



jamk

Hallintajärjestelmä laskuvarjohyppytoiminnalle

Eemil Rantala

Opinnäytetyö, AMK

Kesäkuu 2024

Tieto- ja viestintätekniikan tutkinto-ohjelma

Rantala, Eemil

Hallintajärjestelmä laskuvarjohyppytoiminnalle

Jyväskylä: Jyväskylän ammattikorkeakoulu. Kesäkuu 2024, 34 sivua.

Tieto- ja viestintäteknikan tutkinto-ohjelma. Opinnäytetyö AMK.

Julkaisun kieli: suomi

Julkaisulupa avoimessa verkossa: Kyllä

Tiivistelmä

Opinnäytetyön tavoitteena oli suunnitella ja toteuttaa hyppytoiminnanhallintajärjestelmä Jyväskylän laskuvarjokerholle. Sovelluksen oli tarkoitus korvata kerhon aiemmin käyttämä järjestelmä, sillä se oli vanhanlainen ja epäkäytännöllinen. Uudessa sovelluksessa käytettiin alan uusimpia teknologioita varmistamaan parhaan käyttökokemuksen ja pitkäikäisyyden. Mobiilikäytettävyyteen kiinnitettiin erityistä huomiota, sillä se oli yksi suurimmista syistä uuden järjestelmän toteuttamiselle. Tarkoitus oli mahdollistaa jokaisen kerhon jäsenen itsenäinen sovelluksen käyttö ja vähentää toimistotyön määrää.

Hallintajärjestelmä on Vue-sovelluskehysellä rakennettu web-sovellus, joka tukeutuu Expressillä toteutettuun rajapintaan. Rajapinta on yhteydessä pilvipalveluna toimivaan MongoDB Atlas tietokantaan. Sitä lukuun ottamatta sovelluksen kehitystyö tapahtui täysin paikallisessa ympäristössä. Palvelinpuolella oleva mongoose npm paketti tarjosi yksinkertaisemman rajapinnan luoda hakuja ja malleja MongoDB tietokantaan. Selainsovelluksessa käytettiin Vuetify käyttöliittymäkirjastoa elementtien rakentamisessa. Vuetifyn valmiit komponentit perustuvat Googlen kehittämään Material Design ohjeistoon, joka parantaa saavutettavuutta ja käytettävyyttä.

Opinnäytetyön tuloksena toteutettiin manifestijärjestelmän beeta versio. Sovelluksesta saatiin toiveiden mukainen ja varsinkin käyttöliittymän suunnittelussa onnistuttiin luomaan toimiva ratkaisu eri kokoisille laitteille. Päätoiminnallisuuksina sillä pystyy organisoimaan hyppylentoja ja jäseniä, sekä hoitamaan heidän hyppytilien saldoa. Järjestelmä pitää kirjaa kaikista sen sisäisiä hyppyihin liittyvistä tilastoista ja tarjoaa niille helpot tarkastelutyökalut. Jokaisen jäsenen on mahdollista ottaa sovellus itsenäisesti käyttöön ja lisätä itsensä lennoille tai tarkastella omia tietoja, kuten saldoa. Sovellus on tarkoitus ottaa käyttöön kaudelle 2025. Jatkokehitys mahdollisuuksia on runsaasti moneen eri suuntaan ja projekti on tulevaisuudessa avoin uusille kehittäjille ja ideoille.

Avainsanat (asiasanat)

Express, MongoDB, NodeJS, TypeScript, Vue, Web-kehitys

Muut tiedot (salassa pidettävät liitteet)

-

Rantala, Eemil

Skydiving management application

Jyväskylä: JAMK University of Applied Sciences, June 2024, 34 pages.

Degree Programme in Information and Communications Technology. Bachelor's thesis.

Permission for open access publication: Yes

Language of publication: Finnish

Abstract

The purpose of the thesis was to design and implement a skydiving management application for Jyväskylän Laskuvarjokerho ry. The application was supposed to replace the older system used by the club, as it was old-fashioned and impractical. The new application used the latest technologies in the field to ensure the best user experience and longevity. Special attention was paid to mobile usability, as it was one of the biggest reasons for implementing the new system. The purpose was to enable each club member to use the application independently and to reduce the amount of office work.

The management system is a web application built with the Vue framework, which relies on an API implemented with Express. The API is connected to cloud-based MongoDB Atlas database. Apart from this, the development of the application took place entirely in a local environment. The mongoose npm package on the server side provided an interface to create searches and models for the MongoDB database. The client application used the Vuetify graphics library to build the user interface elements. Vuetify's ready-made components are based on the Material Design guidelines developed by Google, which improves accessibility and usability.

A modern manifesting system was implemented as a result of the thesis. The application was in line with the wishes, and especially in the design of the user interface, we managed to create a functional solution for devices of different sizes. As the main functionality, it can organize jump loads and members, as well as manage the balance of their jump accounts. The software keeps accurate records of its internal jump statistics and offers them facilitation tools. It is possible for each member to use the application independently and add themselves to loads or their own information, such as balance. The application is supposed to be put into use for the season 2025. The project is open for new developers and ideas.

Keywords/tags (subjects)

Express, MongoDB, NodeJS, TypeScript, Vue, Web-development

Miscellaneous (Confidential information)

-

Sisältö

1	Johdanto	6
1.1	Taustaa	6
1.2	Tavoitteet ja toteutus.....	6
1.3	Toimeksiantaja	7
2	Vertailu kilpailijoihin.....	7
2.1	SkyWin.....	7
2.2	Pinnikey.....	7
2.3	Burple	8
3	Teknologiat	8
3.1	Toteutustapa	8
3.2	Frontend.....	9
3.2.1	Vue	9
3.2.2	Vuetify.....	11
3.2.3	Käännökset	11
3.2.4	Axios.....	12
3.3	Backend	14
3.3.1	Rajapinta	14
3.3.2	Express	14
3.3.3	MongoDB	16
3.4	TypeScript.....	17
4	Sovelluksen suunnittelu ja toteuttaminen.....	17
4.1	Sivustokartta	17
4.2	Rautalankamalli.....	18
4.3	Tyylit.....	21
4.3.1	Kirjasintyyppi	21
4.3.2	Värit	22
4.3.3	Kuvakkeet	23
4.4	Käytännön asioita toteutuksessa	23
4.4.1	Rekisteröityminen.....	23
4.4.2	Käyttöoikeudet	24
4.4.3	Lentokoneet ja hinnoittelu	24
4.4.4	Pokalista.....	26
4.4.5	Maksut	26
4.4.6	Tilastot	27

4.5	Turvallisuus.....	28
4.5.1	Rekisteröityminen.....	28
4.5.2	Käyttäjätunnus ja salasana	29
4.5.3	Käyttäjän todentaminen	29
4.5.4	HTTPS	30
5	Lopputulos	30
	Lähteet	32

Kuviot

Kuvio 1.	Sovelluskehysten tagien esiintyvyys Stack Overflowssa.....	9
Kuvio 2.	Sovelluskehysten viikottaiset lataukset 14. - 20.4.2024	10
Kuvio 3.	Koodiesimerkki käännöksistä	12
Kuvio 4.	Kuvankaappaus kerhon asetus sivusta suomeksi ja englanniksi	12
Kuvio 5.	Kuvankaappaus koodieditorista jossa auki käännöstiedostot	12
Kuvio 6.	Koodiesimerkki verkkopyynnön virheenkäsittelystä	13
Kuvio 7.	Axios ja fetch vertailu koodiesimerkki	13
Kuvio 8.	GET -päätepiste jäsentietojen hakua varten.....	15
Kuvio 9.	Esimerkki middleware funktiosta.....	15
Kuvio 10.	Sovelluksen sivustokartta.....	18
Kuvio 11.	Rautalankamalli oletusnäkyvästä	19
Kuvio 12.	Rautalankamalli Kerho -näkyvästä	19
Kuvio 13.	Rautalankamalli Kerho -näkyvästä mobiili resoluutiolla	20
Kuvio 14.	Toteutettu jäsentietolomake täysikokoisena ja mobiilissa	21
Kuvio 15.	Kuvankaappaus sovelluksen navigaatiosta ja osasta jäsensivua	22
Kuvio 16.	Kuvankaappaus käyttäjän asetuksista tummalla teemalla.....	23
Kuvio 17.	Kuvankaappaus toteutuksen kerhon asetussivusta.....	25
Kuvio 18.	Kuvankaappaus lentokoneen muokkauslomakkeesta.....	25
Kuvio 19.	Kuvankaappaus toteutuksen pokalistasta	26
Kuvio 20.	Kuvankaappaus maksun tekemisestä jäsenlistassa	27
Kuvio 21.	Kuvankaappaus maksulomakkeesta	27
Kuvio 22.	Kuvankaappaus kuvitteellisista tilastoista	28
Kuvio 23.	Kuvankaappaus rekisteröitymislomakkeesta.....	28

1 Johdanto

1.1 Taustaa

Opinnäytetyön tavoitteena oli kehittää hyppytoiminnanhallintajärjestelmä Jyväskylän Laskuvarjokerholle, joka korvaa aiemmin käytössä olleen järjestelmän. Idea työlle syntyi, kun kerhon järjestelmä oli vanhanaikainen ja hidas käyttää, eikä siitä saatavia tietoja voinut tarkastella muilla laitteilla. Tarkoituksena oli luoda yksinkertainen ja kustannustehokas sovellus pienen harraste kerhon tarpeisiin. Kaupalliset järjestelmät saattavat tarjota laajan valikoiman toiminnallisuuksia, joihin pienemmillä kerhoilla ei ole tarvetta tai resursseja.

1.2 Tavoitteet ja toteutus

Uuden sovelluksen oli tarkoitus poistaa edellisen järjestelmän rajoittuneisuus Windows-ympäristöön ja mahdollistaa käyttö myös mobiililaitteilla. Kerhon jäsenten haluttiin pystyvän käyttää sovellusta itsenäisesti, mikä vähentäisi toimistotyötä, helpottaen kerhon jokapäiväistä toimintaa. Uuden sovelluksen tuli olla selkeä ja nopea käyttää, toisin kuin edeltäjänsä.

Ratkaisuksi valittiin web-pohjainen ympäristö, mikä yksinkertaisti koodipohjaa ja käyttöönottoa eri laitteille. Vue valittiin JavaScript-sovelluskehikseksi sen suosion ja yksinkertaisuuden perusteella. Vuetify tarjosi valmiita komponentteja, jotka olivat sekä esteettisesti miellyttäviä että intuitiivisia, täyttäen projektin tarpeet toimia niin isoilla kuin pienilläkin näytöillä. Tämä valinta mahdollisti helpon käyttöliittymän kehittämisen, joka toimi saumattomasti eri laitteilla. Sovellus käyttää MongoDB tietokantaa Expressillä kehitetyn rajapinnan kautta.

Uuteen sovellukseen ei tullut alkuun muuta kuin toiminnan kannalta pakolliset ominaisuudet, kuten jäsenten ja lentojen hallinta, hyppytili sekä vuosittaisten tilastojen tarkastelu. Lopputulos on yhdistelmä kilpailijoiden hyvistä puolista, tarpeettomat toiminallisuudet karsittuna. Sovellukseen on jätetty liikkumavaraa, jotta sitä on myöhemmin mahdollista jatkokehittää sen käytön aikana syntyvien ideoiden ja tarpeiden pohjalta.

1.3 Toimeksiantaja

Työn toimeksiantajana toimi Jyväskylän Laskuvarjokerho ry. Kerho tunnetaan myös nimellä Skydive Jyväskylä ja se on perustettu vuonna 1970 (Kerhon esittely n.d.). Yhdistyksen tehtävänä on tukea laskuvarjourheilua eikä sen päämääränä ole tavoitella voittoa (Jyväskylän Laskuvarjokerho ry:n säännöt n.d.). Kerho toimii täysin harraste vetoisesti ja on suhteellisen pieni vuosittaisissa hyppymäärissä mitattuna, sillä vuonna 2023 hyppyjä oli noin 1400 (Lähde 2023).

2 Vertailu kilpailijoihin

2.1 SkyWin

SkyWin on kerhon tämänhetkinen käytössä oleva järjestelmä. Se on vain Windowsille saatavilla oleva työpöytäsovellus, mutta siitä on olemassa uudempi 2021 julkaistu web-pohjainen versio SkyWin One. Ensimmäinen versio SkyWinistä on vuodelta 1992 ja siihen on päivitetty vuosien varrella uusia ominaisuuksia, mutta käyttöliittymä on jäänyt vanhanaikaiseksi. (SkyWin history - the evolution of SkyWin 2024.)

Sovellus pyörii täysin paikallisesti yhdellä kerhon tietokoneista ja sen käyttö rajoittuu vain sille koneelle. Se on koettu rajoittavana tekijänä kerhon toiminnan sujuvuudelle. SkyWinissä on runsaasti ominaisuuksia, joita Jyväskylässä ei käytetä, kuten ryhmien ja tapahtumien organisointi toiminnot. Nämä lisää valikoiden määrää ja sekavuutta. Uusien jäsenten on ollut hidasta omaksua kuinka SkyWiniä käytetään ja se on vähentänyt sen osajien määrää kohdistuen vastuuta pienemmälle käyttäjä porukalle. SkyWinin käyttöä häiritsee myös joka päivä esiintyvä bugi, joka pitää osata kiertää. Järjestelmä kustantaa kerholle noin 150 euroa vuodessa (Lähde 2024).

2.2 Pinnikey

Kerholla on ollut aiemmin kehitteillä vastaava järjestelmä nimeltä Pinnikey, joka oli laskuvarjoharrastaja Tommi Savikon tekemä avoimen lähdekoodin ohjelmisto (Haapsalo 2024). Sen kehitys kuitenkin keskeytyi vuonna 2016, eikä kaikkia suunniteltuja toiminnallisuuksia saatu tehtyä (Pinnikey 2016). Savikko (2017) kirjoitti blogissaan Pinnikey projektista, todeten ajan ja motivaation loppuneen kesken. Projektin lähdekoodi on saatavilla GitHubissa (Pinnikey 2016). Projektia voisi jatkaa

kuka tahansa, mutta opinnäytetyön kannalta oli parempi toteuttaa alusta asti uusi sovellus, jossa on mahdollista käyttää uudempia teknologioita.

2.3 Burble

E erityisesti Yhdysvalloissa suosittu Burble sopii suuremmille hyppykeskuksille, joille toiminta on kaupallista. Sillä hoituu normaalin manifestoinnin lisäksi muun muassa lahjakortit, ajanvaraukset ja myyntiraportit. (Burblesoftware 2024.) Edellä mainituilla ominaisuuksilla ei saataisi kovin suurta hyötyä Jyväskylän pienelle harrastekerholle. Burble tulisi huomattavasti kalliimmaksi kuin SkyWin, sillä sen perustilauksen hinta on 2 % kerhon tuloista, eikä kiinteitä maksuja ole (Gerschwiler 2024). Suomessa Burble on käytössä ainakin Utissa sijaitsevalla Skydive Finlandilla, joka on suomen suurin laskuvarjokerho (Skydive Finland 2021).

Järjestelmään kuuluu myös BurbleMe mobiilisovellus, jolla jokainen voi lisätä itsensä tai ryhmänsä lennoille, jonottamatta manifestiin eli toiminnan organisoinnin palvelupisteeseen. Sovelluksesta voi seurata listaa lennoista ja ilmoittaa avun tarpeesta, mikäli laskeutuu alueen ulkopuolelle. BurbleMe:llä voi myös lisätä rahaa hyppytilille ja seurata tilitapahtumia. (Burble Me 2024; Hyppääminen Utissa 2021.) Poikkeuksena rahan lisääminen sekä hyppytilien väliset siirrot eivät ole mahdollisia Skydive Finlandilla, eikä kaikkia Burblien ominaisuuksia ole mahdollista saada käyttöön Suomessa erinäisten rajoitusten takia (Hyppääminen Utissa 2021).

3 Teknologiat

3.1 Toteutustapa

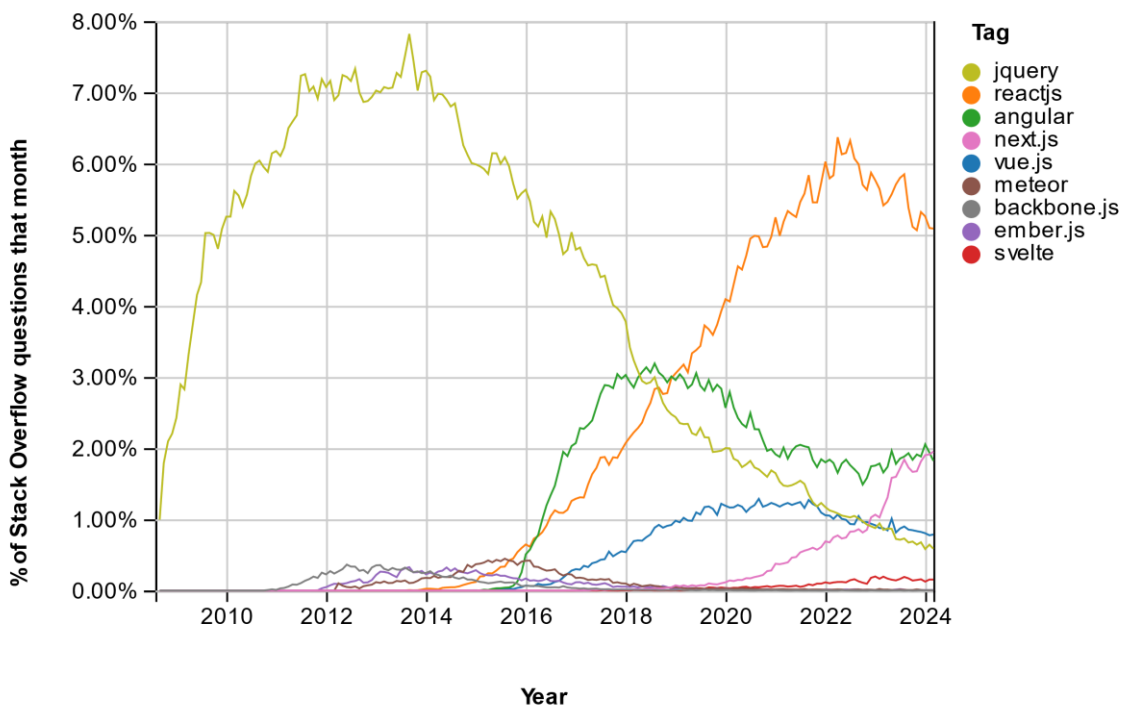
Sovellus päätettiin toteuttaa web-teknologioilla, sillä se tuo mukanaan useita etuja niin sovelluksen kehittämiseen kuin käyttöönkin. SkyWinin kaltainen rajoittuneisuus Windows-ympäristöön haluttiin poistaa ja mahdollistaa sovelluksen käyttö myös mobiililaitteilla. Samalla ei haluttu myöskään, että jokainen käyttäjä joutuisi asentamaan sovelluksen laitteelleen, sillä se olisi yksi käyttöä hidastava vaihe lisää. Yksinkertaisin ratkaisu tähän oli valita web-pohjainen ympäristö. Etuina siinä on, että yksi koodipohja riittää, eikä tarvitse tehdä omia toteutuksia eri laitteille. Web-sovelluksissa käyttäjien ei myöskään tarvitse huolehtia sovelluksen päivittämisestä, vaan päivityksiä voidaan julkaista milloin tahansa.

Oman sovelluksen tekeminen ei maksaisi kerholle mitään, toisin kuin muun vastaavan järjestelmän hankkiminen. Omassa sovelluksessa mahdollisia kuluja syntyisi palvelimesta, jossa sovellus pyörii. On kuitenkin olemassa ilmaisia alustoja palvelun pyörittämiseen, kuten Netlify Core, jossa saa tehdä tietyn määrän verkkopyyntöjä kuukaudessa maksutta (Netlify Core 2024). Kerholla on myös oma verkkotunnus, jota sovellus voi käyttää.

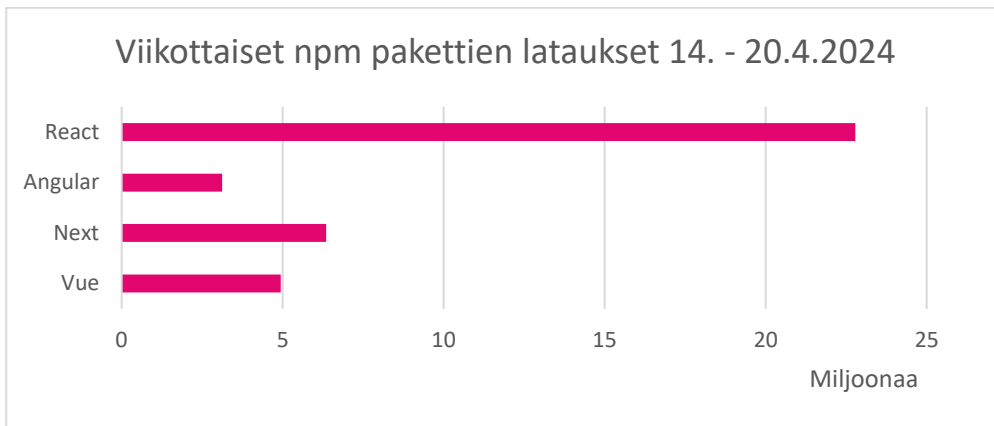
3.2 Frontend

3.2.1 Vue

Sovellukseen haluttiin tunnettu ja helppo JavaScript kehys, joten työssä päädyttiin käyttämään Vuea, sillä se on ollut suosituimpien sovelluskehysten joukossa viime vuosina (Stack Overflow Trends 2024). Kuviossa 1 vertaillut kehukset on koostettu Top 10 Most Popular JavaScript Frameworks For Web Development (2024) ja Top 10 Most Popular JavaScript Frameworks to Choose in 2024 (2023) blogi kirjoituksissa listatuista suosituimmista sovelluskehyksistä. Joukossa on myös jQuery, joka ei ole varsinaisesti sovelluskehys, vaan JavaScript-kirjasto (What is jQuery? 2024). Kuvion 1 trendistä päätellen, ennen ei juuri ollut kunnollisia verkkosovelluskehymiä niin kuin tänä päivänä. jQueryn suosio onkin laskenut jyrkästi Reactin ja muiden kehysten yleistyessä.



Kuvio 1. Sovelluskehysten tagien esiintyvyys Stack Overflowssa (Stack Overflow Trends 2024)



Kuvio 2. Sovelluskehysten viikottaiset lataukset 14. - 20.4.2024 (npm 2024)

Daityarin (2023) mukaan React ja Angular sopivat paremmin suuriin projekteihin, kun taas Vue soveltuu hyvin pieniin ja keskikokoisiin sovelluksiin. Kuviossa 2. esiintyvä toiseksi eniten ladattu sovelluskehys Next pohjautuu Reactiin, tarjoten siihen lisäominaisuuksia (About React and Next.js 2024). Vaikka React on ollut suosituin, niin myös Arushi (2024) toteaa Vuen olevan hyvä vaihtoehto pienempiin sovelluksiin. Näiden perusteella suosituimpien kehysten joukosta Vue oli helpoin oppia ja soveltuvin pienempään sovellukseen, joka mahdollisti sen nopeamman kehityksen.

Vue.js on progressiivinen JavaScript-kehys, joka on suunniteltu helpottamaan web-käyttöliittymien rakentamista. Tässä yhteydessä progressiivinen tarkoittaa, että Vuea voi hyödyntää asteittain pala kerrallaan, ilman että tarvitsisi tehdä suuria muutoksia projektiin. (Introduction 2024.) Vue on helpompi oppia kuin React, sillä se käyttää standardi HTML, CSS ja JavaScript syntaksia tiedostoissaan, kun taas Reactissa on oma syntaksinsa (Raval 2023).

Työssä on käytetty Vue 3:a, sillä Vue 2 tuli elinkaarensa päähän vuoden 2023 lopussa, joten sitä ei ylläpidetä enää. Vue 3 on edeltäjänsä tehokkaampi ja tarjoaa pienemmän pakettien koon, paremman skaalautuvuuden sekä paremman TypeScript tuen. (FAQ 2023.)

Vue mahdollistaa useita eri tapoja toteuttaa sovelluksia, kuten yksittäisten komponenttien tekemisen osaksi perinteistä HTML-sivua tai edistyksellisemmät Single-Page Application (SPA) ja Server-Side Rendering (SSR) tekniikat. Kahden viimeisen erona on se, että luodaanko näytettävän sivun HTML koodi selaimessa (SPA) vai palvelimella (SSR). Työ tehtiin yhden sivun sovelluksena (SPA),

sillä sovelluksen luonteen takia ei vaadita erityistä hakukoneoptimointia, mitä SSR tarjoaa. (Ways of Using Vue n.d.)

3.2.2 Vuetify

Vuetify on UI-kehys, joka hyödyntää Material Designia luodakseen komponentteja, jotka ovat intuitiivisia sekä miellyttävän näköisiä (Introduction n.d.a). Material Design on Googlen kehittämä suunnittelujärjestelmä, joka sisältää ohjeita, komponentteja ja työkaluja, jotka tukevat parhaita käytäntöjä käyttöliittymäsuunnittelussa (Introduction n.d.b). Vuetify tarjoaa laajan valikoiman valmiita komponentteja, yksinkertaisista painikkeista ja lomakkeista monimutkaisiin taulukoihin ja dialogiruutuihin. Vuetifyn valinta oli luonnollinen Vue projektin kannalta, sillä kaikki sen komponentit on rakennettu Vuella. Komponentit ovat tehty responsiiviseksi, joten sovelluksen saa helposti valmiin näköiseksi ja toimimaan saumattomasti millä tahansa laitteella. (Introduction n.d.a.) Vuetify vastaa siis suoraan työn tarpeisiin kehittää yksinkertainen käyttöliittymä, joka toimii työpöydällä ja mobiilissa.

Vuetifyssä on myös valmiina kattava väripaletti niin vaalealle kuin tummallekin teemalle (Colors 2024). Sovellus hyödyntää täysin tätä valmista palettia, sillä oman toimivan paletin tekeminen on haastavaa, varsinkin tummalle teemalle värejä sovittaessa.

3.2.3 Käännökset

Sovelluksen käännökset toteutettiin vue-i18n:llä, joka on Vue yhteensopiva internationalisointi paketti. Sen avulla sovelluksen kieltä pystytään vaihtamaan ilman että sivua päivitetään välissä. Myöskään URL-osoitteeseen ei haluttu tuoda kieliasetusta, jotta se pysyisi siistinä. Sovellus tarkistaa alustuksen yhteydessä selaimen kielen ja asettaa sen käyttäjälle, jos kieltä ei ole vielä manuaalisesti valittu. Jos selaimen kieli on jokin muu kuin suomi tai englanti, niin sovellus on oletuksena englanniksi.

Käännökset tehdään omiin kielikohtaisiin JSON tiedostoihin (kts. kuvio 5), joista arvo voidaan hakea avainten perusteella Vue komponenttiin alla olevan koodi esimerkin tavoin. Paketin \$t -funktio seuraa valittua kieltä ja palauttaa sen perusteella oikeasta käännöstiedostosta parametrinä

saatua avainta vastaavan arvon. Kuvio 4 esittää kuviossa 3 olevan koodin tuotoksen. Myöhemmin on esitelty lyhyesti, kuinka käännökset on toteutettu virheilmoituksiin.

```
<h1>{{ $t('common.clubSettings') }}</h1>
<h3>{{ club.name }}</h3>
<p>{{ $t('common.clubKey') }}: {{ club.clubKey }}</p>
<h3>{{ $t('common.planes') }}</h3>
```

Kuvio 3. Koodiesimerkki käännöksistä

Kerhon asetukset	Club settings
Skydive Jyväskylä Kerhon avain: 1234	Skydive Jyväskylä Club key: 1234
Lentokoneet	Planes

Kuvio 4. Kuvankaappaus kerhon asetus sivusta suomeksi ja englanniksi

```
src > locales > {} en.json > ...
1 {
2   "common": {
3     "edit": "Edit",
4     "delete": "Delete",
5     "settings": "Settings",
6     "language": "Language",
7     "login": "Login",
8   },
9 }

src > locales > {} fi.json > ...
1 {
2   "common": {
3     "edit": "Muokkaa",
4     "delete": "Poista",
5     "settings": "Asetukset",
6     "language": "Kieli",
7     "login": "Kirjaudu",
8   },
9 }
```

Kuvio 5. Kuvankaappaus koodieditorista jossa auki käännöstiedostot

3.2.4 Axios

Axios on erikseen npm pakettina saatava vaihtoehto fetch API:lle, joka on natiivi osa JavaScriptiä ja moderneja selaimia. Molempia käytetään tekemään verkkopyyntöjä palvelimelle, mutta fetch vaa-

tii enemmän koodia saman asian tekemiseen. (Watmore 2021.) Axios tarjoaa muun muassa interceptorit, joilla pystyy helposti puuttumaan pyyntöjen väliin (Getting started n.d.). Muun muassa tätä ominaisuutta käytettiin työssä käsittelemään virheellisiä pyyntöjä ja näyttämään niiden mukaan oikeat virheilmoitukset, josta seuraava koodin osa.

```
// Palautuvia pyyntöjä käsittelevä interceptor, jossa onnistuneet vastaukset
// päästetään sellaisenaan läpi ja virheelliset menevät virheenkäsittelevä funktion kautta
api.interceptors.response.use((response) => response, handleError)

// Osa handleError funktiosta, joka näyttää virheilmoituksen ja palauttaa virheen
...
notification({
  type: 'error',
  // Tunnetuissa virhetilanteissa palvelin palauttaa käynnöksen avaimen
  text: error.response.data.locale || 'error.general',
})
return Promise.reject(error)
...
```

Kuvio 6. Koodiesimerkki verkkopyynnön virheenkäsitteystä

Axios myös käsittelee pyyntöjen JSON datan automaattisesti, kun taas fetch:ssä saman joutuu kirjoittamaan erikseen, josta seuraava esimerkki (Getting started n.d.).

```
// Esimerkki GET-verkkopyyntö Axiosilla
const response = await axios.get('https://example.com/api/users')
// response sisältää JSON muotoisen vastauksen

// Esimerkki GET-verkkopyyntö fetchillä
const response = await fetch('https://example.com/api/users')
const data = await response.json()
// Vastaus joudutaan erikseen muuttamaan JSON muotoon
```

Kuvio 7. Axios ja fetch vertailu koodiesimerkki

3.3 Backend

3.3.1 Rajapinta

Sovellukseen toteutettiin perinteinen REST-rajapinta (API), tietokannan ja selainsovelluksen välille. REST on arkkitehtuurityyli, joka määrittelee tavat, joilla verkkoresursseja voidaan käyttää ja manipuloida verkkopalveluiden avulla. REST arkkitehtuuria käytetään usein sovellusohjelmointirajapintojen suunnittelussa ja toteutuksessa. REST perustuu HTTP-protokollaan, ja se hyödyntää sen perusoperaatioita, kuten GET, POST, PUT ja DELETE, resursseihin pääsemiseksi ja niiden muokkaamiseksi. (Gupta 2023.)

REST-rajapinnassa resurssit tunnistetaan yksilöllisillä URL-osoitteilla (Uniform Resource Locator) (Gupta 2023). Esimerkiksi sovelluksen API:ssa jäsenresurssi voisi olla saatavilla osoitteessa <https://example.com/members/123>, jossa 123 viittaa tiettyyn jäseneseen. Kun halutaan hakea tuotteen tiedot, käytetään HTTP GET -pyyntöä kyseiseen URL-osoitteeseen.

REST-rajapinnan keskeinen etu on sen yksinkertaisuus ja skaalautuvuus. Koska REST hyödyntää olemassa olevia verkkostandardeja ja -protokollia, se on helposti ymmärrettävissä ja otettavissa käyttöön. REST-rajapinnat ovat usein tilattomia, mikä tarkoittaa, että jokainen pyyntö sisältää kaikki tarvittavat tiedot sen käsittelemiseksi, ilman että palvelimen täytyy säilyttää tietoa aiemmista pyynnöistä. (Gupta 2023.)

3.3.2 Express

Express.js, jota usein kutsutaan myös nimellä Express, on monipuolinen verkkosovelluskehys, joka on rakennettu Node.js:n päälle (Express 2017). Express tarjoaa vankan pohjan HTTP-palvelimelle, soveltuen yhden sivun sovelluksille (Single-Page Application), verkkosivuille tai julkisille rajapinnoille (express 2024). Koska tekninen toteutus haluttiin pitää yksinkertaisena, niin oli hyvä valita JavaScript-pohjainen palvelinkehys. Teknologioiden yhtenäinen ohjelmointikielivalinta madaltaa oppimiskynnystä ja nopeuttaa kehitystä.

Expressissä itsessään on vähän ydinominaisuuksia ja onkin tarkoitettu, että ominaisuuksia laajennetaan omilla middlewareiksi kutsutuilla ohjelmapätkillä. Middlewaret ovat käytännössä funkti-

oita, joita voi laittaa käsiteltävien pyyntöjen väliin käsittelemään toistuvia toimintoja kuten käyttäjän oikeuksien tarkistamista. Kuvioissa 8 esimerkki middlewaren käytöstä päätepisteessä (engl. endpoint) ja kuviossa 9 itse middleware funktio. Päätepiste hakee listan jäsenistä, mutta tarkistaa ensin kutsujan oikeudet verifyToken funktiolla, jossa pyynnön mukana saadusta evästeestä poimittu JSON Web Token verifioidaan.

```
router.get('/', verifyToken, async (req, res: Response) => {
  try {
    const member = await Member.find()
    res.send(member)
  } catch (error) {
    res.status(500).send(error)
  }
})
```

Kuvio 8. GET -päätepiste jäsentietojen hakua varten

```
export const verifyToken = (req: Request, res: Response, next: NextFunction) => {
  const token = req.cookies.accessToken
  if (!token) return res.status(401).send('Unauthorized')

  try {
    const user = jwt.verify(
      token,
      process.env.TOKEN_SECRET!
    ) as JwtPayloadWithId
    req.user = user // Return user object
    next()
  } catch (error) {
    res.status(401).send('Invalid token') // 401 are not shown in snackbar
  }
}
```

Kuvio 9. Esimerkki middleware funktiosta

3.3.3 MongoDB

Työssä käytettiin pilvipohjaista MongoDB Atlasta ja tarkemmin sen ilmaisversiota. Valinta perustui hinnan lisäksi sen helppoon ja nopeaan käyttöönottoon sekä sen käytön yksikertaisuuteen. Käyttöä helpottaa entisestään mongoose npm paketti. Tietokantaympäristöä ei tarvinnut erikseen pystyttää paikallisesti kehitystä varten, vaan riitti, että kirjautuu palveluun ja luo uuden tietokannan, johonka sovelluksen rajapinta yhdistää käyttäen samoja tunnuksia. Tämä parantaa sovelluksen liikuteltavuutta ja mahdollisten uusien kehittäjien työn aloittamista, kun tietokannan saa nopeasti käyttöön.

MongoDB on moderni, dokumenttipohjainen tietokanta, joka on suunniteltu käsittelemään suuria määriä jäsentelemätöntä tai osittain jäsenneiltyä dataa. Toisin kuin perinteiset relaatiotietokannat, MongoDB käyttää joustavaa JSON-tyyppistä dokumenttimallia, mikä mahdollistaa nopean ja tehokkaan datan käsittelyn sekä dynaamisen skeeman, joka mukautuu helposti muuttuviin tarpeisiin. Tämä tekee siitä erityisen soveltuvan sovelluksiin, joissa datan rakenne voi muuttua tai laajentua nopeasti. MongoDB tarjoaa myös skaalautuvuutta ja suorituskykyä hajautetun arkkitehtuurinsa ansiosta, mikä mahdollistaa tietokannan kasvattamisen vaivattomasti lisäämällä uusia solmuja. (What is MongoDB? n.d.)

Tuotantokäytössä ilmaisversion rajoitukset saattavat tulla vastaan, mutta asetusten ja tilauksen muuttaminen onnistuu lennosta milloin tahansa. MongoDB Atlasin valinta osoittautui hyödylliseksi myös päivitysten hallinnan kannalta. Palvelun automaattinen päivitysjärjestelmä hoitaa pienemmät päivitykset itsenäisesti, mikä vähentää ylläpidon työmäärää ja minimoi käyttökatkokset. Suurempien versiopäivitysten osalta Atlas antaa mahdollisuuden valita haluamansa version. Tämä ominaisuus parantaa järjestelmän käytettävyyttä ja varmistaa, että tietokantaympäristö pysyy aina ajan tasalla ilman ylimääräistä työtä. (Fully Managed Database Service n.d.)

Tietomallissa jokaiselle kohteelle kuten jäsen, käyttäjä tai maksu, on oma mallinsa ja dokumenttinsa tietokannassa. Kuitenkin lentokoneet on tallennettu kerhon malliin planes avaimen alle ja hyppylinjat ovat tallennettu lentokoneen malliin jumplines avaimen alle. Uuden maksun yhteydessä tallennetaan uusi dokumentti maksutapahtumasta ja niihin on aina vain lukuoikeus. Jokai-

seen malliin sisältyy sen dokumentin luojan ja muokkaajan id, sekä luonnin ja muokkauksen aika-leima. Parantamisen varaa jäi validaation puuttumiseen kontrolleri tasolta, kun validatio toteutettiin ainoastaan mongoosen mallissa.

3.4 TypeScript

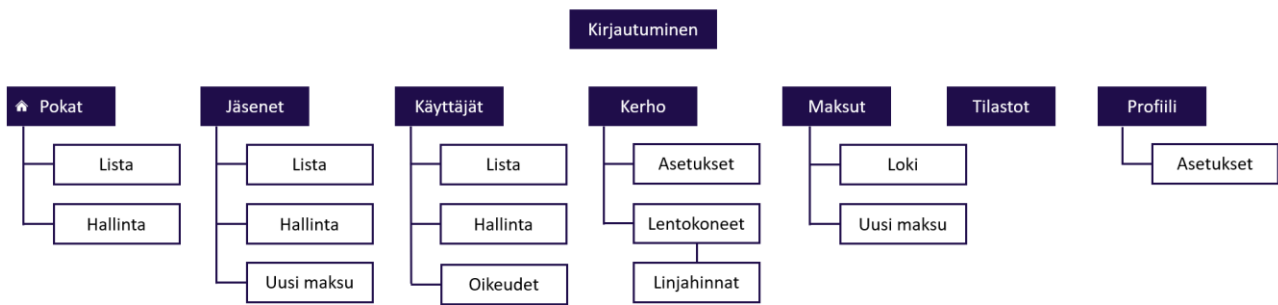
Sovelluksessa käytettiin täysin TypeScriptiä JavaScriptin sijaan, niin selain kuin palvelinpuolella. TypeScript on Microsoftin kehittämä ohjelmointikieli, joka perustuu JavaScriptiin ja laajentaa sen ominaisuuksia tuomalla mukanaan staattisen tyyppityksen. Tämä tarkoittaa, että TypeScriptin avulla kehittäjät voivat määrittää muuttujille ja funktioille tiettyjä tietotyyppisiä, mikä auttaa havaitsemaan ja korjaamaan virheitä jo koodin kirjoitusvaiheessa, ennen kuin koodi ajetaan. TypeScript-koodi käännetään tavalliseksi JavaScriptiksi, joten se toimii kaikissa ympäristöissä, joissa JavaScript toimii. TypeScriptin etuja ovat parempi koodin ennakoitavuus ja luotettavuus, tehokkaampi virheiden hallinta ja parempi yhteistyö isommissa kehittäjätiimeissä. (Introduction to TypeScript 2024.)

4 Sovelluksen suunnittelu ja toteuttaminen

Sovellus suunniteltiin aiemmin käytössä olleen SkyWin järjestelmän pohjalta, keskittyen eniten korjaamaan niitä asioita, joita käyttäjät ovat kertoneet suurimmiksi ongelmiksi.

4.1 Sivustokartta

Sivustokartta on keskeinen työkalu sivuston tai sovelluksen suunnittelussa, joka kuvaa eri osien väliset yhteydet ja rakenteen. Sen avulla voidaan hahmottaa, miten käyttäjät navigoivat sivustolla ja löytävät tarvitsemansa tiedot tai palvelut. Sivustokartan päätarkoitus on luoda käyttäjäystävällinen ja helposti navigoitava verkkosivusto. (Mitsiakina 2023.)



Kuvio 10. Sovelluksen sivustokartta

Kuviossa 10 on sovelluksen sivustokartta, jossa on esitelty yltason sivut ja niiden alta löytyvät näkymät. Sovellus ei tarjoa mitään tietoja ilman kirjautumista, joten aloitusnäkö on kirjautumista varten ja kirjautuneena oletuksena on Pokat -näkö. Pokat, Tilastot ja Profiili ovat ainoita sivuja, jotka näkyvät kaikille käyttäjille ja loput ovat vain ylläpitäjien nähtävissä. Henkilötietoja on käsiteltävä asianmukaisesti ja luottamuksellisesti, joten niitä sisältäviin sivuihin oltava rajattu pääsy (Henkilötietojen käsittely n.d.).

4.2 Rautalankamalli

Rautalankamallin luominen verkkosivuston rakentamisen tai uudelleensuunnittelun yhteydessä antaa mahdollisuuden keskittyä käyttäjäkokemukseen erillisenä kehityksen osana. Rautalankamalli mahdollistaa sivun asettelun ja käyttäjäpolkujen testaamisen, jolloin voidaan nähdä, miten uusi verkkosivusto toimii ja havaita mahdolliset virheet. (Osman 2023.)

Rautalankamalli on verkkosivuston suunnittelun keskeisten ominaisuuksien ja navigoinnin kartta. Se antaa käsityksen sivuston toiminnallisuudesta ennen visuaalisten suunnitteluelementtien, kuten sisällön ja värimaailmojen, huomioimista. Rautalankamalli myös tarjoaa käytännöllisen kartan projektin etenemisestä, jolloin tiimin jäsenet voivat nähdä, mihin kaikki elementit sijoittuvat heidän suorittaessaan siihen liittyviä tehtäviä. (Osman 2023.)

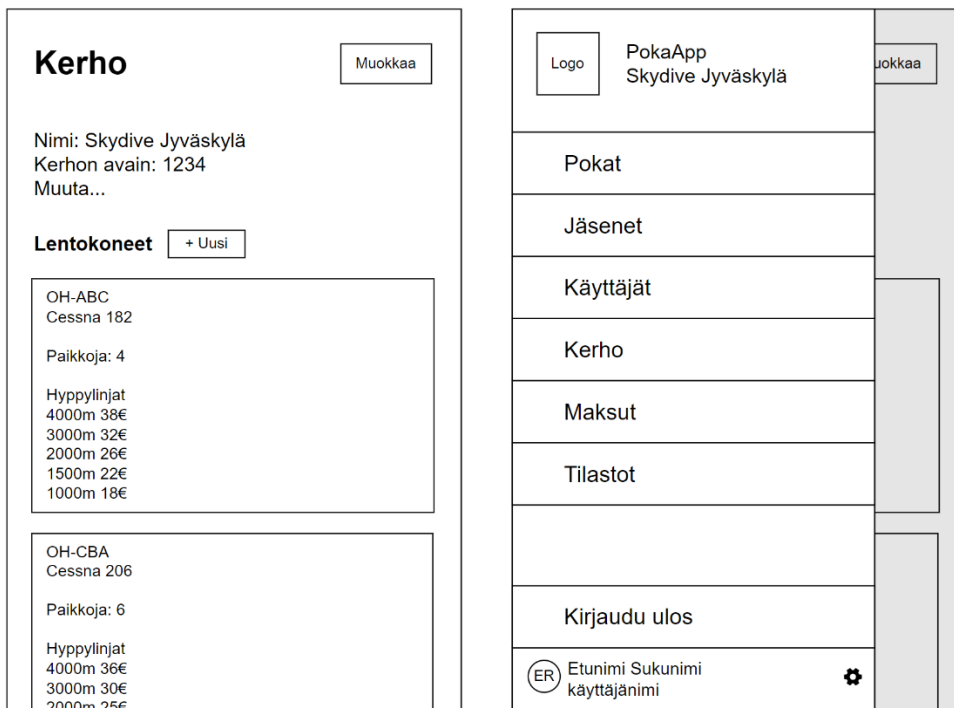
Logo PokaApp Skydive Jyväskylä	<h2>Pokat</h2> <p>📅 24.5.2024</p> <div> <p>1 OH-ABC, Lentäjä: Nimi</p> <p>Hyppääjä 1 Hyppääjä 2 Hyppääjä 3 Hyppääjä 4</p> </div> <div> <p>2 OH-ABC, Lentäjä: Nimi</p> <p>Hyppääjä 1 Hyppääjä 2 Hyppääjä 3 Hyppääjä 4</p> </div> <div> <p>3 OH-ABC, Lentäjä: Nimi</p> <p>Hyppääjä 1 Hyppääjä 2 Hyppääjä 3 Hyppääjä 4</p> </div> <div> <p>4 OH-ABC, Lentäjä: Nimi</p> <p>Hyppääjä 1 Hyppääjä 2 +</p> </div> <p>+ Uusi</p>
Pokat	
Jäsenet	
Käyttäjät	
Kerho	
Maksut	
Tilastot	
Kirjaudu ulos	
ER Etunimi Sukunimi käyttäjänimi	

Kuvio 11. Rautalankamalli oletusnäelmästä

Logo PokaApp Skydive Jyväskylä	<h2>Kerho</h2> <p>Nimi: Skydive Jyväskylä Kerhon avain: 1234 Muuta...</p> <p>Lentokoneet + Uusi</p> <table border="1"> <tr> <td> <p>OH-ABC Cessna 182</p> <p>Paikkoja: 4</p> <p>Hyppylinjat 4000m 39€ 3000m 32€ 2000m 26€ 1500m 22€ 1000m 18€</p> </td> <td> <p>OH-CBA Cessna 206</p> <p>Paikkoja: 6</p> <p>Hyppylinjat 4000m 36€ 3000m 30€ 2000m 25€ 1000m 18€</p> </td> <td> <p>Rekisteri Malli</p> <p>Paikkoja: 3</p> <p>Hyppylinjat 3000m 30€ 2000m 25€ 1600m 22€ 1200m 20€ 1000m 18€</p> </td> </tr> </table>	<p>OH-ABC Cessna 182</p> <p>Paikkoja: 4</p> <p>Hyppylinjat 4000m 39€ 3000m 32€ 2000m 26€ 1500m 22€ 1000m 18€</p>	<p>OH-CBA Cessna 206</p> <p>Paikkoja: 6</p> <p>Hyppylinjat 4000m 36€ 3000m 30€ 2000m 25€ 1000m 18€</p>	<p>Rekisteri Malli</p> <p>Paikkoja: 3</p> <p>Hyppylinjat 3000m 30€ 2000m 25€ 1600m 22€ 1200m 20€ 1000m 18€</p>	Muokkaa
<p>OH-ABC Cessna 182</p> <p>Paikkoja: 4</p> <p>Hyppylinjat 4000m 39€ 3000m 32€ 2000m 26€ 1500m 22€ 1000m 18€</p>	<p>OH-CBA Cessna 206</p> <p>Paikkoja: 6</p> <p>Hyppylinjat 4000m 36€ 3000m 30€ 2000m 25€ 1000m 18€</p>	<p>Rekisteri Malli</p> <p>Paikkoja: 3</p> <p>Hyppylinjat 3000m 30€ 2000m 25€ 1600m 22€ 1200m 20€ 1000m 18€</p>			
Pokat					
Jäsenet					
Käyttäjät					
Kerho					
Maksut					
Tilastot					
Kirjaudu ulos					
ER Etunimi Sukunimi käyttäjänimi					

Kuvio 12. Rautalankamalli Kerho -näelmästä

Koska sovelluksessa on paljon listattua tietoa, niin on haluttu tuoda nähtäville mahdollisimman paljon rivejä kerralla ja se on huomioitu asettelussa. Navigaatio sijoitettiin vasempaan reunaan, koska sijoittaminen sivun yläreunaan veisi suhteellisen paljon tilaa riveiltä etenkin pienemmillä näytöillä. Toinen syy sijoittaa navigaatio vasempaan reunaan, on sen suoraviivainen asettuminen mobiili resoluutioille. Kuten alla kuviossa 13, navigaation saa silloin piilotettua kokonaan ja on helpposti sormella vedettävissä esiin. Ylös sijoitetun navigaation saisi piiloon, mutta se tulisi esiin ylöspäin rullatessa viimeistään ylhäällä, joka taas veisi tilaa sisällöltä. Ylänavigaatioon joutuisi myös tekemään uudelleen asettelua mobiili resoluutiota varten, sillä kuvioden 10 ja 11 mukaisten suunnitelmien mukaan linkkejä eri näkyymiin olisi ainakin kuusi, riippuen sisällyttäväkö käyttäjän toiminnot erillisen valikon alla.



Kuvio 13. Rautalankamalli Kerho -näköymästä mobiili resoluutiolla

Sovelluksen suunnittelussa, tärkeänä lähtökohtana oli sen toimivuus mobiililaitteilla ilman tarvetta monimutkaisille säätötoimenpiteille. Käyttäjäkokemuksen parantamiseksi sovelluksen käyttöliittymä rakennettiin responsiiviseksi, mikä tarkoittaa, että navigaation lisäksi kaikki muutkin elementit mukautuvat automaattisesti erikokoisille näytöille. Tämä saavutettiin käyttämällä joustavia asetteluratkaisuja ja suhteellisia mittoja, jotka varmistavat, että sisältö on aina selkeästi nähtävillä ja helposti käytettävissä.

Erytystä huomiota kiinnitettiin lomakkeisiin ja muihin käyttöliittymäelementteihin, jotta niiden asettelu toimisi vaivattomasti eri leveyksillä. Näiden komponenttien suunnittelussa varmistettiin, että ne ovat helposti saavutettavissa ja käytettävissä sormella, mikä on olennaista mobiililaitteiden käyttäjille. Painikkeet ja valikot sijoitettiin riittävän etäälle toisistaan virhepainallusten minimoimiseksi. Lisäksi interaktiivisten elementtien koko optimoitiin niin, että ne ovat tarpeeksi suuria myös sormella käytettäväksi mutta eivät vie liikaa tilaa näytöltä.

The image displays two versions of a mobile membership form. The left version is a desktop view, and the right version is a mobile view. Both forms are titled 'Jäsen' (Member) and show the following fields and controls:

- Etunimi*** (First name) and **Sukunimi*** (Last name) text input fields.
- Sähköposti** (Email) and **Puhelin** (Phone) text input fields.
- Rooli** (Role) dropdown menu with 'Kelppari' selected.
- Exit paino** (Exit fee) numeric input field with a value of 80 and minus/plus buttons.
- Lisenssi** (License) dropdown menu with 'A' selected.
- Kalustomestari** (Equipment manager) dropdown menu.
- Lähiomaisen nimi** (Emergency contact name) and **Lähiomaisen puhelin** (Emergency contact phone) text input fields.
- Jäsen on** (Member is) section with checkboxes for **VPK**, **PLHM**, **NOVA-HM**, and **Tandem-HM**.
- Footer text: **Luonut Eemil Rantala 7.5.2024 klo 15.56** and **Muokannut Eemil Rantala 27.5.2024 klo 19.54**.
- Bottom bar with **PERUUTA** (Cancel) and **TALLENNA** (Save) buttons.

Kuvio 14. Toteutettu jäsentietolomake täysikokoisena ja mobiilissa

Toteutettua käyttöliittymää esiteltiin ammatti UI/UX-suunnittelijalle, joka totesi pohjan olevan hyvällä mallilla (Roppola 2024).

4.3 Tyylit

4.3.1 Kirjasintyyppi

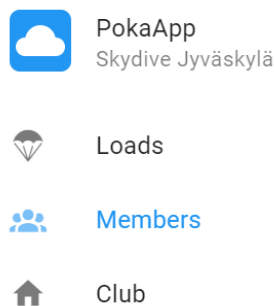
Sovelluksen kirjasintyyppiä valittiin Roboto, sillä sen käytettävyys on laaja-alainen ja sitä voi käyttää yhdessä Roboto Condensed- ja Roboto Slab -perheiden kanssa, mikä tarjoaa joustavuutta ja yhtenäisyyttä eri tekstielementtien välillä. Lisäksi Roboto on lisensoitu Apache License, Version 2.0

-lisenssillä, mikä tarkoittaa, että sitä voi käyttää vapaasti niin kaupallisissa kuin ei-kaupallisissakin projekteissa, sekä painetussa että digitaalisessa muodossa. (Roboto n.d.).

Roboto oli ylivoimaisesti käytetyin fontti Google Fonts -palvelussa. Christian Robertsonin suunnittelema Roboto oli kerännyt viimeisen vuoden aikana 3,5 biljoonaa näyttökertaa, mikä oli huomattavasti enemmän kuin toiseksi sijoittuneen Open Sans -fontin 1,4 biljoonaa näyttökertaa. Vaikka Roboton käyttö oli hieman laskenut viisi prosenttia edellisvuodesta, se säilyttää silti huomattavan etumatkan kilpailijoihinsa nähden. Tämä osoittaa Roboton vankan suosion ja laajan hyväksynnän niin verkkosivujen suunnittelijoiden kuin käyttäjien keskuudessa. (Google Fonts Analytics 2024.)

4.3.2 Värit

Sovelluksen pääväriksi valittiin vaalean sininen, sillä sen ajateltiin kuvastavan sinistä taivasta ja liittyvän sitä kautta laskuvarjohyppäämiseen. Sen sävy poimittiin suoraan Vuetifyn väripaletista, jonka avulla saatiin väri toimimaan hyvin muiden sovelluksen värien kanssa. Pääväriä käytettiin logossa pilvi kuvakkeen taustavärinä (kts. Kuvio 15). Pääväri myös korostaa aktiivista valintaa ja tiettyjä interaktiivisia painikkeita.



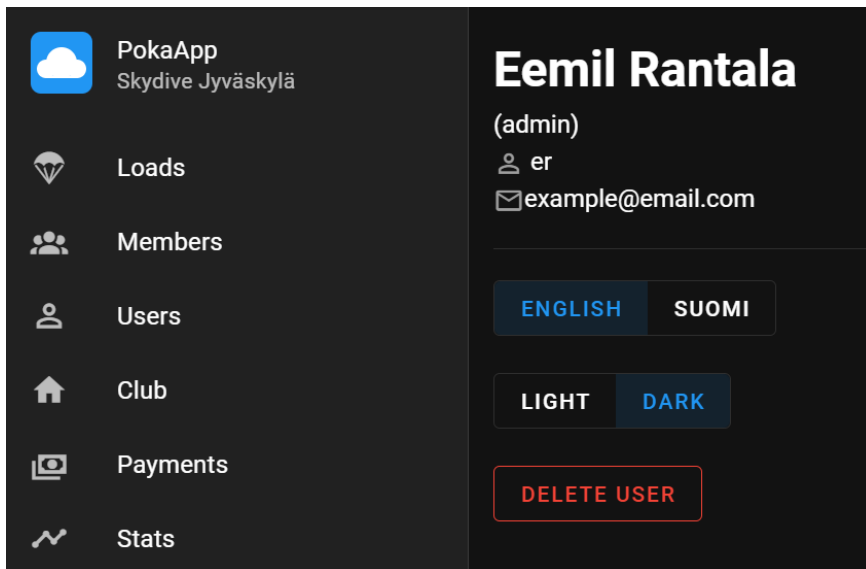
Members

Name ↓

Eemil Rantala

Kuvio 15. Kuvankaappaus sovelluksen navigaatiosta ja osasta jäsensivua

Vuetifyn avulla sovellukseen saatiin toteutettua vaivattomasti myös tumma teema. Värien valitseminen vain Vuetifyn omasta paletista sai kaikki komponentit toimimaan suoraan molemmilla teemoilla ilman muutoksia. Kuten kielivalinta, myös oletusteema poimitaan alustavasti selaimesta, jos sovelluksessa ei ole valittuna tiettyä teemaa. Kuviossa 16 kuvankaappaus käyttäjän asetuksista tummalla teemalla.



Kuvio 16. Kuvankaappaus käyttäjän asetuksista tummalla teemalla

4.3.3 Kuvakkeet

Työssä käytettiin Material Icons paketin kuvakkeita, jotka ovat Material Symbolsin kanssa Googlen kehittämiä ikonisarjoja, jotka on suunniteltu parantamaan käyttöliittymien visuaalista ilmettä. Material Icons on klassinen sarja, joka sisältää erilaisia tyylejä, kuten ääriiviiva, täytetyt, pyöristetyt, terävät ja kaksiväriset ikonit. Vaikka tämä sarja ei enää saa aktiivisia päivityksiä, se on edelleen laajalti käytetty. Vuonna 2022 esitelty Material Symbols vie kuvakkeet askeleen pidemmälle hyödyntämällä muuttuvan fontin teknologiaa, mikä mahdollistaa ikonien yksityiskohtaisen muokkauksen koon, painon, asteen ja täytön osalta. Molemmat ikonisarjat noudattavat Material Design -ohjeistoa ja ovat vapaasti saatavilla Google Fonts -alustalla. (Material Design Icons 2024.) Kyseisiä kuvakkeita muun muassa kuvion 16 navigaatiassa.

4.4 Käytännön asioita toteutuksessa

4.4.1 Rekisteröityminen

Kerholle asetettiin uniikki avain, joka mahdollistaa uusien käyttäjien liittymisen ilman erillistä hyväksyntää. Näin ollen kerhon vetäjät voivat yksinkertaisesti jakaa tämän avaimen uusille jäsenille, mikä poistaa tarpeen manuaaliselle hyväksymisprosessille ja tekee jäsenyyksien hallinnasta huo-

mattavasti sujuvampaa. Käyttäjää ei myöskään pysty luomaan ilman kerhon avainta, jotta järjestelmään ei tulisi ylimääräisiä tai haitallisia käyttäjiä. Käyttäjät ja jäsenet ovat erillään toisistaan, koska kaikkia käyttäjiä ei voida velvoittaa kirjautumaan sovellukseen.

4.4.2 Käyttöoikeudet

Sovellukseen tehtiin kaksi käyttäjäroolia: normaali ja ylläpitäjä. Normaalikäyttäjät voivat käyttää sovelluksen perustoimintoja, kuten muokata pokalista ja nähdä omia tietoja, kun taas ylläpitäjät voivat hallinnoida kaikkea. Käyttöoikeuksia voisi kehittää siten, että yksittäisille käyttäjille saisi oikeuksia tiettyihin toimintoihin, mutta alkuun vain selkeät kaksi roolia. Normaalikäyttäjät eivät pääse katsomaan muiden jäsenten tietoja, vaan ainoastaan omiaan. Käyttäjän ja jäsenen tietojen yhdistäminen onnistuisi silloin automaattisesti, kun molemmissa on sama sähköpostiosoite. Poikkeus tilanteissa ylläpitäjä voisi yhdistää käyttäjän jäseneseen, tämän päästäkseen näkemään esimerkiksi oman saldonsa.

4.4.3 Lentokoneet ja hinnoittelu

Tietomallissa koneet on ryhmitelty kerhon alle, ja jokaisella koneella on omat hyppylinjat, mikä mahdollistaa erilaisen hinnoittelun vierailevien koneiden välillä. Muussa tapauksessa joutuisi olemaan kaksi erillistä järjestelmää samaan aikaan. Kuviossa 17 kuvankaappaus kerhon asetuksista, jossa listattuna kaksi lentokonetta hintoineen. Asettelu on toteutettu kuvion 12 rautalankamallin mukaisesti.

Kerhon asetukset

Skydive Jyväskylä

Kerhon avain: 1234

Lentokoneet

+ UUSI

C182

OH-CAD

Paikkoja 4

Vapaa paino 400

Hyppylinjat

Korkeus	Hinta
2000	21
1600	19
1200	18
1000	17
2500	25

Cessna T206H

OH-ARR

Paikkoja 6

Vapaa paino 660

Hyppylinjat

Korkeus	Hinta
4000	35
3000	30
600	0
300	0
200	1

Muokannut Eemil Rantala 16.5.2024 klo 13.48

Kuvio 17. Kuvankaappaus toteutuksen kerhon asetussivusta

Lentokoneet + UUSI

Lentokone

Malli: C182 Rekisteritunnus: OH-CAD

Paikkoja: 4 Vapaa paino: 400

Hyppylinjat + UUSI

Korkeus: 2000	Hinta: 21	-
Korkeus: 1600	Hinta: 19	-
Korkeus: 1200	Hinta: 18	-
Korkeus: 1000	Hinta: 17	-
Korkeus: 2500	Hinta: 25	-

PERUUTA TALLENNA

Kuvio 18. Kuvankaappaus lentokoneen muokkauslomakkeesta

4.4.4 Pokalista

Pokalistan järjestyksen hallinta piti olla mahdollista raahaamalla, mikä tarjoaa käyttäjille intuitiivisen ja helpon tavan muokata listan järjestystä. Tämä ominaisuus kehitettiin käyttäjäpalautteen perusteella, sillä aiemmassa järjestelmässä SkyWinissä, tämä toiminnallisuus puuttui ja koettiin merkittäväksi puutteeksi. Kuviossa 19 pokalista, jossa järjestystä voi muuttaa raahaamalla kahvana toimivasta sinisestä sivupalkista. Yksittäisiä hyppääjiä voi myös raahata pokien välillä.

Pokat

1 OH-CAD, Lentäjä A
Emil R Hyppääjä A Etunimi S Hyppääjä B

2 OH-CAD, Lentäjä A
Tandem Hyppääjä D + Uusi

3 OH-CAD, Lentäjä A
+ Uusi

UUSI

Kuvio 19. Kuvankaappaus toteutuksen pokalistasta

4.4.5 Maksut

Maksuja voi lisätä maksut sivulla tai suoraan jäsenlistasta kuten kuviossa 20. Maksuja voi tehdä vain ylläpitäjät, mutta käyttäjä voi aina nähdä kuitenkin oman saldonsa. Kuvion 21 lomakkeen kommenttikenttään voi vapaasti laittaa tarkentavia tietoja esimerkiksi virheitä korjattaessa. Kaikista maksutapahtumista pidetään lokia. Sovellus itsessään ei ota maksuja vastaan, vaan saldoa on ostettava joko verkkokaupasta tai kerhon kassalta. Maksujen automatisoimiseksi voisi tulevaisuudessa kehittää integraation kerhon Shopify verkkokauppaan.

Nimi ↓	Saldo	Lisenssi	Rooli
Eemil Rantala	24.00 €	B	Kelppari
Etunimi Sukunimi	3.00 €		Lentäjä
Jäsen A	0.00 €		PL-oppilas
Jäsen B	4.00 €	A	Kelppari
Jäsen C	0.00 €	A	Kelppari

Avaa
Uusi maksu
Poista

Kuvio 20. Kuvankaappaus maksun tekemisestä jäsenlistassa

Maksu

Jäsenelle*
Eemil Rantala Saldo: 24.00 €

Määrä €*
100

Maksutapa*
Kortti

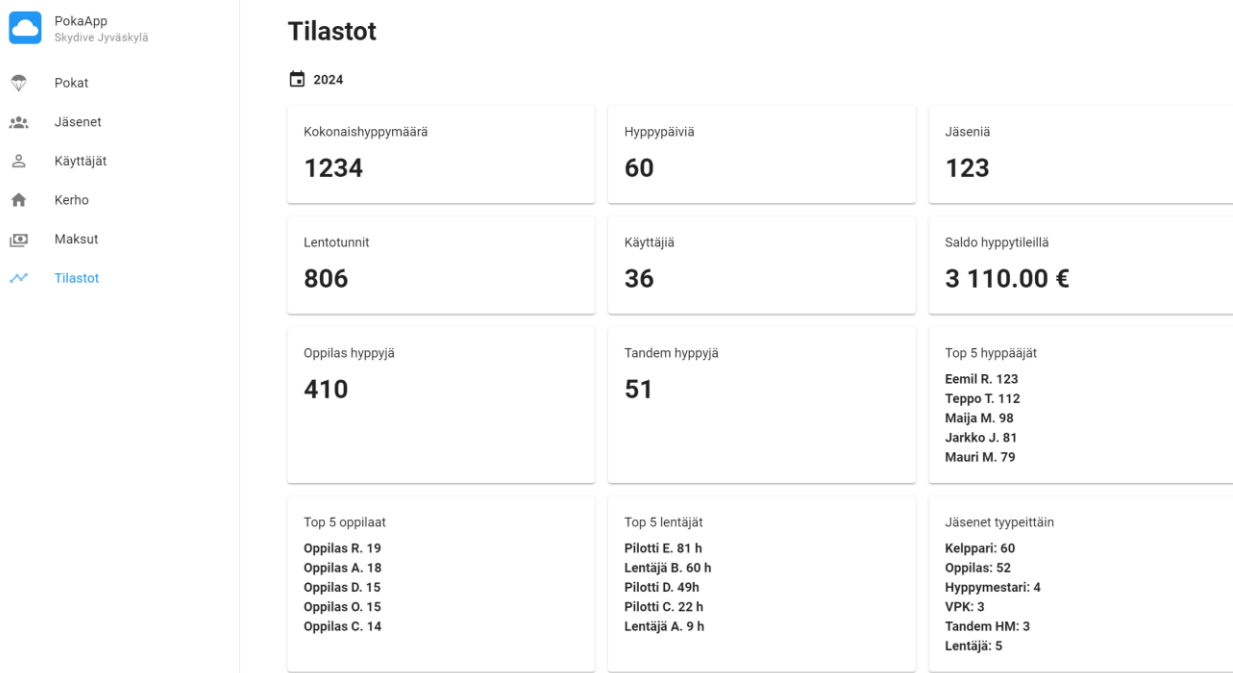
Kommentti

PERUUTA VAHVISTA

Kuvio 21. Kuvankaappaus maksulomakkeesta

4.4.6 Tilastot

Tilastot sivulta on nähtävissä vuoden avainlukuja. Myös kattavampien vuosiraporttien tulostus hoi-
tuu samalta sivulta. Kuviossa 22 kattavampi ylläpitäjille tarkoitettu tilastonäkymä. Normaalikäyttä-
jille on tarjolla suppeampi versio tilastoista, josta on poistettu muun muassa rahalukemat.



Kuvio 22. Kuvankaappaus kuvitteellisista tilastoista

4.5 Turvallisuus

4.5.1 Rekisteröityminen

Käyttäjän rekisteröitymiseen vaaditaan muun muassa oikea sähköposti, salasana ja kerhon avain. Sähköpostiin tulee vahvistus viesti, jonka kautta käyttäjätunnus on vahvistettava kirjautuakseen palveluun. Kerhon avaimen lisäksi tällä pienennetään turhien ja haitallisten käyttäjien riskiä.

Sähköposti

Käyttäjänimi

Nimi

Salasana

Kerhon avain

REKISTERÖIDY

[Kirjaudu](#)

Kuvio 23. Kuvankaappaus rekisteröitymislomakkeesta

4.5.2 Käyttäjätunnus ja salasana

Jokaisen käyttäjän salasana on salattu suolatulla hashilla ennen tietokantaan tallentamista. Näin varmistetaan yksilöllinen ja erittäin vaikeasti murrettavissa oleva salaus. Hash luodaan palvelimella bcrypt nimisellä npm paketilla. Salasanan murtamista arvaamalla vaikeuttaa se, että salasanan minimi pituus on kahdeksan merkkiä ja siinä on oltava joukossa erikoismerkkejä.

Salasanan palautuksessa on noudatettu OWASP:n Forgot Password Cheat Sheet (2024) suosituksia. Huomioitavaa oli, ettei salasanaa palauttamalla pysty saamaan tietoa onko kohdesähköpostilla olemassa käyttäjää, vaan näkymän on oltava joka tilanteessa samanlainen. Onnistuneessa palautuksessa sähköpostiin tulee palautuslinkki, jossa on uniikki kertakäyttöinen token, jonka avulla linkin oikeellisuus varmistetaan. Token on asetettu vanhenemaan vuorokauden kuluessa. Käyttäjän olemassaoloa ei pysty myöskään todentamaan verkkopyynnön kestosta, sillä tietokantakysely tehdään vasta pyyntöön vastaamisen jälkeen.

4.5.3 Käyttäjän todentaminen

Käyttäjän todentaminen on toteutettu JSON Web Tokenilla (JWT), joka perustuu avoimeen RFC 7519 standardiin. Se määrittelee tavan välittää tietoa JSON-objektina osapuolten välillä. Tämä tieto voidaan varmistaa digitaalisella allekirjoituksella, jonka avain on tiedossa vain palvelimella. JWT voidaan allekirjoittaa käyttämällä salaista avainta HMAC-algoritmilla, kuten työssä. Allekirjoitus onnistuu myös julkisen tai yksityisen avainparin avulla (RSA tai ECDSA). Kun käyttäjä kirjautuu sisään, jokainen seuraava pyyntö sisältää JWT:n, jolloin palvelin antaa käyttöön kyseiselle tunnuk-selle sallittuja resursseja. (Introduction to JSON Web Tokens n.d.)

Kirjautuessa palvelin asettaa evästeisiin kaksi JSON Web Tokenia. Ensimmäinen on varsinainen tunnistamiseen käytetty JWT (engl. access token) ja toinen niin sanottu päivitys token (engl. refresh token). Varsinainen token vanhenee puolessa tunnissa ja sen vanhettua palvelin luo uuden tokenin, jos päivitys token on voimassa. Päivitys token on asetettu vanhenemaan seitsemän päivän kuluessa.

Tokenin kuormassa (engl. payload) kuljetetaan käyttäjän id:tä ja totuusarvoa siitä onko käyttäjä ylläpitäjä. Mukana kulkevan Id:n avulla voidaan helposti merkata tietokantaan, kuka on tehnyt viimeisimmän muutoksen. Ylläpitäjä totuusarvosta voidaan tarkistaa oikeuksia palvelinpuolella ilman tietokantakyselyä, sillä tokenin vahvistaminen riittää takaamaan tiedon aitouden.

JWT:n käyttö on erityisen suosittua, koska se tarjoaa tehokkaan tavan välittää tietoa turvallisesti ja vähentää palvelimen kuormitusta pitämällä tunnukset tiiviinä ja helposti käsiteltävinä HTML- ja HTTP-ympäristöissä. JSON Web Tokenit ovat käytännöllisiä erityisesti modernissa web-kehityksessä, missä tehokkuus ja turvallisuus ovat ensiarvoisen tärkeitä.

Kehityksen aikana JWT toteutuksen turvallisuutta parannettiin seuraavilla keinoilla. Käyttäjän henkilötiedot poistettiin tokenin kuormasta ja tilalle jätettiin ainoastaan käyttäjän id. Tällä estettiin henkilötietojen vuotamisen mahdollisuus tokenin osalta. Toinen parannus tehtiin turvaamaan tokenin säilytystä, kun aluksi token tallennettiin selaimen localStorageeen, johonka mahdollisella hyökkääjällä voisi olla pääsy. Tokenien säilytys siirrettiin selaimen evästeisiin, sillä evästeitä ei pysty lukemaan asiakasohjelmassa JavaScriptillä, kun evästeeseen on asetettu httpOnly lippu. (Patel 2023.)

4.5.4 HTTPS

Tuotantoversioon tarvitaan SSL sertifikaatti, jolla saadaan käyttöön HTTPS-protokolla. Kerhon kaikki muutkin palvelut käyttävät HTTPS:ää ja se onkin lähes pakollinen nykypäivänä, sillä useimmat selaimet pyrkivät estämään pääsyn salaamattomille sivuille.

5 Lopputulos

Opinnäytetyön tuloksena suunniteltiin ja toteutettiin nykyaikaisen manifestijärjestelmän beeta versio Jyväskylän laskuvarjokerholle. Sovelluksesta saatiin toiveiden mukainen ja varsinkin käyttöliittymän suunnittelussa onnistuttiin luomaan toimiva ratkaisu eri kokoisille laitteille. Kaikki pakolliset ydinominaisuudet saatiin toteutettua. Käyttäjätestausta ei ole tehty vielä raportin kirjoitus hetkellä. Sovellus on tarkoitus ottaa käyttöön kaudelle 2025 ja testaus- ja kehitystyötä jatketaan siihen asti. Myös vapaaehtoisia kehittäjiä etsitään mukaan projektiin. Seuraavana askeleena olisi saada jaettava testausympäristö, josta saada palautetta jatkokehitystä varten.

Työssä kehittämisen varaa jäi etenkin sovelluksen julkaisuun liittyen, sillä sitä ajettiin vain paikallisessa ympäristössä. Opit pilvi-infrastruktuurista jäi vain teorian lukemisen tasolle, eikä sitä osuutta käsitelty työssä lainkaan. Osasyynä oli, että sovelluskehityksen hitaus yllätti jälleen, jonka kompensimiseksi projektia pyöritettiin paikallisesti, jotta pakolliset toiminnot saataisiin valmiiksi aikataulussa. Palvelinpuolella parantamisen varaa jäi rakenteen selkeyteen, vaikka tietomallit, middlewaaret ja päätepisteet on eritelty omiin tasoihinsa, voisi niitä selkeyttää entisestään. Esimerkiksi kontrolleri tasolla on joukossa sinne kuulumatonta logiikkaa, jonka voisi eriyttää muualle. Myös validointia tulisi lisätä palvelinpuolelle turvallisuuden ja toimivuuden parantamiseksi. Nämä puutteet johtuivat tekijän kokemattomuudesta palvelinpuolen ohjelmien kanssa.

Isona pitkänajan tavoitteena olisi toteuttaa sovelluksesta suurempi ekosysteemi, jossa olisi integraatiot verkkokauppaan ja kassaan, sekä muun muassa waiver- ja tandemlomakkeet. Kaukainen kunnianhimoinen tavoite olisi kaupallistaa sovellus SaaS-tuotteeksi.

Opinnäytetyössä käytettiin sen tekijälle osittain tuttuja teknologioita, mutta myös uutta opeteltavaa oli paljon. Oppimista tapahtui eniten kehitystyön alkupuolella, sillä uuden sovelluksen perustan tekeminen oli tekijälle täysin vierasta. Erityisesti toimivan ja turvallisen kirjautumisen tekeminen toi haasteita JSON Web Token toteutusta ymmärtäessä. Alkuun aikaa vei myös koko teknologiapinin saaminen keskustelemaan keskenään, sillä muun muassa väärät CORS asetukset estivät pyyntöjen kulkua alussa. Neuvoja ja kehitysideoita sai myös paremman Vue-sovelluksen rakenteeseen liittyen.

Kaiken kaikkiaan työ on onnistunut ja antoi tekijälle runsaasti oppeja, joista on hyötyä tulevaisuudessa. Myös laskuvarjokerhon johtokunta oli tyytyväinen näkemiinsä esittelyihin. Yleisesti voidaan todeta kerholaisten odottavan innolla uuden sovelluksen käyttöönottoa.

Lähteet

About React and Next.js. 2024. Next.js verkkosivut. Viitattu 2.5.2024.
<https://nextjs.org/learn/react-foundations/what-is-react-and-nextjs>

Arushi. 2024. Choosing Between Next.js and Vue.js. Viitattu 2.5.2024. <https://medium.com/@1032210476/choosing-between-next-js-and-vue-js-467abdd85b18>

Burple Me. 2024. Burblen verkkosivu. Viitattu 20.4.2024. <https://www.burplesoftware.com/features/burple-me/>

Burplesoftware. 2024. Burblen verkkosivu. Viitattu 17.4.2024. <https://www.burplesoftware.com/>

Colors. 2024. Vuetify:n dokumentaatio. Viitattu 15.5.2024. <https://vuetifyjs.com/en/styles/colors/#material-colors>

Daityari S. 2023. Angular vs React vs Vue: Which Framework to Choose. Viitattu 2.5.2024.
<https://wpshout.com/angular-vs-vue-vs-react/>

Express. 2017. Express.js kotisivu. Viitattu 20.5.2024. <https://expressjs.com/>

express. 2024. Express.js npm sivu. Viitattu 20.5.2024. <https://www.npmjs.com/package/express>

FAQ. 2023. Vuen usein kysytyt kysymykset. Viitattu 2.5.2024. <https://vuejs.org/about/faq.html>

Forgot Password Cheat Sheet. 2024. OWASP:n parhaat käytännöt salasanan palautukseen. Viitattu 3.6.2024. https://cheatsheetseries.owasp.org/cheatsheets/Forgot_Password_Cheat_Sheet.html

Fully Managed Database Service. N.d. MongoDB:n tietosivu Atlas palvelun ominaisuuksista. Viitattu 23.5.2024. <https://www.mongodb.com/cloud/atlas/efficiency>

Gerschwiler, F. 2024. Burble pricing for small DZ. Sähköpostiviesti 18.4.2024. Vastaanottaja E. Rantala.

Getting started. N.d. Esittely axiosin verkkosivulla. Viitattu 13.5.2024. <https://axios-http.com/docs/intro>

Gupta, L. 2023. What is REST?. Viitattu 21.5.2024. <https://restfulapi.net/>

Google Fonts Analytics. 2024. Tilasto käytetyimmistä kirjasintyypeistä Google Fonts palvelussa. Viitattu 26.5.2024. <https://fonts.google.com/analytics>

Haapsalo, V. 2024. Jyväskylän laskuvarjokerhon jäsen. Keskustelu 13.4.2024

Henkilötietojen käsittely. N.d. Tietosuojavaltuutetun toimisto -verkkosivu. Viitattu 23.5.2024.
<https://tietosuoja.fi/henkilotietojen-kasittely>

Hyppääminen Utissa. 2021. Skydive Finland verkkosivu. Viitattu 20.4.2024. <https://www.skydivefinland.fi/hyppaajille/#hyppaaminenutissa>

Introduction. 2024. Esittely Vuen verkkosivulla. Viitattu 2.5.2024. <https://vuejs.org/guide/introduction>

Introduction. N.d.a Esittely Vuetifyn verkkosivulla. Viitattu 3.5.2024. <https://vuetifyjs.com/en/introduction/why-vuetify/>

Introduction. N.d.b. Esittely Material Designin verkkosivulla. Viitattu 3.5.2024. <https://m2.material.io/design/introduction>

Introduction to JSON Web Tokens. N.d. Esittely JWT:n sivuilla. Viitattu 28.5.2024. <https://jwt.io/introduction>

Introduction to TypeScript. 2024. Artikkelit GeeksforGeeks sivustolla. Viitattu 21.5.2024. <https://www.geeksforgeeks.org/introduction-to-typescript/>

Jyväskylän Laskuvarjokerho ry:n säännöt. N.d. Skydive Jyväskylän verkkosivu. Viitattu 19.4.2024. <https://skydivejkl.fi/pages/jyvaskylan-laskuvarjokerho-ry-saannot>

Kerhon esittely. N.d. Skydive Jyväskylän verkkosivu. Viitattu 19.4.2024. <https://skydivejkl.fi/pages/kerhon-esittely>

Lähde, S. 2024. Varapuheenjohtaja. Jyväskylän laskuvarjokerho ry. Keskustelu 19.4.2024.

Material Design Icons. 2024. Material Design Icons GitHub sivu. Viitattu 27.5.2024. <https://github.com/google/material-design-icons>

Mitsiakina, J. 2023. Website Development Process: Full Guide in 7 Steps. Viitattu 23.5.2024. <https://xbsoftware.com/blog/website-development-process-full-guide/>

Netlify Core. 2024. Palvelun hinnasto Netlifyn verkkosivuilla. Viitattu 18.4.2024. <https://www.netlify.com/pricing>

npm. 2024. Node Package Managerin verkkosivu. Viitattu 21.4.2024. <https://www.npmjs.com/>

Osman, M. 2023. Website Wireframe Beginner's Guide: Processes, Tools, & Examples. Viitattu 24.5.2024. <https://blog.hubspot.com/website/website-wireframe>

Patel, N. 2023. LocalStorage vs Cookies: the best-practice guide to storing JWT tokens securely in your front-end. Viitattu 3.6.2024. <https://www.cyberchief.ai/2023/05/secure-jwt-token-storage.html>

Pinnikey. 2016. Pinnikey:n GitHub sivu. Viitattu 20.4.2024. <https://github.com/savikko/pinnikey>

Raval, N. 2023. React vs Vue: Comparison of the Best JavaScript Frameworks at a Glance!. Viitattu 2.6.2024. <https://radixweb.com/blog/react-vs-vue>

Roboto. N.d. Roboto kirjaintyyppi Google Fonts palvelussa. Viitattu 26.5.2024. <https://fonts.google.com/specimen/Roboto/about>

Roppola, J. 2024. Viesti keskustelu käyttöliittymästä. Keskustelu käyty 16.5.2024.

Savikko, T. 2024. Hyppytouhujen välitilinpäätös. Viitattu 20.4.2024. <https://savikko.medium.com/hyppytouhujen-v%C3%A4litolip%C3%A4%C3%A4t%C3%B6s-b0903c9d8e70>

Skydive Finland. 2021. Skydive Finland verkkosivu. Viitattu 17.4.2024. <https://www.skydivefinland.fi/>

SkyWin history - the evolution of SkyWin. 2024. SkyWin sivustolla. Viitattu 15.4.2024. <https://skywin.se/skyWin/history>

Stack Overflow Trends. 2024. Tilasto Stack Overflown verkkosivulla. Viitattu 21.4.2024. <https://insights.stackoverflow.com/trends?tags=vue.js%2Creact.js%2Cangular%2Csvelte%2Cember.js%2Cmeteor%2Cnext.js%2Cbackbone.js%2Cjquery>

Top 10 Most Popular JavaScript Frameworks For Web Development. 2024. Blogikirjoitus GeeksforGeeks sivustolla. Viitattu 21.4.2024. <https://www.geeksforgeeks.org/most-popular-javascript-frameworks-for-web-development/>

Top 10 Most Popular JavaScript Frameworks to Choose in 2024. 2023. Blogikirjoitus Medium sivustolla. Viitattu 21.4.2024. <https://medium.com/@evincedevlop/top-10-most-popular-javascript-frameworks-to-choose-in-2024-269453cdf35>

Watmore, J. 2021. Axios vs Fetch - HTTP GET Request Comparison by Example. Viitattu 13.5.2024. <https://jasonwatmore.com/post/2021/10/04/axios-vs-fetch-http-get-request-comparison-by-example>

Ways of Using Vue. N.d. Vuen dokumentaatio. Viitattu 20.5.2024. <https://vuejs.org/guide/extras/ways-of-using-vue>

What is jQuery?. 2024. jQuery:n verkkosivu. Viitattu 2.6.2024. <https://jquery.com/>

What is MongoDB? N.d. Artikkelit MongoDB:stä IBM:n verkkosivulla. Viitattu 21.5.2024. <https://www.ibm.com/topics/mongodb>