednesday. Adrian was the presenter and I was doing a live rapid prototype on Figma. We went through the presentation, discussed about some tweaks for it and I practiced the rapid prototype we were going to create on the spot.

I received an email requesting some semantic wording changes to the app, so I linked the google document we use to update the localizations in the app and service and told that all changes can be done directly there, to lessen communications overhead. The client can have their internal discussions about the tone of voice, make the changes in the document and we'd just pull the changes when creating a new build.

Localizations in the app and web service are updated with a script that parses the locales from the google spreadsheet and used as a component or pure string in the app.

```
68 <View style={styles.section}>
69   <Msg style={styles.sectionTitle} id='profile.my-intro'>My Introduction</Msg>
70   <Text style={[constants.regularText, { marginVertical: 4 }]}>
71     { user.about || <Msg id='profile.empty-about' style={{ color: colors.grey.deep }}>Write something about yourself :)</Msg> }
72   </Text>
73 </View>
```

Figure 07. Msg localisation component used in own profile

```

7 import en from '../..//locale/en.json'
8 import fi from '../..//locale/fi.json'
9 import sv from '../..//locale/sv.json'
10
11 const locales = {
12   en, fi, sv
13 }
14
15 const renderData = (str, data) =>
16   Object.keys(data)
17     .reduce((str, key) =>
18       str.replace(
19         new RegExp(`{{\\s*${key}\\s*}}|%${key}%`, 'g'),
20         data[key]
21       )
22     , str)
23
24 const translate = (language, id, value, data) => {
25   const translation = (locales[language] && locales[language][id]) || value
26   return data ? renderData(translation, data) : translation
27 }
28
29 const getLang = async () => {
30   const locale = Localization.locale // await Localization.getCurrentLocaleAsync()
31   const language = locale.replace(/-/ , '_').split('_')[0]
32   return { locale, language }
33 }
34
35 const Msg = ({ id, children, data, condition, uppercase, ...props }, { language }) => {
36   if (!id) {
37     console.warn('Msg missing id', children)
38   }
39   if (condition !== undefined && !condition) {
40     return null
41   }
42
43   const translation = translate(language, id, '' + children, data)
44   if (!translation) {
45     console.log('Missing translation', language, id)
46   }
47   const text = translation || id
48   return <Text {...props}>{ uppercase ? text.toUpperCase() : text }</Text>
49 }
50

```

Figure 08. Frontend part of the localisation process in the app excluding data parsing.

After some while the client had updated the wording in the spreadsheet, and I pulled the changes into the project and continued on the Parameters API since I had not touched it on Friday.

I looked into the query and had issues with passing proper data through the API call, so I was trying to debug the cause. I was not receiving any error messages, only *undefined* through the output. After spending some time trying to blindly change different parameters in the query, I decided to leave it for tomorrow, since I was not being productive.

Tuesday 24 September 2019

Today's objectives are to finish the Parameter API on the web, address an email regarding Apple Push Certifications and practice for the DPW which is tomorrow.

I started the day with where I left off yesterday, working on the Parameter API. With fresh eyes I could focus better on the quality and logic of code. I quickly came to realise I had been passing a whole object as the query, instead of deconstructing it properly where I could pick the parameters passed (*function(q) was supposed to be function({ q, type })*). Destructuring is an expression that unpacks values from arrays or objects, into distinct variables. (MDN 2019a) After that I modified the model of the Parameter API and tested it out and it worked. With the logic working on creation, delete and search I cleaned up some of the leftover code on the backend and started drafting the UI for the web parameters.

```
9  const ParamSchema = new Schema({
10     name: { type: String, required: [ true, 'Param cannot be empty' ] },
11     value: { type: String },
12     type: { type: String, enum: ['tag', 'skill'], required: [true, 'Param type required'] },
13     language: LangString
14 })
15
16 const getParamSearchQuery = q => ({
17     $and: [
18         {
19             $or: [
20                 { name: new RegExp(q, 'i') },
21                 { type: new RegExp(q, 'i') }
22             ]
23         }
24     ]
25 })
26
27 ParamSchema.statics.search = async function ({ q, type }) {
28     const query = q ? getParamSearchQuery(escapePunctuation(q).trim().split(',').join('|')) : { $and: [] }
29     if (type) query.$and.push({ type })
30     if (query.$and && query.$and.length === 0) delete query.$and
31     const results = await Param.find(query)
32     const count = await Param.count(query)
33     return {
34         results,
35         count
36     }
37 }
```

Figure 09. Finished Parameter Model and search query

Doing the UI was mostly just adhering to the current structure of the Page component on the web, so it did not leave much room to experiment to get it done in a short time. I drafted some UIs and played around with the placement and UX of it all.

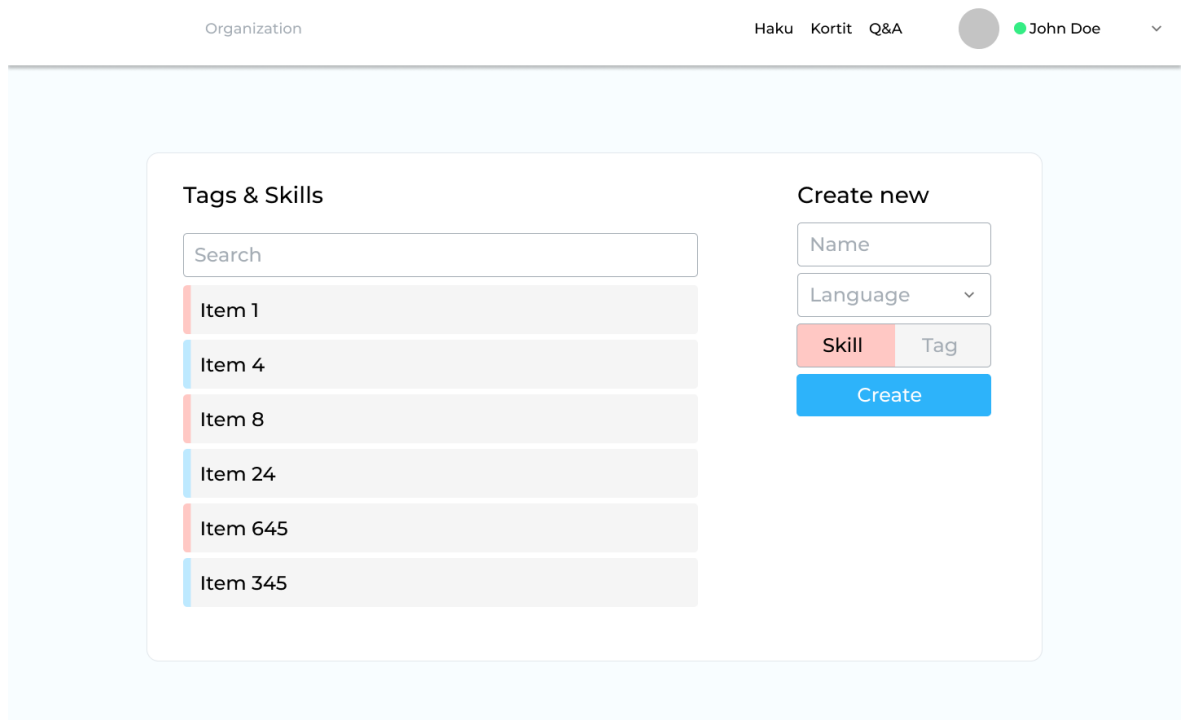


Figure 10. Figma UI draft for the parameters

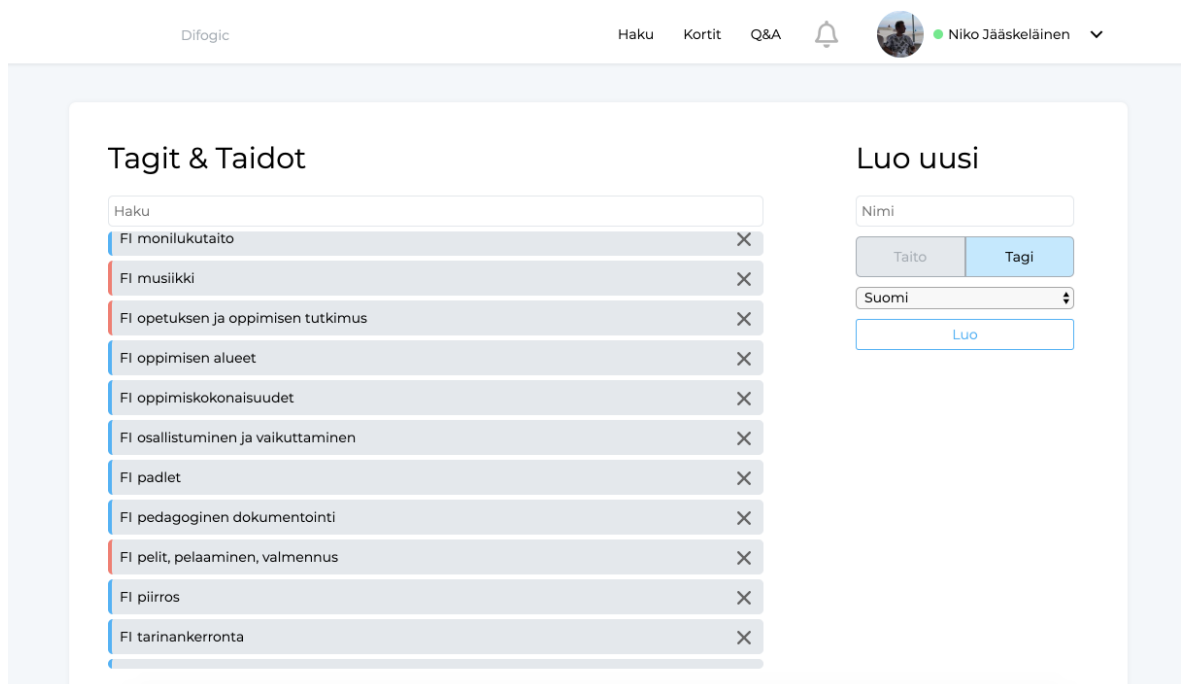


Figure 11. Actual result

I spent a few hours on the front-end of the Parameters and finished it relatively quickly once I had the Figma draft as reference. I thought I'd have to spend more time on the web, but since it was finished so early, I added support for the parameters on the mobile app as well.

```

Niko Jääskeläinen, 7 days ago | 1 author (Niko Jääskeläinen)
1  const { Param } = require('../db')
2  const { userHasRole } = require('../lib/util')
3
4  const { NotAuthorizedError } = require('../lib/errors')
5
6  const getParam = async (req, res) => {
7    const param = await Param.findById(req.params.id)
8    return param.toJSON()
9  }
10
11 const searchParams = async (req, res) => {
12   const { query } = req
13   const params = await Param.search({ ...query })
14   return params
15 }
16
17 const createParam = async (req, res) => {
18   const data = req.body
19   const user = req.session.user
20   const isAdmin = userHasRole(user, 'admin')
21
22   if (!isAdmin) return new NotAuthorizedError('Admin role required for creating parameters')
23
24   const param = new Param(data)
25
26   await param.save()
27
28   return param.toJSON()
29 }
30
31 const deleteParam = async (req, res) => {
32   const param = await Param.findOne({ _id: req.params.id })
33   const user = req.session.user
34   const isAdmin = userHasRole(user, 'admin')
35
36   console.log('Deleting param', param)
37
38   if (!isAdmin) return new NotAuthorizedError('Admin role required for deleting parameters')
39
40   await Param.deleteOne({ _id: param._id })
41   return { message: 'ok' }
42 }
43
44 module.exports = {
45   getParam,
46   searchParams,
47   createParam,
48   deleteParam
49 }

```

Figure 11. Param API with some leftover code

After the web and mobile support for parameters were done, I looked into an email which was about expiring push certificates for Apple. The technical manager from the client's side had forwarded this email to us and I responded we'd look into it. After some quick googling I created new certificates through the Expo build process and emailed the client that it was done. With a new iOS build on the way I also made a new android build with a few different changes.

After the builds were published and processing in their respective app stores, we practiced for tomorrow's digital prototyping workshop the rest of the day. I went through the live prototype a couple of times to get the flow for it and we discussed if the presentation needed some fine tuning. Some thoughts were exchanged, and we agreed it's good to go.

Wednesday 25 September 2019

Today's objectives are to find out why the dev version was not accepted, to find out the reason for the consumer version leak and do the DPW.

Google Play had processed the new APK files in the evening and rejected the latest update while iOS version updated with no issues. I emailed the client to get the message Google Play had sent them and received the email quite promptly, which made pinpointing the issue simple.

I had made a slightly different build for android which was using the fork of the react native WebView API, which the official RN docs even suggest using. The new WebView API has fixed the Apple's deprecated UIWebView API issue, but apparently it conflicts with Google's Terms of Service if the WebView is using YouTube, which the content in the service does use.

I looked into the fix of this and found there was an open pull request which addresses this issue, but it had not been reviewed and merged yet, so there was no way of implementing it in the project in an easy way. I discussed with my boss what we should do, and we decided we'll use the old WebView for now and look at it when the Apple issue is more urgent. I sent an email to the client saying it's fixed with the next build.

After the WebView issue I looked into the reason of the consumer version leak. Since the client had described what they were doing when it happened, I quickly noticed that in a collection the delete function takes the user to the homepage of the enterprise version. It was hardcoded there, and we currently did not have any "active" high level way of checking the user's roles and doing the navigations accordingly. I fixed the code and went through other possible locations in the code where this could happen.

After the code fixes we head to The Shortcut to prepare for our workshop. Adrian was facilitating it, being the presenter and voice of the workshop. I did the live prototype and

after that we all chatted with the workshop attendees and helped with any questions they might have had. We stayed there until 19:00 when the workshop ended.

Thursday 26 September 2019

Today's objectives are to look for bugs in the app and create new builds for android and iOS so it's stable for the new incoming testing session.

I started the day with fixing some known bugs which had not been much of a priority. One of them was the users collections not updating properly after one of them is deleted. I noticed this yesterday when I was working on the consumer leak fix. The collections were saved in the store of the app, so the fix was a matter of updating the store right after the deletion of the collection.

One other bug I came across was the `followedTags` of the user not being saved and/or visible after the onboarding process. I debugged the issue and noticed that the user is saved in the database, but the store is not updated in the app. The whole file was missing Redux from it, so the issue was very obvious in the end.

After those I scoured through the app to find any logic issues or UI breakages, overall a very uneventful day. No major communications with the client.

Friday 27 September 2019

Today's objectives are to maintain high levels of communication with the client and making sure the testing of the consumer version can be kicked off without issues.

I had a lot of email exchanges with the client walking them through the process of using the App Store Connect testing panel and wrote instructions which they would pass to the testers.

Some of the app testers had to be added to the App Store Connect users so they would get access to the newest build immediately, since the public beta version was still being processed by Apple. This is not really a good practice, but the client needed the testers immediately for some reason so I added the people there so they'd have access.

No code changes were made today.

Weekly Analysis

This week's highlight was the Digital Prototyping Workshop which we were facilitating at The Shortcut on Wednesday. Other themes were working on the Parameters API and ensuring the stability of the next releases of the app.

This week I finished the Parameters API and it seemed to make more sense after I had rested properly and came to work with a fresh mind. This allowed me to be relatively efficient in its creation and understand the logic behind it. I worked on the API without asking for help from my boss, since I thought he's busy with other matters regarding a project he's working on and planning the DPW day as well. While I could've just asked for help from him, I felt like I need to figure it out on my own to develop my problem-solving skills and general understanding of programming logic. After getting over the big obstacles of passing and getting data properly through the API the rest was simple and quick to do. I really enjoy doing UI wireframes since Figma is a really good and intuitive tool for that purpose.

While I enjoy creating new UI's, at times it seems like I do not know what is the "right" way of structuring the interfaces. I understand general visual theory, grouping of elements and colours etc. but the user experience causes me sometimes to stop and think if I should be even doing the thing I am doing at the moment. I had to ask my colleagues multiple times what they think of the wireframes and since one of my colleagues' role is a UX designer I value their opinion a lot in this area. Having people give confirmation to my designs, be it code or visual, makes me feel like I'm on the right track and that I do know these things. This helps to battle the impostor syndrome. According to Clance and Imes impostor syndrome is the false belief that one's successes are the product of luck or fraud rather than skill (1978). This feeling was very strongly present in the DPW this week.

We rehearsed for the prototyping workshop during the the week to make sure it would go as smoothly as possible. We were representing our business venture Perspektives, which has a business model around rapid prototyping, so we wanted to make an impression that shows that we know what we do. Our workshop structure consisted of a presentation of a general explanation of prototyping, how we work and how it can be done efficiently. We use Figma in our daily work, so we incorporated it into the presentation and created a small template for the attendees to work with when they start using Figma. At the end of the presentation we created our own small prototype app with the template, which was a copy of Instagram just to show how a simple looking app can be done with existing Figma

components. I was doing the prototype on Figma and describing my steps and how the various functionalities worked. People were eager to use the tool and we received a lot of questions regarding prototyping which seemed quite basic, but afterwards we came to the realisation with Adrian that we've learnt a lot in the time we've worked at the company, alleviating the imposter syndrome greatly.

In addition to the prototyping this week I've been fixing a lot of bugs. I do not know whether the way I write code for this project is sloppy or these are just humane errors. For example, what I thought to be a "one-liner fix" in the backend last week resurfaced again in the frontend and the cause for it was obvious, in hindsight. The file was missing Redux, which the functionality depended on. I need develop my skills in tracing the root cause of bugs and making sure there are no side effects left from them.

3.3 Week 40

Monday 30 September 2019

Today's objectives are to address the new people to be added to the beta testing on iOS through TestFlight and check that everything is properly set up, because of some test access issues.

I started the day with removing the just added beta testers from the internal test group and sending them an email regarding the link through which they can access the beta. App Store had processed the public beta and approved it, so I decided it'd be better to keep the beta testers away from internal project rights. I received an email about adding new people to the test group to which I responded that the new people can access this by clicking on the link which I provided in the response. I CC'd the people to be added and half an hour later the client sent me an email that she had given me the wrong email address. Because the client did not want to give the beta link to outsiders, I had to create a new public beta access link and render the existing one useless. After I had done that one of the actual beta testers emailed about the link not working and I responded with an apology and giving out the proper link. I was doing the clients' communications for them for some reason.

After the whole confusion about the beta access I checked the App Store Connect page and did some clean-up there, removing old testers and making sure all the appropriate builds are available for testing etc.

With all the beta testing issues I also checked the production version logs, because I received an email about some bugs on login. The issues ended up being from the user side, so it required no action. There was also a small “leak” from the consumer version to the enterprise version which I fixed. Problem was caused by the “magic link” login which did not check the users’ role, so it navigated them straight to the enterprise home screen.

I also messaged my boss asking if we could have a one-on-one meeting tomorrow, because I was not too happy about the lack of direction we were currently having in the company regarding new ventures.

Tuesday 01 October 2019

Today’s objectives are to express my concerns about the state of the company at the moment with my boss and hopefully get some action points from that.

Is started the day with the one one-on-one with my boss. I expressed my concerns about how we’re lacking direction, structure and focus at the moment and he was aware of this fact. Apparently, there were some personal matters going on in the background, so he was aware of it affecting our general work motivation. We also discussed about what we want to do in the future with the company and how we are going to approach it. Our company has its own venture which we’re pursuing at the moment and it is not going forwards as expected. I also mentioned how having weekly meets would be probably beneficial so we’re all onboard with what’s going on at the moment, because it felt like we’re just doing our own things separately. We agreed that I can start with facilitating these weekly meets, because my boss would most likely forget them and be occupied with other matters when we should have them. The meeting allowed me to get some things off my chest and it felt better to continue working. I know that changes do not happen immediately, but now I had the motivation and goals to make the changes happen.

After the meeting we had a call with our full stack dev about starting on app tests for the project, we discussed which critical functions should be covered in the initial tests.

Wednesday 02 October 2019

Today's objectives are to address emails from the Helsinki University testing and other small changes regarding the project.

Apparently the testers at Helsinki University had some issues with returning tasks in the app so I started looking into that. I responded to the emails which I was CC'd into by saying I'll look into what's happening. This was met by a response from the HU contact person that they did not want to cause any extra work for the developers and that they'll be putting off the usage of the service until January. I'm 100% certain the client was not too thrilled about this so I was ready to receive a lot of heat in the upcoming days.

I continued into looking at what is the issue and found out that the logs were reporting an error with indexing in MongoDB. I had come across this issue earlier as well, when I added another recipient to a task manually in the database, so I thought that was the cause. Now that there were no modifications to the tasks, I needed to look into it deeper. I waited for my boss to come to the office so we could look at it together. Turned out the issue was with old indexing in the database. Task recipients used to be an array of ObjectIDs and they were indexed in the database. The task schema was restructured in the summer for better support and it still had the recipients being indexed in the database and now it was an array of objects instead of IDs. This was the cause and it was fixed by simply deleting the recipients index from the database. Of course, it was too late for the HU testing group and this error was an anomaly which we could not have predicted happening. As soon as this was fixed, we emailed the client letting them know about it.

Besides the task issue, there was a request for shortening the required password length for the service, currently it was 12 characters. I changed it to 8 characters and let the client know as soon as I deployed the change, it was a simple 1-character change in the backend code.

After some while again I received an email about another found consumer version leak into the enterprise version, so I had a look around the code what the cause was. It was an issue with link handling in the App.js file, so I added some role checking there as well.

Thursday 03 October 2019

Today's objectives are to write our weekly newsletter and to answer any emails from the client and act upon them.

Our company has a small newsletter which focuses on product management, product development and other things product, so we gather most interesting links of the week and send out a newsletter every Friday. Since my colleague did not have any time for the newsletter and I did not have anything urgent on my task list, I was collecting links and writing the copytext for it. This was a quite straightforward day, searching the web for interesting articles.

Friday 04 October 2019

Today's objectives are to plan our marketing strategy for our prototyping venture and to create advertisements for it.

The day started off with a workshop where we discussed who we are going to target with our marketing for our prototyping venture, Perspektives. The idea behind the venture is creating rapid prototypes to validate ideas without creating a large team or project behind it. The simplified version is "why spend 100K when you could spend 10K to see if it actually works", low cost risk elimination.

Our plan is to target small to medium sized businesses which do not utilize technology to their fullest potential, for example automation of time-consuming manual tasks prone to human error. We're going with the assumption that this is a blue ocean strategy, because we previously tried to target people who are in the IT field, tech savvy and understand the benefits of using technology in daily tasks.

After we decided on the target audience, we discussed what sort of advertisements we want to create and what would catch the attention. We had no idea what works, so we decided to try out multiple different types of ads and to run them for a while to see which ones get clicks.

We spent the rest of the day creating multiple different type of advertisements and coming up with various taglines and copytext for the ads.

Weekly Analysis

This week's theme was feedback, bringing up personal concerns about the direction and decisions made in the company lately and general setbacks.

My work had lately been shadowed by a general lack of direction inside the company, there were so many things to be done at the same time, things being put on hold and never coming back to them making it feel like the whole business is just a mess. No one really knew, or communicated what others had on their to do lists creating this strange unstructured feeling inside the company. Each employee could be doing their own thing without anyone knowing about it until a week later when it comes up randomly in a water cooler conversation.

I asked for a one-on-one meeting with my boss to bring these issues to light, since it was causing a serious lack of motivation. He was open to feedback and shed some light on why he also has not been the most focused at work. We talked about the current situation of our company, our goals and practices at the moment. I suggested that we could have weekly meetings to have some sort of structure and to create a feeling of cohesion in the company. He was completely in favour of this and said I can facilitate the meetings, since he said he was most likely going to forget them.

In addition to things not going as well as they could inside the company the same was happening with the client. One of the pilots decided to drop out due to an unexpected bug, which I am sure did not make them too thrilled about it. Even though there was no realistic way to find out the issue in time with low usage, I think the client might want to blame us for it, but that's yet to be seen. The project has gone through so many changes in a short span of time so there are bound to happen some sort of issues.

What I found really interesting though was that the public has a very low tolerance for technical instability in a service. Any sign of a feature not working right out of the box they are very likely to drop it, at least from what I've seen so far. One pilot program has been very patient with the project, but the rest have been slightly annoyed by the issues. While we're doing our best to keep it as stable as possible it is hard to keep up with the wishes of the client since they seem to have so many of them. The feature requests and lack of specifications causes us to spend a lot of time drafting specifications for which we do not always have the big picture. Sometimes the specifications are done in the way we see best, from a technical and business logic side, but the client might have something completely different in mind, causing an increase in development time.

3.4 Week 41

Monday 07 October 2019

Today's objective is to attend a meeting with the client at a Vantaa kindergarten and to gather action points from that.

I started the week with going to the Vantaa kindergarten for 12:00 and joined the client in their meeting with the teachers and Vantaa lead. Once I arrived at the kindergarten, I was told that I can take notes and that I did not need to participate in the general discussion in the beginning. There was discussion relating to current issues in the teachers' daily work and shortcomings of current solutions. I took some notes of what was said in the meeting and how they could be translated into the service.

The whole day was spent at the kindergarten and I got zero action points from it. I did discuss with the client about the current situation and how things are going with the project after we left the kindergarten. In general, quite an uneventful day for me.

Tuesday 08 October 2019

Today's objectives are to pick the ads we want to display on our ad campaign, facilitate the first weekly meet after the one-on-one and to check my inbox.

I started the day with the weekly meet, because I was away the whole day yesterday. I had created a small structure which we'd go through. It's as simple as what was done last week, what's upcoming this week, what do we want to achieve this week, possible pain points and the current status of prospects. We went through the points, so everyone would have a general view of where we are at currently. It seemed to be a good idea, because it felt like we can have some structure to our work.

After the weekly meet, we went through the ads which were created on Friday. We decided to go through each one and simply vote yes or no to make the process of picking ads as efficient as possible. No justifications needed, if the majority – 3 out of 4 - voted yes then the ad was selected to the campaign. If it was 2 yes, then the "creative director", our user experience expert, decided whether the ad would be chosen.

After the ads were chosen, I received an email about content visibility in the service and got asked to copy card content from one organization to another. I thought this is not the

best way of doing it because it polluted the database and I knew that it would confuse the client (who was asking for the content to be copied) even more. I decided that it would be smarter to create a function of determining the visibility of the card content in the service, so I discussed with my boss what would be the smartest way of implementing it. We decided on adding a field to the existing card schema, which would have an array of ObjectIDs. If the organization ID would be there, it would be visible in search inside the organization. I started working on the API for the visibility for the rest of the day and emailed the client that they would be able to set the visibility themselves soonish.

```
232 CardSchema.methods.addVisibility = async function (orgId) {
233   if (this.visibility && this.visibility.includes(orgId)) {
234     this.visibility.pull(orgId)
235   }
236   this.visibility.push(orgId)
237
238   return this.save()
239 }
240
241 CardSchema.methods.removeVisibility = async function (orgId) {
242   if (this.visibility) {
243     this.visibility.pull(orgId)
244   }
245
246   return this.save()
247 }
```

Figure 12. Card schema methods for visibility

Wednesday 09 October 2019

Today's objectives are to work on the visibility API.

Today I worked on the visibility API, figuring out how to pass the correct parameters once again. I also received an email about "cleaning up" the chat view of the app, the client wanted to change it a bit, so I took notes of what improvements they wanted to make.

```

169   const adminSearchCards = async (req, res) => { // search for cards that have the passed
170     const { query } = req
171     const organization = query.organization
172     console.log('organization', organization)
173     const cards = await Card.search({ organization })
174     return {
175       results: cards.results,
176       count: cards.count
177     }
178   }
179
180   const adminCardsToOrg = async (req, res) => {
181     const org = req.params.id
182     const { card } = req.body
183     await (await Card.byId(card._id)).addVisibility(org)
184     return { message: 'ok' }
185   }
186
187   const adminRemoveCardsFromOrg = async (req, res) => {
188     const org = req.params.id
189     const { card } = req.body
190     await (await Card.byId(card._id)).removeVisibility(org)
191     return { message: 'ok' }
192   }

```

Figure 13. Visibility API

```

347     if (organization) {
348       query.$and.push({ $or: [{ organization }, { visibility: organization }] })
349     }

```

Figure 14. Card search change to support visibility

Part of creating the visibility support was to make it backwards compatible, so it would not need any changes to the web frontend and more importantly mobile app. After some testing, making sure it works as intended, I moved on to the web frontend.

Thursday 10 October 2019

Today's objectives are to continue work on the visibility API.

I had done the basic frontend UI for choosing the card visibility and were able to select and unselect cards which are visible in an organisation in the database. I had quite a lot of troubles updating the UI accordingly. I managed to make the UI update on selecting a card and showing that it is visible, but when unselecting it, it did not change the UI and I just could not figure out what was the issue. I had been looking too closely at the code for a flaw in logic, that I did not produce any results. The whole day was spent on troubleshooting what possibly could be the issue.

Friday 11 October 2019

Today's objectives are to hopefully finish work on the visibility API.

After yesterday's problems, I decided to distance myself from the function that does all the logic and look at the new code from a high level. I realised I had been checking the visibility in the wrong place, directly from the DB and not keeping the UI updated accordingly. I needed to save the visible cards in the state and then render them properly. I also was checking the wrong logic when selecting the cards, I was modifying the state according to the visibility of the card and checking the organizations existence in it and not the state where all the real-time changes were happening. With pinpointing the issue, I refactored the code and it worked. I was very relieved, but at the same time extremely annoyed, because I was looking straight at the issue for a long time and wasted so much time on it. My work performance was not the best this week.

After I got it working properly, I tested it out multiple times and deployed the changes to development server. I emailed the client saying that the dev version now has this functionality and that they should test it out and see that it does what it's supposed to. If there are no issues, I will deploy it to production on Monday.

```

145   onAddCardToOrgPress = async e => {
146     e.preventDefault()
147     this.setState({ visibleCards: [] })
148     const { selected } = this.state
149     const organization = selected._id
150     const { data } = await api.adminSearchCards({ organization })
151     const cards = data.results || []
152     this.setState({ visibleCards: cards })
153     this.toggleSelectOrgCardModal(true)
154   }
155
156   toggleCardInOrg = async card => {
157     let { selected, visibleCards } = this.state
158     const id = selected._id
159     const exists = this.checkSelected(card)
160     console.log('onClick', 'card', card._id, 'exists', exists, 'in org', id)
161     if (exists) {
162       visibleCards = visibleCards.filter(c => c._id !== card._id)
163       await api.adminRemoveCardsFromOrg(id, { card })
164     } else {
165       visibleCards.push(card)
166       await api.adminCardsToOrg(id, { card })
167     }
168     this.setState({ visibleCards })
169   }
170
171   checkSelected = card => {
172     const { visibleCards } = this.state
173     return visibleCards.some(c => c._id === card._id)
174   }
175

```

Figure 15. Web frontend functions with logic for toggling visibility

Weekly Analysis

This week's theme was creating a new API and autonomy.

This week I facilitated our first weekly meet. It seemed like we have now some sort of structure and know where each person stands with their current workload. This allowed us to plan our next steps and track our own progress, maybe even have some proper KPI's in the future. Key Performance Indicators (KPI) are values used to measure effectiveness in reaching strategic and operational goals. (The Balance Careers, 2019)

The week was focused on creating a new API toggling the visibility of card items between organizations. This was started by a client asking through email if I could copy some content from one organization to another. Since this process literally requires for us to copy data inside the database, I suggested to my boss that we should create a function in

the service which enables toggling the visibility of an item inside the service. He supported the idea and we discussed the specifications of how the data structure and how it should work. After we decided on how it should work in the backend, I started creating it.

What I realised this week was that I have a lot of autonomy in the company, my ideas are taken into account and that I can discuss them with my colleagues freely. I feel like I might have taken some of this freedom for granted and not realising it when having a hard time at work. I seek support and validation from my colleagues and the results are never met with a completely negative response. Of course, there is feedback on how I do things, but it is constructive and never malicious. From the feedback and discussions, I pick up on best practices and other viewpoints which I try to take into account when working.

3.5 Week 42

Monday 14 October 2019

Today's objectives are to address the received emails over the weekend.

The marketing person from the client had emailed asking what the issue with posting announcements on the app is, so I looked into the server logs. Turns out the user trying to post an announcement was not assigned as an organization admin, and to post announcements the user must have the role admin and be assigned as an organisation admin. I added the user into the organization admins and responded to the person with explaining the two required roles. He had a few more questions regarding this functionality and I elaborated on them and giving out step by step instructions as to what works how.

The client had also collected feedback from the consumer version testing during the past weeks and there were a few points I could immediately change. One of them being adding a simple clear button to search bars. Others were some networking related issues, which required timeouts in the app, it's currently completely dependent on there being network access. I made a note of this and mentioned it to my boss, and he said we'd look into it a bit later.

After going through the changes, I received more emails in the evening regarding some notification bugs. Apparently, notifications were being received by wrong users as far as I understood. We speculated with my boss that it has something to do with how the push tokens are created and that they're not cleared when the accounts are switched on the

device. This was to be taken under work once the full-stack developer had finished the mobile app tests.

Tuesday 15 October 2019

Today's objectives are to translate the landing page aimed at small businesses into Finnish.

Our boss had finished the landing page for our prototyping venture in English and deployed it and now he wanted for it to be translated into Finnish. The advertisements campaign was to be aimed at English and Finnish businesses, so I started on the translation in the morning. Surprisingly, it took me the whole day, because the terminology and general tone of voice was quite hard to translate into Finnish. If I had finished the web page today, I could've moved on the ads, but since it took me so much time, I left the translation of the ads for tomorrow.

Wednesday 16 October 2019

Today's objectives are to finish translating the ads for the small business campaign and structure our workshop for tomorrow's meeting with a big client.

I started the day by going through the advertisements and translating them. This was a more straightforward task, because I only had a dozen of taglines to translate and slightly reword to make them seem more natural.

After the ads were done, we started drafting the structure for our meeting and workshop which we would do tomorrow with a big client. The general workshop structure is to map out stakeholders, do a vision canvas modified for our purposes, map out the problem solution space and then assign the solutions on a risk impact matrix.

The project is perfect for us, because it will be defining whether our business idea of Perspectives, rapid prototyping, works and if we make a good impression, we might be able to get the whole project after it.

We worked on and rehearsed for our meeting for tomorrow for the rest of the day. We found it important to make a professional impression.

Thursday 17 October 2019

Today's objectives are to perform good at the prototype meeting.

At the start of the day, we went through our presentation and realised it does not work that well in the structure we did yesterday. We had already received some information regarding the prototype and had some initial stakeholders, so we decided to restructure the workshop so that it would consist of three parts, stakeholders mapping, vision canvas and risk impact matrix if there would be left any time. We decided the new structure works better and rehearsed it a bit at the office before heading to the meeting.

The meeting went relatively well, and the workshop structure worked well, even though we ran out of time to fill in the vision canvas properly. At least we managed to map out a large pool of stakeholders and arrange them by order of importance, so we'd know whose needs would be addressed first in the initial prototype.

Friday 18 October 2019

Today's objectives are to have a video call with the existing client and do some research regarding the new prototype.

I started the day with having a scheduled video call with the existing client regarding their project. We went through some items on the agenda and got action points on what to develop next. Now that our full-stack developer should have the tests ready by Monday we can move on to other things.

Action points from the meeting were creation of public and private collections in the service, consumer web UI, adding of people to a group chat after it's created, task recipient modification, ability to add skills to the cloud from mobile edit profile screen, deletion and modification of questions on mobile, a bug with repeating characters in controlled inputs, admin view for managing marketplace cards and fluent organisational switching for admin accounts. The action points were ranked by priority and that gave us a clear direction on what development work we would start on next week.

After the meeting and lunch, we started on going through yesterday's meeting and started processing all the available information we had received regarding it. There was a lot of

research already done, so we were reading through the documents and links and transforming the written notes into digital format. Rest of the day was spent on reading through the information and making notes which would help us structure our work regarding the prototype. We were not in the development phase yet, there were interviews to conduct with stakeholders and the vision canvas needed going through, so we'd have more actual current problems to solve in the prototype.

Weekly Analysis

This week's theme was copywriting and meetings with existing and potential clients.

Beginning of the week was a lot of communications with the client from the existing project and addressing their notes about the current status of the service. There were a lot of small changes to be done part of which were taken under work immediately.

The work that required the largest amount of time from my side was translating the copywriting of our alternate website into Finnish. I have not done any copywriting in Finnish and have not studied or practiced it, so it was very heavy for me. As I was translating the text it could not be translated word for word, because the two languages are so different, and the general flow of the text did not work the same way as it did in English. This extended the writing time exponentially.

What I originally assumed also came to be true, that it is easier for me to write in English than it is in Finnish when it comes to technical themes. I was constantly searching the internet for proper ways of expressing certain words and concepts in Finnish, for example service design. While it is a literal translation in Finnish, I had to double check the terminology because I have not worked in the field using the language at all. Having a certain degree of uncertainty while writing something is apparently very stressing as well, which made this week very unpleasant.

Being the only native Finn in the company, I did not receive any big improvement suggestions to my writings after they were done, and we had to go with the assumption that it is good enough. I learned that I most certainly do not want to work in copywriting in Finnish.

Once the copywriting part of the week was done it was time for client meetings. First up there was an initial meeting with a potential client for a prototype which we would be working on. This caused nervousness as I did not know how to be of value in the meeting.

Our boss was doing most of the talking and I was processing all this new information and realised I have no idea how anything works. I came to understand that all of the people in the room have had over a decade of experience in their respective fields which allowed for them to have insightful and valuable inputs. I was merely a note taker.

An interesting contrast was the meeting next day, where I knew the client and I could genuinely predict what they would want to be done with the project next. With them I was relaxed, but confident. I did not find them intimidating when they would ask questions, because I knew most of the answers and I was so well versed in their project, since I had worked on it for over a year now. With familiarity and experience came confidence and I did not have that with new business meetings. I would assume that once I have attended enough of these “initial” meetings with potential clients, I will have a degree of confidence and knowledge to provide value in them.

3.6 Week 43

Monday 21 October 2019

Today’s objectives are to have the weekly meet and to start on the action points from last Friday’s meeting.

The day started off with the weekly meeting and we discussed how the work on the prototype should proceed and what we have on our to-do list. For the prototype we had a clear goal of setting up interviews with the mapped stakeholders and my colleague started with emailing the people and setting the times for interviews. I had the action points from last week regarding project 1 to work on.

I started working on the collection visibility on the web thinking it’s a simple schema field addition and a toggleable value on the collection creation and edit screen. I added a field called `private` in the schema and started working on the web frontend. I found out quickly that doing any functions referring to the schema field would cause ESLint to throw errors, because the JS was in strict mode and would not allow the usage of the variable `private` in the code. I had to change the schema field name to `visible`. After doing some work on the web I came across an issue where the checkbox would not change value even when I toggled it. After some while trying to figure out the issue I decided to come back to it later and ask my boss for advice.

I moved on to the mobile app to add supporting changes and those were straightforward, a bit of time was spent on coming up with an indicator in the UI if the deck is private or not. I went with a lock icon in front of the title if the collection was private. Only the author of the collection would see decks with the lock icon, others' private collections would not even show up in the search. After handling the UI, I needed to make sure the UI state would be updated accordingly which required some redux prop dispatching.

After doing the mobile side of private collections I asked my boss what the issue could be with the web checkbox and after some time figuring it out the solution was simply changing the inputs *checked* property to `defaultChecked`. After the change was done, I tested out the code and it seemed to work as intended with no bugs arising.

During the day I received a call from the client telling me that they had figured out what was causing one of the notification issues and they said that it was exactly the thing we had suggested but they'd shot it down immediately. The client was apologetic, and I responded by saying that they do not need to worry, now we will not waste time on trying to figure out the cause. Personally, I was not bothered with it, even if the client's tone was a bit aggressive when I had received the email them saying we're wrong about the cause.

After the collection visibility changes, I started the rework on the chat UI, because the client wanted a facelift for it and one of the action points was about adding new people to an existing group chat so might as well work on those changes at the same time. Instead of trying to reinvent the wheel, I decided to go with what works and simply use Telegram's UI as a baseline for the chat view. Biggest changes were the text input area and chat view header, other changes were adding a menu to the chat which consists of only leaving the chat. I did some small changes at the end of the day and decided to get back to it tomorrow.

Tuesday 22 October 2019

Today's objectives are to continue on the chat UI rework, add group chat options screen, and the ability to add users into a group chat

I continued with the Chat UI changes mimicking Telegram's UI, so it was relatively straightforward, just had to make sure the UI was responsive with the keyboard visible. After finishing the chat view UI, I started working on the chat options view. I realised there

were now 4 JS files all relating to chat, so I moved all of them into one folder to keep the code structured.

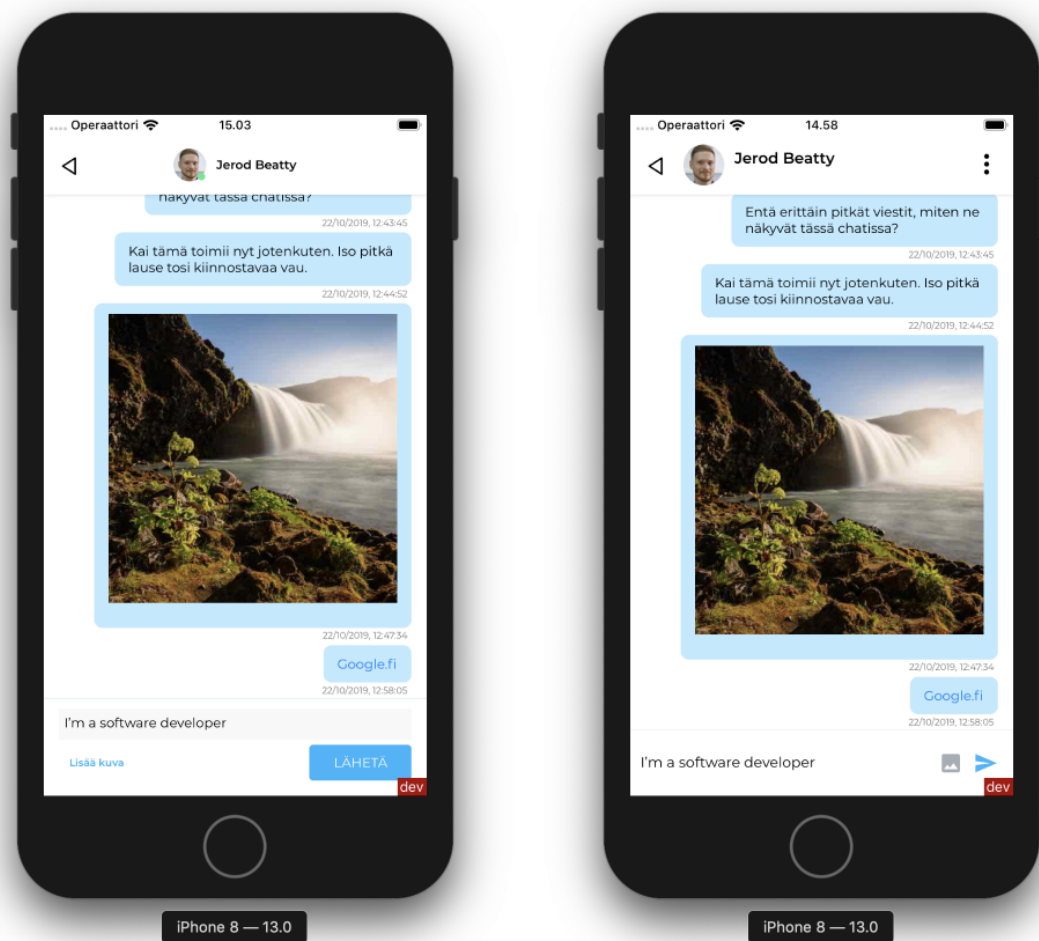


Figure 16. Chat UI before and after

The chat options view took the rest of the day, since there were no existing components which would fit the needs specifically. After getting the chat options done in the evening the next step was to start work on the API tomorrow.

Wednesday 23 October 2019

Today's objectives are to work on the chat subscribeMany API and get most of it done.

As I started my day the first thing I did was checked if the existing API would support adding of multiple users. It did not, so I reverse engineered the existing APIs and made new ones that used an array of ids as the parameter instead of one ObjectID.

Rest of the day was spent on taking notes from prototype stakeholder interviews.

Thursday 24 October 2019

Today's objectives are to figure out a bug with the new API and write tomorrow's newsletter.

Yesterday evening I noticed a bug with the `subscribeMany` API where it would add duplicates of the users into the chat if the amount was more than one. For example, if a user would add 3 new people into the chat, it would add the 3 people 3 times totalling at 9 "new" users in the chat. I went on to debug this issue with console logging every line in the backend code and it would show up as all normal. The logs did not give any indication that anything would be wrong. I asked my boss about this; he was quite busy but managed to allocate some time for the debugging and did not figure it out with the short time he had. He said let's put it on pause and he'll take a closer look at it when he has the time.

Half of the day was spent on trying to debug the issue with the chat API and the rest of the day was spent on collecting links for the newsletter and writing it up.

Friday 25 October 2019

Today's objectives are to take a look at Clubhouse and see what needs to be done for the ongoing project.

I started with going through Clubhouse which has all of our development tasks in one place in a KANBAN. There some tasks which were not actual anymore, so I archived them. After going over the tasks and leaving the necessary ones in I assigned a few of them to our full-stack developer, since he was about to finish his current tasks. I also asked him to take a look at the API issue we were having, since he might have a different approach on it.

After going through Clubhouse, I started working on the app functionality allowing the user to add skills to the skill cloud on their own, since it was a requested feature by test users. This required some modification of the existing components.

After finishing the adding of skills functionality the full-stack developer had figured out the issue with the API duplicates. We were using an asynchronous `map` function for subscribing the new users instead of a `for of` loop, as he said it is "not the most elegant

solution, but it works” so we decided to go with it. With the issue figured out the new build was almost ready for deployment. We decided to wait for next week, since deploying on Friday has caused some issues in the past.

Weekly Analysis

This week’s theme was creating a chat settings view with accompanying APIs and functionality.

I had been having some issues with redoing an UI in the past regarding the same project and tried to come up with new ways to display information. That has caused me to waste a lot of time trying to create the “perfect” UI for one single screen which most likely is not going to be the final result or even used in the end product either way. So, instead of repeating the same mistakes which I did a few months ago, I decided to go with what works and simply take inspiration from already existing (and successful) services. For the chat views, chat settings and other future views the chat might get I decided to simply copy Telegram’s UI. This gave a clear guideline on what to do and not spend time on the UI invention. This allowed for a lot faster work on the UI and most of the work was left for the other aspects of the chat.

With the other aspects of the chat I had to look into how the existing functionalities worked and tried to reverse engineer them for the new adding and deleting users feature. On a logical level it worked fine, but there was an issue with asynchronous functions which I was completely oblivious to. Asynchronous functions operate in a separate order than the rest of the code. (MDN 2019b) I ended up with finding out this issue through my colleague after I had asked him to check what is wrong with it. I still seem to lack some fundamental knowledge regarding to writing JS, but I feel like coming across these issues makes me learn and pick up on them. By doing so I would not make the same mistakes again, or in the very least be able to debug and find root causes for bugs more easily.

3.7 Week 44

Monday 28 October 2019

Today’s objectives are to create a new admin dashboard page for maintaining marketplace content and to go over the mobile changes making sure it’s stable for deployment.

I spent the whole day working on the marketplace admin page, drafting the wireframe, UI and then coding it. The process was relatively straightforward, the ideation phase and the actual UI development took the most time, even though there was already a lot of boilerplate code in place.

Biggest takeaway from today was that no matter how much already in place UI code there is, there's always some small things which require custom code.

Tuesday 29 October 2019

Today's objectives are to deploy the new build for dev and address the email I received last evening.

I had received an email from the client saying the visibility changing of cards is not there in the UI, so I went to check the issue. Everything seemed to be in order, so I sent an email to the client asking for screenshots and how to reproduce the problem, I even tried impersonating the user profile in the service before sending an email for clarifications in case it was user specific for some reason. The client responded with saying it was their mistake, they simply did not see the button and label and that everything worked as intended. After receiving this information, I started updating the localisations in the admin panel, since some of the labels and texts were not translated completely and figured that would prevent future mistakes like this.

With the localisations done I made new builds for the development version and made sure they worked fine with physical devices.

Today's events made it clear that no matter how simple one tries to make the UI there's always the element of human error.

Wednesday 30 October 2019

Today's objectives are to fix the failing mobile tests.

After merging a lot of changes to the main repository the mobile tests started failing. It failed in 7 parts, most relating to deck visibility. I started debugging the issue and went to look at the visibility parameters in the query. Most of the tests were fixed through trial and

error and took an excessive amount of time, but since there were no urgent tasks it did not affect current situation negatively.

I learned that I still need remember to run tests before merging new changes into the main trunk.

Thursday 31 October 2019

Today's objectives are to get actionable points from the meeting with the client.

The meeting with the client was at 11AM and the agenda was drafted by the client, since we had nothing on our side. We were finishing up the action points from 18/10 meeting, so this meeting was timely.

We went through the Google analytics dashboard and some modifications into what's tracked and what's not, discussed future features and possible insights the client has gathered from the users' feedback. The meeting lasted 2 hours and we got more action points to work on, so I started on them after the meeting.

Action points included comment editing in various places, small UX improvements for the chat, small UI changes, notification bugs and new notification events.

I spent the rest of the day polishing small UI changes.

Friday 01 November 2019

Today's objectives are to work on the action points from yesterday's meeting.

The day was spent on fleshing out the analytics of the app to determine which organizations in the app use what features the most, code improvements and removal of dead code from the project. Many small changes in a lot of places, very time consuming.

Weekly Analysis

This week's theme is the downtime and maintenance of a service.

After today's week I realised that the project is in a maintenance sort of phase, because it is relatively stable, but the client still has not gone forwards with publishing it. There are always some small things to change, but they do not affect the general service on the high level. It was discussed in the meeting on Thursday that it'll go live in App Stores on Monday, but I have some personal doubts about that, because the client is not even familiar with their own apps' store pages where the content is written, and the general metadata is done. The project is not gaining any momentum and the client is hesitating with going forward and even bringing up a possibility of the service's name change, which does not emanate a lot of confidence from their side. Our part in the project has become solely "the development team", because any suggestions and advice we give them are brushed over. This personally leads to a level of indifference towards the project, I still aim to do the work well, but it is not personal to me anymore. The roles of the project in everything else but the development area are not clear and it still lacks a product owner.

I started to notice towards the end of the week that we are really running out of things to do in the project, which was an interesting feeling. We'd been previously bombarded with requests and new features that overflow our work backlog and now that we were "running out of fuel" we we're scrambling to find things to work on. On a high-level picture there are a lot of things that do need work, mainly the web service, but since the client has to give us the OK on what we should and can do, we did not have anything actionable.

3.8 Week 45

Monday 04 November 2019

Today's objectives are to deploy new changes and make sure the dev and production builds work without issues.

I started the day with going through the small changes I did at the end of the last week and making sure we got all the small details the client asked for. This included the more fleshed out analytics of the app, where the users' organization is tracked on events and screen views, removal of dead code, anti-patterns and some general ESLint compliant changes, for example changing a function which uses `map` into `forEach` when it does not return anything (MDN 2019c). After there were no other visible issues, I deployed the development and production builds and waited for the mobile builds to process. The rest of the day was spent on the prototype stakeholder interviews taking notes while my colleague was running them.

Tuesday 05 November 2019

Today's objectives are to go through the projects' app stores and store presence information.

I checked the app store pages and the information was incomplete, so I emailed the client asking if the descriptions and images are ready for the app. I received a response asking which specific details the app stores need, so I sent a link to the app stores saying it would be easier if they could look at the fields themselves, since the instructions are quite specific. After a while I received an email saying if I could just tell them what needs to be done, so as I understood they could not even bother to open the link which had all of the necessary information. I got personally a bit annoyed by this, so I sent a screenshot of the page with an accompanying text "Here it is.". I received the updated texts and images in the evening and copy pasted them onto the app store pages. The texts were too long, and the image aspect ratios were wrong, so I had to spend extra time on making them compliant with the app store rules.

Wednesday 06 November 2019

Today's objectives are to fill the day with something useful.

I came to the office having nothing actual to do, so I poked around the app and figured that I could try to fix the UIWebView issue which is still persistent. I decided with removing the web view component from the app and just use an image which grabs the thumbnail from the YouTube video and has an overlay with a play button on top of it. Half of the day was spent on tweaking pixels and overlay opacity of the video links.

After midday we had some discussions with our team about the prototype. It was about the general layout of it and the user experience of the general service. Some ideas came up about selecting multiple items to edit and publish and how the general user flow should be for that, so it accounts for users with single items and users with multiple items.

Thursday 07 November 2019

Today's objectives are to write up the newsletter.

The day was spent scouring for links for the newsletter and writing it. Subscribe to our newsletter at productfriday.com.

There was also a lengthy discussion about categories for the prototype. We decided to go with 5 categories, and that the users would be able to specify them further with tags themselves.

Friday 08 November 2019

Today's objectives are to see what comes up.

I had no goals for this day, so I had to wait until something happened. The client activated again and remembered the timeline which was already once scrapped from the app and wanted to know estimates for its development. We discussed the amount of work it would take with my boss and came to a conclusion with a careful estimate that it's going to take about a week with the query specifications, backend and frontend. We answered the client with these estimates and asked if the current priorities are still the same and they were.

After confirming that we're on track with the current project we discussed the slides we're going to present to the prototype project owner on Monday. We went through the findings, insights and proposals for the prototype, made some slight changes and refined the wording of some slides to be cut clear.

Weekly Analysis

This week was a continuation of the previous one, with an even larger decline in motivation when it came to work with a client. I found that my communication was not as clear as it could have been, and a lot of the frustration was caused by this lack of communication, resulting in a cycle of demotivation. If I had elaborated on matters a bit more to the client and explained why something needs to be done and what the benefits of that are, the results and the clients understanding of matters would've been clearer.

Next week we will be meeting the prototype project owners and present our findings, insights, wireframes and suggestions as how to go forwards. If all goes well, we will sign a contract with them and allocate 90% of our work time towards the prototype, which should be ready by the beginning of December. Hopefully that will help with my motivation, since I'll be working on a completely new thing and it would change my attitude towards work at the moment.

3.9 Week 46

Monday 11 November 2019

Today's objectives are to attend the prototype insights meeting and work from that.

The day started off with 3 of us, our boss, UX designer and me meeting the prototype project owner and the technical lead. We went through the insights and discussed the findings and found that a lot of the initial assumptions about the pain points in existing processes were correct. The project owner and technical lead were positive about the work we'd done and asked for all of the materials we had gathered during the past few weeks. We said we'll share all the information with them, since all of it was in one place in a Google Drive folder.

After the meeting we decided to forward with the existing wireframe design, since the project owner was satisfied with it. The wireframes were done with an UI kit and the kit had an existing React library, so my boss assigned me to take a look at it and familiarise myself with it and see if we could use it for the prototype. This would allow for us to create the prototype in a much faster way, saving time from building the UI from scratch. I started examining and dissecting the UI kit and on the side we were talking about the user flow in the prototype and how it should work. We quickly realised there are going to be a few small details which need to be agreed on.

With us soon having a lot of work to do, we thought that we need to have some sort of process for tracking our work to keep it cohesive. First idea was to use Clubhouse, but that was too complex for our usage. We decided on going with Trello, having a simple KANBAN board would be easy to keep up to date and see what everyone is doing. I set up the Trello team for us and we put up some initial tasks there.

In the evening I received an email from the existing client about the status of a small feature in the content cards, which was a "partners" field. I had forgotten about this, but it was not on our to-do list either way, so I had a quick chat with my boss regarding this and talked how it would work. The idea is to add a "partner" to a card and there would be an accompanying image with the partner, the logo. Instead of having the user look for a photo and uploading it, we decided we could just find some npm package which scrapes the logo from the website and uses the image URL from there. npm is the world's largest software registry, which has a large public database of JavaScript software (npm 2019). I sent an email to the client proposing this solution and saying it would be ready tomorrow.

The client responded with saying that it's good if it works. After the answer I added a new field to the card schema and decided that I'll finish it tomorrow.

Tuesday 12 November 2019

Today's objectives are to finish the partners field in the card and discuss possible ideas for the prototype.

I continued from where I left yesterday and started working on the API for the logo image scraper. I looked a bit for a suitable package and found one after 15 minutes. At first I tried using the scraper directly in the frontend but it crashed the whole local server and after 5 minutes of figuring out what is the issue I asked my boss why it happens. It was because one can not make requests from the frontend. I'd completely forgotten that. After this basic mistake I continued work on the API and the frontend of the web. After the web was done, I added support to the mobile side. The whole thing took about 2 hours with basic mistakes along the way. After I finished it before lunch, I sent an email to the client informing the changes are live on dev and production and that the mobile builds are processing.

After lunch we discussed the features we are going to do for the prototype and who does what. I'm going to be working on the frontend of the prototype, my boss on the backend and Adrian on the UX and UI. We also discussed what technologies are going to be used and the basics are React, Firebase (Firebase 2019) and Algolia (Algolia 2019) for search. My boss was still drafting the contract and asked a colleague for another pair of eyes to look at it, so he was busy with that the whole day. I was still digging into the UI kit and looking how it works. My boss also asked me to look if we could use TailwindCSS with the UI kit, and its potential. TailwindCSS (TailwindCSS 2019) is "A utility-first CSS framework for rapidly building custom designs", and it looked promising. This way we would not be confined to other design systems and could create our own. I was going to try it out tomorrow in practice and today I would read through the documentation.

Wednesday 13 November 2019

Today's objectives are to try out the new CSS framework and create some components for the prototype with it.

I started the day with setting up my own project to test the CSS framework and create some UI components with it. I started with some basic card items and tested out how the

classes work and the mobile responsiveness as well. The learning curve was quite low, once the few core class names are familiar and the syntax writing small scale components becomes quick. I created a card component for a product in 15 minutes after creating some random UI elements. I went on with creating the smaller UI elements from our wireframes while still looking up the syntax constantly.

From initial impressions Tailwind is a really efficient CSS framework which is nice to work with. The simple and understandable class names help a lot.

Thursday 14 November 2019

Today's objective is to continue with the components which I started yesterday.

I continued with creating components with Tailwind for the prototype while handling some communications with our full-stack developer. We were going through specifications for a timeline for the ongoing project, because I was occupied with the prototype's UI creation and general ideation with my colleagues at the office. I created some mock data mimicking the timeline's future data return for the developer to work with on the mobile UI.

I'm slowly starting to realise having two projects at the same time is very exhausting for me, because I am able to focus fully on only one project at a time. I notice that my quality of work is not the best it can be if I have to constantly juggle between two completely different projects.

Friday 15 November 2019

Today's objective is to start on the product information form.

Today I started on the product information form for the prototype. As the wireframe was already in place for it, creating a base for it did not take too long. At first, I had some issues integrating the Dashboard UI react framework into the project and for it to work with Tailwind CSS. My issues were caused by my inattentive way of writing boilerplate code. I was simply trying to do things in the wrong files, which with hindsight seems very obvious. I finished an initial view of a product form in 2 hours after which I continued with creating the logic handling for it. The mobile responsiveness needed some work as well, so I spent my time polishing the design. I created a "Widget" component which would work as the basic structure for a form group and added a new widget for each form section.

```

1
2   import React from 'react'
3
4   export default function Widget ({ title, description, children }) {
5     return (
6       <div className='my-10 border border-gray-200 rounded sm:flex sm:flex-row bg-white p-4'>
7         <div className='sm:w-2/5 sm:pr-3 mb-4'>
8           <h1 className='text-lg font-semibold text-black mb-2'>{title}</h1>
9           {description}
10        </div>
11        <div className='sm:w-3/5 sm:pr-3 text-gray-800'>
12          {children}
13        </div>
14      </div>
15    )
16  }
17

```

Figure 17. A form Widget

After some time, I realised I had been doing the form logic wrong. I was saving the state in the individual widgets themselves, instead of having the data in the form itself. After this I started moving the logic to the form root and passing all the functions and data as props to the widgets.

At the end of the day we had a call with my colleagues to have a week wrap up of sorts, because there was no Monday meeting. We discussed our current status and upcoming tasks and felt like we were right on schedule with the prototype.

Weekly Analysis

This week made me realise that working two projects at the same time is not easy for me, because I need to focus my attention on one project at a time to achieve the best results. I learned a lot of new things this week, mainly using new tools and syntax. I also aimed to learn to use react hooks, which allow of using a state without creating a class (React Docs 2019) for the prototype and gave me a basic understanding of them.

I'm slowly learning to think how to write even more efficient code, even though I understand my code is not always the cleanest. Most of the times it is due to lack of understanding how or why something works as it does. I found myself reading the documentation a lot more than I have before, and actually understanding it. I used to read documentation on new frameworks, tools etc. and not comprehending it properly some

months ago. I feel like my logical thought processes have developed with writing more code and having to use different technologies.

What I've also found is that having a new project to work on increases my motivation to do work, especially frontend code, but it significantly decreases my motivation to work on the older project in general. With the older project the release cycles vary from days to weeks and when they extend to a longer time period, it doesn't feel rewarding (37signals p.81, 2006). And once the cycles extend a lot, I tend to forget all the small details which have been done and when I inform the client of new updates the small details are not mentioned. Occasionally this causes some remarks from the client implying that we don't do work on their service and that the visible changes are the only things we have done. A lot of the time most of the time is spent on the underlying changes, except when creating new UIs, then the time is sunk into adjusting pixels.

I've found that when doing some small backend changes at times and then making the frontend support it, the frontend part takes 80% of the total work time and there doesn't seem to be a way to avoid it. New features seem to have specific designs and they do not fit with the already existing components. This might be something that needs to be taken into account and developed further, creating an all-purpose component library that eliminates most of the verbose, but necessary code. Hopefully I will have time to work on that at one point, although refactoring older components would take a lot of time and the benefits should outweigh the cost of development.

3.10 Week 47

Monday 18 November 2019

Today's objectives are to continue with the forms for the prototype.

I continued with the product information form today. I spent a lot of time just writing the code and realised the files are going to be very large and no matter how efficient code I'm going to write, all of the code is needed. The forms widgets all require a basic `onChange` function which updates the state of the form accordingly, but they also have specific functions which handle data differently. One particularly different data handling function was for images, where you could drag and drop images or open a file browser to select images. I asked my boss about this and he said it's better to use an existing solution for that, so I found one on npm. The package used react hooks as well, so it's up to date and

maintained. Most of the day was just repeating the same structure for each form widget and making sure the data is saved properly.

I've noticed now that there's TailwindCSS and Dashboard UI kit in use, the styles can be conflicting at times, which makes overriding them time consuming and requiring at times writing verbose code.

Tuesday 19 November 2019

Today's objectives are to create a company information form and flesh out the product information form.

I started the day with creating a form for a company profile. The product information form was not finished yet, and it is going to be the biggest part of the prototype to my knowledge, since it seems to expand exponentially day by day. Creating the company information form was relatively straightforward, requiring only 5 widgets, an index page and 7 functions for handling various data.

After the company form was done, I continued with the product information and started working on a pricing widget, which would list the different prices for a product. A product is going to have multiple categories and the pricing system is going to vary depending on it, so there are going to be multiple pricing widgets. These widgets require for it to handle arrays of items, sorting them and deleting them from the arrays. Once again, I did the mistake of doing the array handling logic in the widgets themselves, instead of in the parent form. I had to move the logic handling up again and create completely new functions for the data arrays.

I'm noticing that the form is starting to become increasingly daunting at first glance, because of its size. I simply do not know if there are better ways of doing the form. That is something I need to ask my boss.

Wednesday 20 November 2019

Today's objectives are to continue with the product information form.

I continued from where I left yesterday, creating more widgets for the pricing of the form. I worked on the form widgets for half of the day. My boss had created a main repository for the project on Monday and asked me to review a few pull requests he'd done and to make

a pull request to merge my code into the main repository. I went through the changes and it had the general frontend, backend, build and deployment scripts there. Everything looked good to me, so I merged the changes and made a clone of the repository onto my laptop so I could make a PR. I had to change the file structure a bit so it would work with it. After the merge my boss started fleshing out the layout of the prototype and I continued on the form. At this point the form creation started feeling like a chore. There is not much new to learn in what I am doing currently, and it is only time consuming.

I'm starting to learn proper data handling practices and reverse engineering old projects helps with that as well.

Thursday 21 November 2019

Today's objectives are to review the latest pull request from yesterday evening and to continue with the forms on the prototype.

I started the day with going through the pull request my boss had made. He'd worked on improving the layout of the prototype, authentication routes and backend. It looked good to me and I did not have any idea what could be done better so I accepted it and merged it. After my boss came to the office, we checked out Google Stadia and how it works, since he had just received the codes to use it. It worked pretty okay for what it was promising, cloud gaming.

After the morning's easy start, he walked me and my colleague through what had been done in the prototype. We're using schema.org schema structures in the prototype, but with some slight modifications to it, trimming down the whole structure of it. We're not using all of the fields, since it would be way too big and most of the properties are not necessary for our use cases. The data passed to the schemas are run through a json validator called AJS (Ajv 2019) which supports the latest JSON Schema draft (JSON-schema 2019), which is a vocabulary for annotating and validating JSON schemas.

With the changes gone through we discussed our areas of responsibility and we agreed that I will continue on creating the widgets for the form and my boss will start with "wiring up" the form with rest of the service. We're going to need to modify the data from the form so it will be adhering to the schema specifications. I'll continue on creating the UI of the form and its widgets. Creating new widgets with slightly varying functions is starting to become very monotone and the initial excitement with the new is slowly calming down. Since my code was a lot of boilerplate and expanding all the time, I asked my boss if there

are any better ways of writing the forms and he said we could look into using hooks instead of components in the widgets as well. He googled a bit and found a very good library, react-hook-form (React Hook Form 2019), for react forms which utilises hooks and on first glance it looks really interesting and efficient. We looked at the documentation and decided that we'll look into refactoring the form tomorrow to utilize this framework once I'm done with the rest of the widgets. Now I have something exciting to look forward to. Rest of the day I spent on creating the form widgets for product pricing and availability.

Friday 22 November 2019

Today's objectives are to look into creating an event calendar and use the react hooks form.

Today's task was to finish the last widget for the initial product form. I was supposed to find an existing library which we could just add to the widget to create it faster. I asked my boss if there was some specific library he wanted to use, and he said there was one he came across that seemed pretty good but did not remember the name of it. I googled for a react event calendar for 5 minutes and came across a simple one which seemed to suit our needs. For a while it seemed difficult to implement into the widget, but after some reading through the documentation the usage turned out to be very simple. I added the calendar to the widget and started to modify it to our needs.

The aim was to have a weekly view in the calendar where the user could select timeframes an event would repeat throughout the year. The library was using different headers for the days than we needed so I started figuring out how they can be rendered differently. Reading through the documentation I found that we can create a component for it which renders the datetimes in the headers, so I started working on it so it would format them into weekdays. After spending 20 minutes trying to understand the passed data structure of the component, I noticed in the documentation that there was a simple prop where I could just pass the formatting options as a string. This rendered all the work on the small custom component useless.

After figuring the issue with the headers I continued on to the data handling, creating the events, modifying the events and deleting the events. This seemed to be difficult, but then I realised to look into the source code of the example and I just copy pasted the functions for the data handling from there. I renamed the variables used to match the form and passed the functions from the parent container into the event calendar as props. This worked without any issues, so I started looking into mobile responsiveness. Turns out the

calendar is not responsive at all and the cell sizes are fixed, causing the calendar to overflow on smaller screens. Adding some overflow rules allowed to move the calendar to see all the days and times, it was not very user friendly. Since this was a prototype and as Steve Krug said where there are constraints, there are trade-offs when it comes to design (2014, p. 310).

During the day as I was working on the event calendar my boss looked into the react hook forms library and came to the conclusion it does not work with our complex form structure. He ended up forking the library and using the form hooks and adding them to our utilities folder. He also refactored the company info form to utilize this new way of handling forms with the hooks. The day ended for me with creating a pull request for the newly added event calendar, my boss merging the changes into his project and making his own pull request. I looked at his pull request for a brief moment before having to head out and I realised I still have a lot to learn.

Weekly analysis

This week's theme was code efficiency and getting introduced to react hooks.

My week started off with writing widgets for a form and then handling the data there. I initially had one simple function to handle input field changes, `handleChange = (name, value) => this.setState({[name]: value})`, then having to create more and more different input handling functions due to nested arrays. With every new array I had to create 3 more functions for adding, editing and deleting the values inside them. I thought about merging the almost duplicated functions into 3 handlers, but after asking about that from my boss he said it does not matter much, and I can just continue with what I am doing.

As I was creating the same type of widgets for most of the week, I constantly felt like these can be done better without writing so much code, but I did not know how. I had a simple structure that I could follow but it did not seem sophisticated enough and my boss was not too bothered about it and whenever I made a pull request. He said it looked good and at times gave some headers on how something should be done. While my work done was received well by a senior software developer, I did not feel too accomplished by what I had done. If there were any changes to be done to the functions, I would have to make the small changes in all the duplicates, and this violates the "do not repeat yourself" and "single point control" principles as pointed out by Diomidis Spinellis (2006).

My concerns were addressed at the end of the week, once I saw my boss' pull request with his fork of the react-hook-form `useForm` API. He had created one higher-order function to handle the input values in any field without requiring an individual function for it. Higher-order functions are functions that take functions as arguments or return functions. (Eloquent JavaScript, 2018) He had also refactored the company information form to use this way of handling the input values.

```
22 export function useForm ({ defaultValues }) {
23   const getDefaultValue = name => _get(defaultValues, name)
24
25   const register = ref => {
26     if (ref && !ref.value) {
27       const val = _get(defaultValues, ref.name)
28       // console.log('register', ref && ref.name, ref && ref.value, '===', val)
29       if (val !== undefined) {
30         ref.value = val
31         // delete defaultValues[ref.name] // consume the value
32       }
33     }
34   }
35
36   const handleSubmit = onSubmit =>
37     event => {
38       event.preventDefault()
39       const data = parse(event.target).body
40       onSubmit(data)
41     }
42   return { register, handleSubmit, getDefaultValue }
43 }
44
```

Figure 18. `useForm` function for handling input values

What I want to do next week is to study how the functions work in the form, from what I understand the function receives some initial values if provided, registers the used inputs with a reference and populates the fields accordingly. Hopefully I'll be using it more in the coming days to get a proper understanding of it.

3.11 Week 48

Monday 25 November 2019

Today's objective is to see where I can continue from last week.

I came to work and started going through the pull request from Friday more closely. My boss had made some additions to it during the weekend so there were some more new changes to be reviewed. There were some schema changes to support arrays as values,

the company form rewired to work with react hooks and some other changes which I did not understand completely. Once my boss came to work, he walked me through the changes he'd made, explaining how the queries now check cached data, so there are no unnecessary queries ran for optimization purposes.

After the walkthrough of the changes we discussed on what I should do next and I was tasked with rewiring the product form to use the hooks just like the company form and to conform to the data models. I started on doing that and the first few widgets were easy to do, since they were simple ones with only text inputs. Once I got to the more complicated form widgets with checkboxes and nested array values, I had to ask my boss for help and instructions on how to do them. I told him the issues I was having, he realised it is going to be a lot more complicated than initially thought and said I can leave it to him and that I can start on a curator workspace UI. I committed my changes, made a pull request and started working on a new page and its layout.

Rest of the day was spent working on the layout and implementing the wireframe we had for the curator workspace. I finished the curator workspace UI by the end of the day while using some mock data, the real data usage and functions are yet to be done.

Tuesday 26 November 2019

Today's objective is to finish the curator workspace functionality.

I continued with the workspace by starting to create the functionality of it. The page was very static at the moment and was not interactive. Selecting items to display did not work as intended and there was no individual state management for the listed items. The intention was to be able to inline edit items on the righthand side of the screen once the user clicked on an item from the list, so I started with that. I ended up using the custom form hooks that were created by my boss for the inline editing and saving the selected item in the state on click. I left the API calls to save the item into the backend commented out, because I did not know how the data would be saved yet.

After the leaving the curator workspace about 80% done, I asked what I could do next and there was a small onboarding page, channel management page, category selection modal and empty states. I started with the channel management page, since I assumed it was the simplest one to do, because all it needs is to render available channels and the channel item components were already done by me in the beginning of the prototype. As I started working on it, I realised my components needed to be refactored and modified

since they did not work well with the page layout. I refactored the UI elements from react classes to functions reducing their size greatly and passing their internal logic to the parent file.

After the components I went on to work with the modal and category item components. I had to redo the category items as well and the modal from scratch. It took the rest of the day and at the end I finally modified the TailwindCSS configuration file to include the primary colour of the service.

Wednesday 27 November 2019

Today's objective is to do the welcome page and see what comes after that.

I started the day with creating a new page where the user would navigate after signing up. We already had an existing design for the welcome page in our Figma file, so I used that as a reference. After creating it, I showed it to my colleagues and the general consensus was that it should be changed a bit. I started doing the modifications and part of the welcome page was to let the user know about the feedback functionality so there was a fixed feedback button on the right side. At first it was just a visual element, but after some discussion we decided that it's better to just have the actual button there, which would already bring the feedback modal to front. I started creating a feedback modal for the welcome page.

After the feedback modal I went back to the curator workspace item view, adding some icons and polishing the UI a bit more. There were lots of small elements to be placed, such as visual indicators how the user would know if the product is published or not, so we were discussing the options with my colleagues as I was doing them.

I remembered that we had also planned on an option to import already existing data from another service and asked my colleagues if we should add that. The answer was yes, and I went on to create that as well. I did the UI and basic form submission functionality, added the loading state to it and left the wiring of the backend for my boss.

Towards the end of the day we were discussing what's been done so far and I mentioned how the modal is still unfinished and isn't the most polished and my boss said that the existing UI kit has a modal in it. I went to check the documentation and it was listed there as a component, but the information was for another component making the documentation a bit odd. Nevertheless, I was mildly frustrated by this, since it made the

work I had done useless and I would be changing the modals to the ones in the kit tomorrow.

Thursday 28 November 2019

Today's objective is to replace the modal with the one from the existing UI kit and other changes that come from that.

I came to the office early to get the modal done with and started reading the documentation and it seemed completely off. There was a simple description about it, but no examples or any further information about it. I went on to check the source code of it and there was no folder for a modal. I was quite certain that it didn't even exist in the kit, so I tried to import it in case the file structure was just different but got an error that it doesn't exist. That was all the confirmation I needed that it really wasn't there.

With no existing modal I started bettering the existing modal and other modal components like feedback, import and image guidelines. Most of the work was just CSS and some occasional JS to toggle the visibility.

After somewhat done modals I started with checking that the components are mobile responsive, and the page layouts conform to basic media queries. I went through the various parts of the project and made them fit for mobile views.

During the day we had a video call with the older client and went through their agenda. The client had decided on a name and were really confident and positive about it and the general rebranding. It seemed that they were investing in the copywriting and the story behind the brand so that was welcome and would explain why they weren't too invested in doing any copywriting for the "old" brand. The general atmosphere was positive, and I found that my previous doubts about their faith in their own product seemed wrong, I just didn't have all the details. We went through what needs to be done and luckily, I had assigned our full-stack developer to work on those during the week, so we were able to say in the video call that their wishes are being worked on. There were some general inquiries about the workload for changing the name of the app and we told that it's not that much work, we just need to find all the right files where there are mentions of the old name. Generally, a simple procedure and that I would start on it tomorrow and merge the changes our full-stack developer has done by then so we can create a new build for them to test out.

After the call I had written down the next steps for this project and decided to get to them right from the morning tomorrow. Rest of the day I spent on fine tuning the media queries for the prototype.

Friday 29 November 2019

Today's objectives are to do all the rebranding of the other project's app and make a new build.

I started with searching for all graphical assets in the app which have the old brand and replacing them with the new brand visuals. There weren't many of them so it was relatively quick, after which I went into the code and checked for any mentions and metadata that would need changing. The binaries were going to be left as is, because changing them would require for us to upload the app again to both app stores, only the user facing side would be rebranded. After the changes I deployed new dev builds for both platforms and sent the client an email notifying them of latest changes and asked if there are any parts we missed, so we can fix them.

I asked my boss what I could work on next and he said we're still missing a general settings page, some empty states and a placeholder for the dashboard. I started with creating a placeholder for the dashboard which just was an SVG image from our wireframe pasted onto the page and added an overlay with a large text saying "Coming Soon" just to give the users a visual clue as to what's going to be developed there in the future.

Next I started creating a new user settings page and using already existing form widgets to display the input fields and general structure. After assembling the visuals and contemplating for a moment if I would make a pull request with existing changes, so my boss can do the functionality I decided to start on the functionality myself. I looked over how the company information form was done, how the data was being saved and applied those practices to the settings form. It took me a bit and I finished it by the end of the day. Once I tested it out a few times it seemed to work fine so I made a PR. At the end of the day we had a call with my colleague who was working remotely to have a weekly wrap up and what's coming next week. We're still on track with the prototype but have wasted some time on small unnecessary details. My boss said it happens all the time and it's normal as he also had misplaced some time into figuring out matters which are not actual

to the prototype. After the call I asked my boss to look over the PR and he said it was fine, so I ended the work week there.

Weekly analysis

This week's theme was finding my own limits to my knowledge and being able to see our company's workflow in a more analytical way.

I found a reoccurring theme this week that I created visuals for the prototype, but quite often didn't create the underlying functionality for the frontend. Many of the required functionalities were really advanced for me and I passed them for my boss to do. He was fine with this and while it generally created a good workflow, I personally felt lacking in my own skills. Once he had created proper form and image handling, I looked over the code and couldn't even guess where to start with all the things that were done. My knowledge on optimization and best practices seemed to be lacking. I would guess all of this comes with experience and by doing things. With creating these new functionalities, I might come across details which I didn't know at first and upon doing them I would know they exist.

A good workflow seems to be that my colleague does the wireframing and visuals for the prototype, I create the visuals on the frontend with some functionality and our boss finishes the rest connecting the frontend with the backend while improving my work after I've done it. I keep picking up on ways to write better code from my boss and try to incorporate the ways into my future work. Quite often I spend some time digging into the code and trying to get to the fundamentals of it.

I have been discussing my general thoughts about this prototype project with my colleague, who is also on a junior level. Our boss wants to have a flat structure in our company, making everyone in the company equal, and that is a good thing. There are side effects to this as well, such as when facing a problem where us, the junior level employees don't know how to proceed we ask our boss for help and he might answer like we are perfectly capable of making these decisions. This leads to some stress as we do not know enough and how our decisions can have crucial implications in the projects. We are not as experienced as he is. Sometimes I feel like my boss needs to be more like a boss and use his authority purely based on his experience and skillsets. There should be a clear acknowledgement of who to refer to in these situations (Devaney, 2019). There

would be no questions asked and we would have direction. This is something that I will bring up in our next company feedback session.

Working in a small flat structured company has had a very steep learning curve and I feel like I would not have learned this much in a larger corporation where responsibilities might not be as vast. Having responsibility for certain areas of a project and just doing them without asking for permission has a degree of freedom which is very rewarding.

Personally, I am glad to work here, even though there are times when my motivation gets really low. The motivation drops seem to come from times when I feel like my work done is futile. This seems to happen when I am dealing with clients and their wishes for a long period of time. When it happens, I have noticed my work ethics are not highest and the quality of my work suffers due to simply not caring enough about it. This is something I need to work on, not let my current feelings affect the quality of work done.

4 Discussion and conclusion

When I started the diary thesis, I had worked at the company for almost a year and most of my work had focused on developing a mobile app for a web service with some occasional supporting web functionalities. I had gotten quite comfortable in my position and often would receive tasks which were similar to the ones done before. With starting the diary thesis, I coincidentally started receiving tasks which I had not done before, allowing me to record my progress and learning with the diary thesis. With new tasks I could write down my current practices which allowed me to come back to them at a later time when my skillsets had developed more and spot areas which required improvement.

On the first week of documenting I was constantly coming across bugs in the existing service, doing hotfixes for them and being in contact with the client actively. I attended meetings with the client and was part of their user testing groups as technical support. In the user testing sessions, I saw how situations develop when there are issues with the service which are crucial to the user experience. Even though some technical parts worked as intended, the user facing side of the service felt lacking. Since the users had not participated in the development process in any way, it seemed to them that the service was broken. Being in a situation which does not go smoothly was not comfortable, but it is good to experience, because then I can predict what might happen with certain features not working as intended and focus on the problematic areas in preparation for the user testing sessions. Besides technical support I also started work on a completely new API, which I had not done before. When starting to create the API I had no idea where to begin but decided to go with an atomic level of approach, starting from the backend with basic getter and setter functions. Most of the initial work was done by reverse engineering already existing API's. I looked for the simplest ones to grasp the general idea. This approach worked well, and I managed to create an API structure and connect it between the front- and backend.

On the second week I finished the API which I started last week and had the Digital Prototyping Week workshop session on Wednesday. I had some basic issues with passing parameters properly through the API, like destructuring the query properly. Coming across these basic issues made me realise I still need to learn a lot and I felt completely lacking in general knowledge at times. Looking back on those errors they seem really simple now, but then it was quite perplexing as to where the issues are coming from. I found one of my most used ways of debugging is to console log everything. Many times, I was stuck at doing one change, then reading the logs in hopes of spotting what works and what doesn't. After all the changes and getting the API to work, it felt

good to do something new, even though it was just a simple API. Making these small accomplishments was part of my everyday job, and in the moment, it seems just like any other task, but in the end the small differences build up to making big changes.

On the second week we also held a prototyping workshop at The Shortcut, which allowed us to bring some visibility to our new company branch, which focuses on making rapid prototypes. It was a good experience and somewhat validated me as an actual person in the IT-field. Before I felt like I don't know enough to present anything to anyone, but the questions in the workshop made me realise I do know some things and were able to help people with their questions.

The third week was a collection of different happenings, including technical issues, feedback sessions, creation of a marketing campaign and a premature pilot ending. There were some underlying issues in the way our company had been operating for a while and it started to hinder my performance and affect my motivation. Without letting the negativity completely take me under I decided to bring these issues to light with my boss and it was the right choice. Being able to discuss and bring up feedback with others is very important and relieving. This prevents from tension building up in the company. While it was difficult to bring negative points to light, it was for the better, because that way they could be addressed and worked out. In addition to some issues internally there was a case of a pilot failing due to an unexpected database issue. All other pilots had been quite patient and ok with some mishaps in the past, but this pilot ended it the moment there was an issue, even though it was stated that there might be some stability issues. But, in the end it also seems that the pilot had completely different expectations as to what the service is providing, so it is understandable why they ended it, didn't have the right value propositions for them.

On the fourth week most of the time was spent working on another API which allows admins to toggle the visibility of content in different organisations. Once again I came across an issue that I was looking at too closely completely missing the larger picture and trying to implement a wrong logic to it as well. Trial and error and with the help of my boss I figured the issue out. Most of the time I've been confident to ask help with issues and this was also one of those times.

On the fifth week there was a lot of copywriting, which I found not to be part of my skillset. Since I was the only native Finn in the company the job landed on me and I had to translate a lot of text into Finnish while trying to maintain the tone of voice and other intricacies from the English text. The task was very draining, and I genuinely hope I will

never have to do that again, because I am not good at it. Besides copywriting we had a new client meeting which went well and landed us a prototype to work on. The prototype project would be validating our intended business model for the company's prototyping branch which is interesting to see how it pans out.

On the sixth week most of the work was spent on redoing a chat UI and adding a settings page to it. One principle which seemed good was to not reinvent the wheel and follow current best practices on how UI's might look and work. Following Telegram's UI was a good baseline for the new changes and saved a lot of time. Once again, I came across issues with API's and asynchronous functions which delved from my lack of some basic understanding. I kept making the same mistakes quite a few times and every time learned a bit more.

On the seventh week there were some additions to be made to an admin panel and other work that was just small usability and UI changes and fixes all around the service. There was a lot of confusion from the client's side for some reason which resulted in many email exchanges trying to explain processes very in much detail while keeping it simple to understand. Week eight was not very eventful and it required for me to find work to do to keep me occupied. This resulted in me finding some small product app store page details being rewritten and fleshed out a bit more, even though it isn't even my job to do them.

On the ninth week the workload increased exponentially. We attended a meeting with the new client regarding the prototype and held a small workshop of our own to map stakeholders and priorities. With agreed priorities we started work immediately and chose a react library which would fit the purpose best. I started with testing out the library and creating individual components which was enjoyable and productive work. Most of the week I created new components and by the end of the week I had to add some abstraction to them so they would be compatible with the main repository. That required a bit of rewriting, mostly replacing the individual component states with props that would be passed into them.

On the last weeks my tasks rose in difficulty having me create forms for the prototype. The new functionalities seemed very challenging at first, but once I started approaching them piece by piece, I noticed good patterns which allowed for writing code faster. I've always found creating forms to be challenging and this was the perfect opportunity to overcome those challenges and learn new things. I performed well and even though I didn't finish up 100% of the functionalities in the forms, I got good feedback on my work from my colleagues.

Writing this diary thesis, I've learned to spot good common practices in programming, which has been achieved by doing similar functionalities multiple times throughout the observation period. Once I don't focus on the details and think on a more abstract level the patterns seem obvious. With applying this thinking coming up with simple solutions to new problems also becomes easier. One example of coming up with a simple solution was the visibility of content in a service. The implementation was easy in the end and it is very simple, toggling ids in an array.

I've learned to debug code better, since I understand the underlying behavior of the code. Not everything, but a lot more than I did in the beginning and I find myself doing less repeated mistakes of basic nature. While I do mistakes, they are mostly of different nature than simple "wrong condition in a filter function".

I've also been able to spot my own shortcomings and weaknesses. I tend to keep my opinions to myself and rarely voice them thinking that others will say the same thing so there is no reason for me to say them. This is something I still need to work on, and probably will take a lot longer than learning a new JavaScript library.

References

- Clance, P. R. & Imes, S. 1978. The Imposter Phenomenon in High Achieving Women: Dynamics and Therapeutic Intervention. Georgia State University
- 37signals 2006. Getting Real. The smarter, faster, easier way to build a successful web application. URL: <https://basecamp.com/books/Getting%20Real.pdf> Accessed 14 November 2019
- Haverbeke, M. 2018. Eloquent JavaScript, 3rd Edition. A Modern Introduction to Programming
- Krug, S. 2014. Do not Make Me Think, Revisited. A common sense approach to web usability. 2nd ed. New Riders.
- Spinellis, D. 10 tips for spotting low-quality open source code. Enterprise Open Source Journal, 2(5):18–21, Sept/Oct 2006.
URL: <https://www.spinellis.gr/pubs/trade/2006-EOSJ-BadCode/html/10tips.htm>
Accessed 22 November
- Devaney, E. 2019. 9 Types of Organizational Structure Every Company Should Consider
URL: <https://blog.hubspot.com/marketing/team-structure-diagrams> Accessed 30 November 2019
- Algolia 2019.
URL: <https://www.algolia.com/> Accessed 15 November 2019
- Ajv 2019.
URL: <https://ajv.js.org/> Accessed 21 November 2019
- The Balance Careers. The Basics of Key Performance Indicators 2019
URL: <https://www.thebalancecareers.com/key-performance-indicators-2275156>
Accessed 11 October 2019
- Firebase 2019.
URL: <https://firebase.google.com/> Accessed 15 November 2019
- Google Analytics 2019.
URL: <https://analytics.google.com/analytics/web/> Accessed 16 September
- JSON Schema 2019.
URL: <https://json-schema.org/draft/2019-09/json-schema-validation.html> Accessed 21 November 2019
- MDN Web Docs. Destructuring assignment (2019a)
URL: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Destructuring_assignment Accessed 24 September 2019

MDN Web Docs. async function (2019b)

URL: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/async_function Accessed 25 October 2019

MDN Web Docs. forEach function (2019c)

URL: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Map/forEach Accessed 4 November 2019

npm 2019.

URL: <https://www.npmjs.com/> Accessed 01 November 2019

TailwindCSS 2019. A utility-first CSS framework for rapidly building custom designs.

URL: <https://tailwindcss.com/> Accessed 12 November 2019

React Docs. Facebook 2019. React documentation.

URL: <https://reactjs.org/docs> Accessed: 15 November 2019

React Hook Form 2019.

URL: <https://react-hook-form.com/> Accessed 21 November 2019

React Redux 2019.

URL: <https://redux.js.org/> Accessed 26 September 2019