

**SAVONIA**

ammattikorkeakoulu

OPINNÄYTETYÖ - AMMATTIKORKEAKOULUTUTKINTO  
TEKNIIKAN JA LIIKENTEEN ALA

# LOGISTIIKAN HALLINNOINTI WEB- KEHITYSTYÖLLÄ

TEKIJÄ/T Oskar Toivanen

Koulutusala Tekniikan ja liikenteen ala	
Tutkinto-ohjelma Tietotekniikan tutkinto-ohjelma	
Työn tekijä(t) Oskar Toivanen	
Työn nimi Logistiikan hallinnointi webkehitystyöllä	
Päiväys 24.1.2024	Sivumäärä/Liitteet 23
Toimeksiantaja/Yhteistyökumppani(t) Kantoapu Joensuu Oy	
Tiivistelmä <p>Tämän opinnäytetyön tavoitteena oli kehittää sovellus kuljetus- ja muuttoalan yritykselle. Sovelluksen avulla pyrittiin hallinnoimaan yritysasiakkaiden kuljetuksia tehokkaasti ja innovatiivisesti. Projektissa keskityttiin tarjoamaan asiakkaille helppo ja nopea tapa ilmoittaa kuljetuksista, pyrkien eliminoimaan perinteisen sähköposti-viestinnän ongelmia.</p> <p>Työssä esiteltiin ja hyödynnettiin moderneja sovelluskehityksen menetelmiä, joilla tavoiteltiin yrityksen työtaakan helpottamista tilausten käsittelyssä. Opinnäytetyö korosti digitalisaation merkitystä logistiikka-alalla, esitellen uusia tapoja tehostaa yrityksen toimintaa ja parantaa palveluiden laatua.</p> <p>Projektin kautta tuotiin esiin, kuinka teknologian hyödyntäminen voi tarjota ratkaisuja alalla kohdattaviin haasteisiin. Analysoitiin, miten teknologia voi muuttaa yrityksen sisäisiä prosesseja ja parantaa asiakaskokemusta, ja esiteltiin saavutettuja tuloksia. Tuloksissa näkyi, että sovelluksen käyttöönotto paransi merkittävästi sekä kuljetusprosessien tehokkuutta että asiakastyytyvyyttä.</p>	
Avainsanat Sovelluskehitys, Next.js, Tietokanta, PostgreSQL, Käyttöliittymä	

Field of Study Technology, Communication and Transport	
Degree Programme Degree Programme in Information Technology	
Author(s) Oskar Toivanen	
Title of Thesis Improving Logistics Management by Web Application Development	
Date 24 January 2024	Pages/Appendices 23
Client Organisation /Partners Kantoapu Joensuu Oy	
<b>Abstract</b> <p>The objective of this thesis was to develop an application for a transportation and moving company. The aim was to manage the transportation of business clients efficiently and innovatively by using the application. The project focused on providing customers an easy and fast way to report their transport needs, thus seeking to eliminate the problems of traditional email communication.</p> <p>In this project, modern application development methods were introduced and utilized, aiming to ease the company's workload in processing orders. The thesis emphasized the importance of digitalization in the logistics sector by presenting new ways to enhance the company's operations and improve the quality of Services.</p> <p>As a result, the project highlighted how the use of technology can offer solutions to challenges faced in the logistics sector. The analysis made in the thesis clarified how technology can change internal business processes and improve customer experience. The outcomes showed that the implementation of the application significantly improved both the efficiency of transportation processes and customer satisfaction in the client organisation.</p>	
<b>Keywords</b> Application Development, Next.js, Database, PostgreSQL, User Interface	

## SISÄLTÖ

1	JOHDANTO .....	6
2	LOGISTIIKKA- JA MUUTTOPALVELUALAN DIGITALISOITUMINEN .....	7
2.1	Palvelualojen piirteet.....	7
2.2	Ohjelmistoalan tulevaisuus logistiikka- ja muuttopalvelualalla .....	7
2.3	Tuotteistamisen hyötyjä, riskejä ja haasteita.....	7
2.4	Milloin palvelu kannattaa digitalisoida? .....	8
2.5	Käyttökohde ja palvelupiste .....	8
3	KEHITTÄMISPROSESSI .....	9
3.1	Teknologian valinta .....	9
3.2	Tietokannan suunnittelu .....	9
3.3	Arkkitehtuurin määrittely .....	9
3.4	Aikataulutus ja versionhallinta.....	10
4	OHJELMISTON TESTAUS JA VIIMEISTELY .....	11
5	TOIMINNAN KUVAUS.....	12
5.1	Tilausten hallintajärjestelmä .....	12
5.2	Datan analytiikkapaneeli.....	16
5.3	Analytiikkakaavio .....	17
5.4	Dashboard.....	19
6	KEHITYSKOhteet JA JATKOKEHITYS .....	21
7	YHTEENVETO.....	22
	LÄHTEET .....	23

## KUVALUETTELO

KUVA 1 Tilausten hallinnan näkymä.....	12
KUVA 2 Tilausten hallinnan JSX-rakenne.....	13
KUVA 3 Listan toiminnot .....	14
KUVA 4 Navigointielementtien JSX-rakenne .....	15
KUVA 5 Analytiikan yleisnäkymä.....	16
KUVA 6 Uusimmat kuljetukset.....	17
KUVA 7 Analytiikkakaavio .....	17
KUVA 8 Analytiikkakaavion JSX-rakenne .....	18
KUVA 9 Ilmoituslista .....	19
KUVA 10 Ilmoituslistan JSX-rakenne .....	20

## 1 JOHDANTO

Tämä opinnäytetyö keskittyy logistiikka- ja kuljetusalalla kohdattavaan keskeiseen haasteeseen: tehokkaaseen kuljetusten hallintaan ja työn vastaanottamisen nopeuttamiseen. Työn keskiössä on sovelluksen kehittäminen, minkä tarkoituksena on helpottaa yrityksen työtaakkaa kuljetusten ja tilausten vastaanottamisessa hyödyntäen nykyaikaisia sovelluskehityksen menetelmiä. Tavoitteena on luoda kuljetus- ja muuttoalan yritykselle alusta, joka mahdollistaa yritysasiakkaiden kuljetusten hallinnoinnin ja tarjoaa asiakkaille nopean ja vaivattoman tavan ilmoittaa kuljetuksista, välttämällä sähköpostin tuomat haasteet.

Aloittava yritys voi tulla toimeen aluksi yksinkertaisilla menetelmillä, kuten sähköpostilla, puhelimitse tai paperikalenterilla. Kuitenkin nyky-yhteiskunnassa jopa pienikokoiset yritykset, joissa on vain muutamia työntekijöitä, hyötyvät merkittävästi palvelin pohjaisesta tilaustenhallintajärjestelmästä. Tulevaisuuden trendit osoittavat, että yhä useammat yritykset siirtyvät palvelin pohjaiseen tilaustenhallintaan. Tämä siirtyminen tarjoaa useita etuja, kuten parannettua kilpailukykyä ja tehokkaampaa hallinnointia, joita tämä opinnäytetyö tarkastelee yksityiskohtaisemmin. Työn tarkoituksena on tarjota konkreettisia ratkaisuja, jotka vastaavat kuljetus- ja logistiikka-alan kasvaviin vaatimuksiin ja auttavat yrityksiä kehittämään toimintaansa uudella, innovatiivisella tavalla.

Toimeksiannon antajana tässä opinnäytetyössä toimii Kantoapu Joensuu Oy, joka perustettiin vuonna 2018 kahden ammattikorkeakoulun opiskelijan toimesta. Yritys tarjoaa muuttopalveluita ja on laajentanut toimintaansa perustamisen jälkeen. Alkuunsa Kantoapu Joensuu Oy tarjosi palvelujaan vain Joensuun alueella, mutta vuonna 2021 se laajensi toimintaansa Kuopioon, perustaen Kantoapu Kuopio Oy:n. Yrityksen perustajilla on yli seitsemän vuoden kokemus muuttotyöstä, ensin isompien yritysten palkkalistoilla ja sittemmin yrittäjinä. Tämä kokemus on auttanut heitä ymmärtämään kuljetus- ja muuttoalan haasteita ja mahdollisuuksia, mikä puolestaan on ohjannut heidän yrityksensä kehitystä. Kantoapu Joensuu Oy:n kasvun myötä on syntynyt tarve tehokkaammalle tilausten ja kuljetusten hallintajärjestelmälle, mikä innoitti tämän opinnäytetyön kehittämistä. Yrityksen tavoitteena on jatkaa toimintansa laajentamista ja parantaa palveluidensa laatua ja tehokkuutta nykyaikaisten teknologisten ratkaisujen avulla.

## 2 LOGISTIIKKA- JA MUUTTOPALVELUALAN DIGITALISOITUMINEN

### 2.1 Palvelualojen piirteet

Sana logistiikka on melko uusi käsite, sillä logistiikanmaailman mukaan kyseinen ilmestyi 1970-luvulla ja tuli usein käytetyksi 1980-luvulla. Nykyisin sana on arkipäiväinen monella, mutta sen sisältö ja tarkoitus ei ole monelle tuttu aihe. Ennen sanaa käytettiin metsästyksen ja keräilyn yhteydessä, jossa oli tarvetta ratkoa logistisia ongelmia. Nykyisin logistiikka tarkoittaa suppeassa käsitteessä tavaroiden kuljetusta ja niiden varastointia sekä laaja logistiikka on erilaisten materiaalien, valuuttojen ja tietokokonaisuuksien hallitsemista. (Logistiikan maailma, 2023)

Logistiikan palvelualaa mietittäessä tulee huomatuksi nopea digitalisoituminen kuten Fintrafficin sivuilla todetaan. Vuoteen 2025 pyritään siirtämään rahtikirjat sähköisiksi eFTI-asetuksella, tämän avulla sähköiset kuljetustiedot olisivat aina saatavilla ja käännettävissä missä tahansa EU-alueen maassa kuljetettava rahti on. Tämä pienentäisi hiilijalanjälkeä, vähentäisi väärinymmärryksiä ja antaisi alustan ponnistaa digitaalisiin rajapintoihin logistiikka-alalla. (Fintraffic, 2023)

### 2.2 Ohjelmistoalan tulevaisuus logistiikka- ja muuttopalvelualalla

Yritysten ja organisaatioiden toiminnassa sovelletaan älykstä logistiikkaa usealla eri tavalla. Näitä eri tapoja ovat mm. mallinnus- ja simulaatio-ohjelmistot, joiden avulla pystytään havainnollistamaan ja vertailemaan vaihtoehtoja, ohjelmistoja, joilla pystytään arvioimaan karkeasti tulevan kuljetustarpeen sekä päivittäin käytettäviä ohjelmistoja kuten toiminnanohjausjärjestelmiä ja sekä tunnistus- ja paikkatietotekniikkaa. Digitalisaatio on luonut uusia mahdollisuuksia logistiikka-alalle ja yksi näistä mahdollisuuksista on toimitusketjun läpinäkyvyys. Toimitusketjun läpinäkyvyys vaatii toimiakseen verkkosivustopohjaisia sovelluksia ja sanomanvälitysratkaisuja. (Logistiikan maailma, 2023)

Digitalisaatio kehitty vauhdilla ja yksi sen keskeisimmistä ilmiöistä on tekoäly. Tämä kehitys näkyy selvästi monilla alueilla, kuten visuaalisessa taiteessa, chattibotteina ja niiden kasvavassa läsnäolossa mediassa. Tekoälyn etunenässä kulkevat innovaattorit luovat uusia mahdollisuuksia ja sovelluksia, jotka ovat saavutettavissa myös niille organisaatioille, jotka vasta tutustuvat tekoälyyn.

Terminä tekoäly on yleisesti käytetty, ja Rusasen (2023) mukaan jopa automaatiototeutuksia saatetaan kutsua tekoälyksi. Tulevaisuudessa tekoäly tulee ottamaan merkittävämpää roolia useilla eri aloilla, mutta varsinkin aloilla, jotka vaativat datan pohjalta ennustamista. Kuljetusala on yksi näistä aloista, mitkä vaativat datan pohjalta ennustamista. Tekoälyn hyödyntäminen edellyttää muutosta ja organisaatioiden tulisi arvioida liiketoimintaansa ja miettiä, miten tekoäly voisi tehostaa niiden toimintaa. (Rusanen, 2023)

### 2.3 Tuotteistamisen hyötyjä, riskejä ja haasteita

Tuotteistamisen prosessissa yksi keskeinen haaste on palvelun tai tuotteen kuvaamisen monimutkaisuus, etenkin kun kyseessä ovat asiantuntijapalvelut, jotka ovat luonteeltaan monimuotoisia. Tärkeää on määritellä selkeät rajat palvelulle ja sen elementeille sekä ymmärtää niiden suhde yrityksen muihin palveluihin. Tässä vaiheessa on otettava huomioon eri sidosryhmien, kuten asiakkaiden, pal-

veluntuottajien ja markkinoinnin, tarpeet, sillä kukin tarvitsee palvelusta eri tarkkuustason kuvauksen. Yleisen mallin luominen, joka toimii yhteisenä suuntaviittana kaikille osapuolille, on tässä yhteydessä avainasemassa. (Tuominen;Järvi;Lehtonen;Valtanen ;& Martinsuo, 2015)

Tuotteistamisessa hyödynnetään monenlaisia kuvaustapoja, kuten visuaalisia malleja ja prototyypppejä, jotka auttavat esittämään palvelun eri osapuolille merkityksellisessä muodossa. Eri kuvaustavat palvelevat eri vaiheita tuotteistamisprosessissa, ja valintaan vaikuttavat sekä tuotteistamisen tavoitteet että osallistujien roolit ja tarpeet. Kuvaustavan valinnassa on tärkeää ottaa huomioon, että lopullisen tiedon on oltava ymmärrettävä ja mielekäs kohderyhmälle. Tuotteistamisen onnistuminen vaatii usein myös työntekijöiden ja asiakkaiden aktiivista osallistumista sekä yhteistyötä, jotta voidaan luoda ja jalostaa palvelumallia. (Tuominen;Järvi;Lehtonen;Valtanen ;& Martinsuo, 2015)

#### 2.4 Milloin palvelu kannattaa digitalisoida?

Digitalisoinnin kannattavuus riippuu monista tekijöistä, kuten palvelun luonteesta ja asiakkaiden odotuksista. Digitalisaatio tarjoaa mahdollisuuden parantaa palvelun tehokkuutta ja saatavuutta, mutta se edellyttää, että palveluntarjoajilla on selvä visio siitä, miten digitaalinen alusta tukee ja parantaa asiakaskokemusta. Tämän lisäksi digitalisaation tulisi tapahtua käyttäjien tarpeiden ja toimintaympäristön ymmärtämisen pohjalta, jotta digitaalinen palvelu todella tuo lisäarvoa sekä asiakkaille että yritykselle. (Tuominen;Järvi;Lehtonen;Valtanen ;& Martinsuo, 2015)

#### 2.5 Käyttökohde ja palvelupiste

Käyttökohde ja palvelupiste ovat olennaisia elementtejä palvelun tuotteistamisessa. Kun palvelua kehitetään, on tärkeää tietää, missä ja miten palvelua käytetään. Tämän ymmärtäminen auttaa määrittelemään, millaisessa ympäristössä palvelu toimii ja miten asiakas kohtaa palvelun. Palvelupisteiden suunnittelussa on keskityttävä siihen, miten palvelu on tavoitettavissa ja miten se integroituu asiakkaan arkeen ja toimintaprosesseihin. Tämän prosessin aikana on tärkeää arvioida palvelun käytettävyyttä ja saavutettavuutta, jotta varmistetaan sujuva asiakaskokemus koko palvelupolulla. (Tuominen;Järvi;Lehtonen;Valtanen ;& Martinsuo, 2015)



### 3 KEHITTÄMISPROSESSI

#### 3.1 Teknologian valinta

Haikala ym. (2011) toteaa ohjelmistojen arkkitehtuurin yleisestä väärin ymmärryksestä - yleisesti se tarkoittaa ohjelmistoja ja niiden komponenttien välisiä suhteita. Haikalan (2011) mukaan lähes poikkeuksetta mukaan ajatellaan jatkokehitys toimenpiteet ja suuntaviivat sille. Arkkitehtuurin tehtävä on pystyä luomaan kuva ja määrittelemään työnjako ohjelmiston komponenttien välille. Lopulta kyse on monimutkaisen käsitteiden ja käytäntöjen hallinnasta arkkitehtuurin määrittelemän käsitteistön mukaan. (Haikala & Mikkonen, 2011, ss. 178-180)

Opinnäytetyön tekemiseen valittiin sovelluskehikseksi React.js:n pohjalta rakennettu Next.js, joka on yksityisen Vercel yrityksen luoma verkkokehityskehys, joka tarjoaa verkkosovelluksia mitkä ovat kasattu palvelimen puolella jo valmiiksi loppu käyttäjälle. Hyödyt tästä valinnasta opinnäytetyötä varten tulevat Next.js verkkokehityksen toiminnoista, jotka tukevat nykyaikaisempaa rajapintaa, sovelluksen ulkonäköä, optimointia ja virheen hallintaa. (Vercel, 2023)

Nykyaikana teknologia on kehittynyt käyttämään useita eri kirjastoja kokonaisen sovelluksen luontiin. Harvoin enää luodaan kokonaan uusia ohjelmisto- tai komponenttikirjastoja, ellei tarvita jotain hyvin yksityiskohtaista. Opinnäytetyö otti myös tästä ajattelu tavasta mallia ja päätyi käyttämään valmiiksi kehitettyjä ratkaisuja kirjastojen avulla. Tästä näkyvin ratkaisu käyttäjän päässä on Radix ui -komponenttikirjasto, joka on muokattu tilaajan haluaman näköiseksi käyttämällä css-kehystä nimeltä Tailwind css. Näiden avulla saadaan nykyaikainen verkkosivuston ulkonäkö luotua, mutta siihen ei tarvitse käyttää projektin ajasta suurinta osaa, vaan säästetyn ajan pystyy kehittäjä käyttämään toimintojen ja rajapinnan suunnitteluun. (Radix, 2023)

#### 3.2 Tietokannan suunnittelu

Opinnäytetyössä tietokannan suunnitteluvaihe oli keskeinen. Tässä prosessissa opinnäytetyön tekijä yhdessä tilaajan kanssa päätyi valitsemaan relaatiotietokannan. Tämä valinta tehtiin ottaen huomioon tulevaisuuden tarpeet ja relaatiotietokantojen monipuoliset hyödyt. Relaatiotietokannat ovat tunnettuja niiden joustavuudesta ja kyvystä käsitellä monimutkaisia tietorakenteita, mikä tekee niistä ihanteellisia monenlaisiin sovelluksiin. Lisäksi ne tarjoavat vankan perustan datan järjestelmälliselle hallinnalle ja kyselyille, mikä on erityisen tärkeää opinnäytetyön kaltaisissa projekteissa, joissa tiedon tarkkuus ja saatavuus ovat avainasemassa. Tämän päätöksen myötä tietokannan suunnittelussa keskityttiin luomaan sellainen rakenne, joka tukee opinnäytetyön tavoitteita ja vastaa tilaajan tarpeisiin sekä tämänhetkisiin että tulevaisuuden käyttötarpeisiin. (Microsoft, 2023)

#### 3.3 Arkkitehtuurin määrittely

Tarkastellaan valitun ohjelmiston arkkitehtuuriin liittyviä periaatteita ja käytäntöjä. Tässä yhteydessä otetaan huomioon erilaiset arkkitehtuuriset tyylit ja suunnittelumallit, jotka mahdollistavat joustavan ja laadukkaan ohjelmistokehityksen. Esimerkiksi käytetty Next.js-ympäristö toimii pohjana käyttöliittymän rakentamiselle, PostgreSQL tietokanta tarjoaa luotettavan ja skaalautuvan datan hallinnan, ja Azure DevOps mahdollistaa projektin aikataulutuksen ja hallinnan.

Arkkitehtuuria määriteltäessä korostetaan eri komponenttien ja palveluiden välistä vuorovaikutusta, jonka tavoitteena on varmistaa sujuva tiedonkulku ja optimaalinen suorituskyky. Suunnitteluohjelma Figmaa käytetään tässä yhteydessä suunnittelutyökaluna, joka auttaa visuaalisen käyttöliittymän ja käyttäjäkokemuksen suunnittelussa. Ohjelmistoprojektissa hyödynnetään myös eri arkkitehtuurimalleja, kuten monikerrosarkkitehtuuri, palveluorientoituneita arkkitehtuureja ja repositorioarkkitehtuureja, jotka edesauttavat modulaarista ja hyvin määriteltyä ohjelmistosuunnittelua.

Tärkeää on myös tunnistaa, että hyvä ohjelmistoarkkitehtuuri ei ainoastaan tue nykyisiä vaatimuksia, vaan se on suunniteltu siten, että se voi mukautua tulevaisuuden muutoksiin ja laajentumiseen. (Haikala & Mikkonen, 2011, s. 180). Opinnäytetyössä käsitellään, miten valitut arkkitehtuuriset päätökset vaikuttavat sovelluksen laajennettavuuteen, ylläpidettävyyteen ja käytettävyyteen pitkällä tähtäimellä. Tämä kattava arkkitehtuurin määrittely toimii perustana ohjelmiston kestäväälle kehitykselle ja on olennainen osa opinnäytetyön teknistä dokumentointia.

### 3.4 Aikataulutus ja versionhallinta

Projektinhallinnan ja versionhallinnan käyttö ovat nykyaikaisen ohjelmistokehityksen kannalta keskeistä. Azure Boards tulee ketterää kehitystä, kuten Scrum-menetelmää, tarjoten ominaisuuksia, kuten sprintit ja tilauskannat, jotka auttavat aikataulutuksessa ja tehtävien hallinnassa. (Microsoft, 2023).

GitHub on suosittu versiohallintatyökalu, joka helpottaa koodin jakamista, tarkastelua ja yhteistyötä kehittäjien kesken. Tämä on erityisen hyödyllistä projekteissa, joihin odotetaan uusia ohjelmoijia tai joita kehitetään pitkään. GitHub sisältää ominaisuuksia kuten pull requestit, issue trackerit ja wikisivut, jotka auttavat dokumentoinnissa ja ongelmanratkaisussa kehityksen aikana. Näiden toimintojen yhdistäminen Azure Boardsin projektinhallintaan tehostaa työnkulkua, sillä tehtävien hallinta ja koodin kehitys integroituvat saumattomasti. (Github, 2023)

#### 4 OHJELMISTON TESTAUS JA VIIMEISTELY

Next.js:n käyttö projektin kehityksessä tarjoaa monia hyötyjä, erityisesti suorituskyvyn ja julkaisuprosessin osalta. Tämä web-kehityksen alusta (framework) mahdollistaa suorituskykyisen sovelluksen rakentamisen, mikä on tärkeää käyttäjätestauksen ja demovaiheen aikana.

Projektin viimeistelyssä käyttäjätestaukseen, Next.js:n optimointiominaisuudet, kuten automaattinen koodin jakaminen, kuvien optimointi ja lazy loading, parantavat latausnopeuksia ja vähentävän resurssien kulutusta. Nämä toiminnot ovat avainasemassa sovelluksen tehokkaassa toiminnassa. (Vercel, 2023)

Julkaisuvaiheessa Next.js helpottaa sovelluksen julkaisua tarjoamalla esikatselutuen ja automatisoidun resurssien optimoinnin. Next.js tukee lisäksi erilaisia hosting-ratkaisuja ja ympäristöasetusten määrittelyä, mikä tekee siitä joustavan valinnan eri julkaisutarpeisiin.

Next.js:n käyttäminen projektin kehityksessä varmistaa, että sovellus on valmis niin sisäisiin demotestauksiin kuin julkiseen käyttöönottoonkin. Sen tarjoamat optimointi- ja julkaisuominaisuudet takaavat, että sovellus on paitsi suorituskykyinen, myös helppo julkaista ja ylläpitää pitkällä aikavälillä. (Vercel, 2023)

## 5 TOIMINNAN KUVAUS

### 5.1 Tilausten hallintajärjestelmä

Kuvassa 1 on tilausten hallintajärjestelmän näkymä, joka näyttää listan tilauksista. Listalla on seuraavat sarakkeet: ID, Myyjä, Viimeinen toimituspäivä, Arvioitu toimituspäivä, Muokattu, Asiakas #, ja Tila. Järjestelmä on suunniteltu auttamaan myyjiä ja hallinnoijaa tilausten käsittelyssä. Se näyttää jokaisen tilauksen tunnuksen (ID) ja tilausta käsitelleen myyjän nimen. Yritys asettaa asiakkaille viimeisen toimituspäivän, ja hallinnoija eli toimitusta hallinnoiva yritys arvioi toimituspäivän, jonka he itse päivittävät järjestelmään. Tilausvaihe vaihtelee "Uusi" -tilasta "Käsittelyssä" -tilaan, kun arvioitu toimituspäivä on asetettu. "Toimitettu" -tila vahvistetaan, kun toimitusta hallinnoiva yritys merkitsee tilauksen toimitetuksi.

#### Tervetuloa Oskar

demo

ID	Myyjä	Viimeinen toimitus pv	Arvioitu toimitus pv	Muokattu	Asiakas #	Tila
7	Oskar	19.10.2023	17.6.2024	17.12.2023	746136	Toimitettu
8	John Doe	10.12.2023	17.8.2024	17.12.2023	158826	Toimitettu
9	Matti Meikäläinen	8.5.2023	4.5.2024	17.12.2023	046678	Toimitettu
10	Oskar	15.4.2023	31.12.2023	17.12.2023	515999	Toimitettu
11	John Doe	18.5.2023	28.12.2023	17.12.2023	058652	Käsittelyssä
12	Matti Meikäläinen	24.9.2023	17.12.2023	17.12.2023	974530	Käsittelyssä
13	Oskar	27.10.2023	30.6.2024	17.12.2023	685783	Uusi
14	John Doe	14.12.2023	14.8.2024	17.12.2023	468181	Toimitettu
15	Matti Meikäläinen	20.12.2022	11.8.2024	17.12.2023	428266	Toimitettu
16	Oskar	14.7.2023	18.5.2024	17.12.2023	451107	Uusi

0 of 50 row(s) selected. Rows per page 10 Page 1 of 5

KUVA 1 Tilausten hallinnan näkymä

Kuvassa 2 näkyvä koodi käynnistää renderöintiprosessin palauttamalla JSX-rakenteen, joka muodostaa käyttöliittymän näkymän. Ulkoinen html elementti määrittelee säiliön, joka on keskitetty ja varustettu sisäisellä tyhjällä tilalla, mahdollistaen mukautuvan ja esteettisesti miellyttävän layoutin. Käyttämällä responsiivisia Tailwind CSS -luokkia, kuten flex, flex-col, ja md:flex, komponentti pystyy näiden avulla mukautumaan eri näyttölaitteiden kokoihin ja suuntiin.

Sisäinen html elementti määrittelee alueen, jossa käyttäjän tervehdys ja organisaation tilan indikaattori sijaitsevat. Tervetuloviestissä käytetään dynaamista JavaScript-syntaksia käyttäjän etunimen esittämiseen, olettaen että käyttäjäobjekti on saatavilla ja siinä on määritelty etunimiominaisuus. Organisaation nimen näyttäminen on ehdollista: jos tiedot ovat ladattu ja organisaationobjekti on määritelty, näytetään sen nimiominaisuus; muussa tapauksessa käyttäjälle näytetään 'Loading...' -teksti, joka indikoi tietojen lataamista.

<UserNav /> on komponentti, joka sisältää käyttöliittymän navigointielementit, kuten toiminnot tietojen etsimiseen, kuljetusten luomiseen, tilan suodattamiseen ja näkymän muokkaukseen.

Lopuksi <DataTable />-komponentti, joka saa data- ja columns-attribuutteina deliveryRequests- ja columns-muuttujat, luo visuaalisen esityksen kuljetuspyyntöjen tiedoista. Tämä taulukko ilmestyy, kun kaikki tarkistukset ovat valmiit ja positiivisia, ja se mahdollistaa käyttäjälle organisaation kuljetustietojen tarkastelun ja hallinnan.

```

return (
  <div className='container mx-auto p-4'>
    <div className=' h-full flex-1 flex-col space-y-8 p-8 md:flex'>
      <div className='flex items-center justify-between space-y-2'>
        <div>
          <h2 className='text-2xl font-bold tracking-tight'>
            Tervetuloa {user?.firstName}
          </h2>
          <p className='text-muted-foreground'>
            {isLoading && organization ? organization.name : 'Loading...'}
          </p>
        </div>
        <div className='flex items-center space-x-2'>
          <UserNav />
        </div>
      </div>
      <DataTable data={deliveryRequests} columns={columns} />
    </div>
  </div>
)
}

```

KUVA 2 Tilausten hallinnan JSX-rakenne

Kuvassa 3 on tilausten hallintajärjestelmän näkymä, jossa on käytössä suodattimet. Käyttäjä voi käyttää suodattimia etsiäkseen tilauksia asiakasnumeron perusteella, luomaan uusia kuljetuksia tai selaamaan kuljetusten eri tiloja. Suodatinmahdollisuus tilasarakkeessa antaa käyttäjälle mahdollisuuden tarkastella kaikkia tiloja tai rajata tiettyyn tilaan. Käyttäjä voi myös järjestää listan sarakkeet suuruusjärjestyksessä ja muokata tai poistaa vain itse luomiaan kuljetuksia. Lisäksi pylväitä voi piilottaa, mikäli käyttäjä haluaa yksinkertaistaa näkymää.

## Tervetuloa Oskar

demo

ID	Myyjä	Viimeinen t...	Tila	Käsittelyssä	Reset	Muokattu	Asiakas #	Tila
11	John Doe	18.5.2023	<input checked="" type="checkbox"/>	10	...	17.12.2023	058652	Käsittelyssä
12	Matti Meikäläinen	24.9.2023	<input type="checkbox"/>	25	...	17.12.2023	974530	Käsittelyssä
18	Matti Meikäläinen	25.9.2023	<input type="checkbox"/>	15	...	17.12.2023	828869	Käsittelyssä
20	John Doe	10.9.2023	<input type="checkbox"/>	10	...	17.12.2023	375170	Käsittelyssä
23	John Doe	28.3.2023	<input type="checkbox"/>	10	...	17.12.2023	800688	Käsittelyssä
28	Oskar	10.11.2023	<input type="checkbox"/>	10	...	17.12.2023	691506	Käsittelyssä
34	Oskar	26.10.2023	<input type="checkbox"/>	10	...	17.12.2023	935771	Käsittelyssä
40	Oskar	2.12.2023	<input type="checkbox"/>	10	...	17.12.2023	213478	Käsittelyssä
49	Oskar	17.9.2023	<input type="checkbox"/>	10	...	17.12.2023	825728	Käsittelyssä
52	Oskar	14.6.2023	<input type="checkbox"/>	10	...	17.12.2023	890918	Käsittelyssä

0 of 10 row(s) selected. Rows per page: 10 Page 1 of 1

### KUVA 3 Listan toiminnot

Kuvassa 4 näkyy listan navigointi komponentti, jossa keskitytään ensisijaisesti käyttäjän interaktiivisiin toimintoihin, kuten tietojen hakuun ja kuljetusten hallintaan. Luodussa rakenteessa on tärkeässä roolissa hakukenttä, jonka avulla käyttäjät voivat suodattaa tietoja dynaamisesti. Hakukentän arvo ja sen päivityslogiikka on sidottu tiiviisti taustalla olevaan datarakenteeseen, tarjoten reaaliaikaisen ja intuitiivisen käyttäjäkokemuksen.

Lisäksi kuljetusten luomiseen tarkoitettu painike yhdistää toiminnallisuuden ja visuaalisen ilmeen, käyttäen ikonografiaa viestimään käyttäjälle painikkeen tarkoituksesta. Tämän lähestymistavan ansiosta käyttöliittymästä tulee sekä toiminnallinen että esteettisesti miellyttävä. Kuljetusten tilan hallintaan liittyvä suodatuskomponentti lisää käyttöliittymään lisäkerroksen, mahdollistaen tarkemman tietojen rajaamisen ja hallinnan, mikä on erityisen tärkeää suurten tietomäärien käsittelyssä.

Mielenkiintoinen yksityiskohta on myös suodatusten resetointiin tarkoitettu painike, joka ilmestyy vain, kun suodattimia on aktiivisesti käytössä. Tämä elementti tuo käyttöliittymään dynaamisuutta ja

käyttäjälähtöisyyttä, tarjoten selkeän mekanismin suodatusten nopeaan ja tehokkaaseen nollaamiseen.

Lopuksi, <DataTableViewOptions>-komponentti, vaikka se onkin visuaalisesti vähemmän ilmeinen, on tärkeä osa kokonaisuutta, tarjoten lisäasetuksia ja hallintamekanismeja datan esittämiseen.

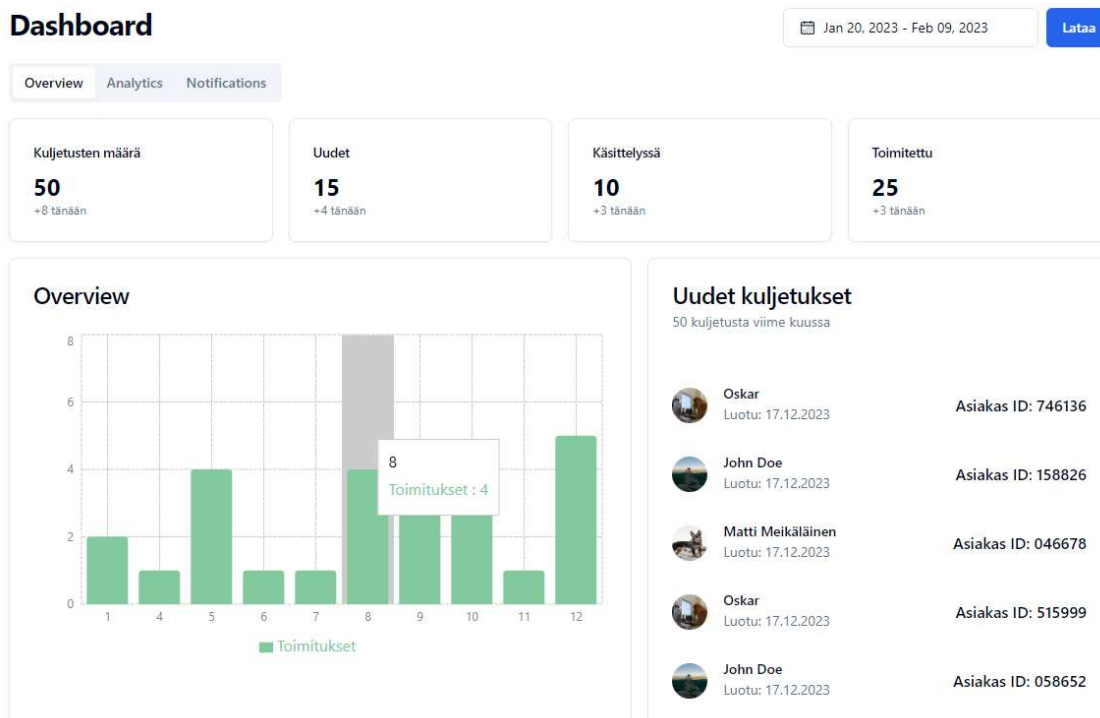
Tämä elementti täydentää käyttöliittymän tarjoamalla käyttäjille lisäkontrollia siitä, miten ja millaisessa muodossa data esitetään, mikä on keskeistä erityisesti monimutkaisissa tai tietointensiivisissä sovelluksissa.

```
return (
  <div className='flex items-center justify-between'>
    <div className='flex flex-1 items-center space-x-2'>
      <Input
        placeholder='Etsi...'
        value={
          (table.getColumn('clientNumber')?.getFilterValue() as string) ?? ''
        }
        onChange={(event) =>
          table.getColumn('clientNumber')?.setFilterValue(event.target.value)
        }
        className='h-8 w-[150px] lg:w-[250px]'
      />
      <Button variant='outline' className='h-8 px-2 lg:px-3'>
        Luo kuljetus
        <PlusCircledIcon className='ml-2 h-4 w-4' />
      </Button>
      {table.getColumn('state') && (
        <DataTableFacetedFilter
          column={table.getColumn('state')}
          title='Tila'
          options={state}
        />
      )}
      {isFiltered && (
        <Button
          variant='ghost'
          onClick={() => table.resetColumnFilters()}
          className='h-8 px-2 lg:px-3'
        >
          Reset
          <Cross2Icon className='ml-2 h-4 w-4' />
        </Button>
      )}
    </div>
    <DataTableViewOptions table={table} />
  </div>
)
```

KUVA 4 Navigointielementtien JSX-rakenne

## 5.2 Datan analytiikkapaneeli

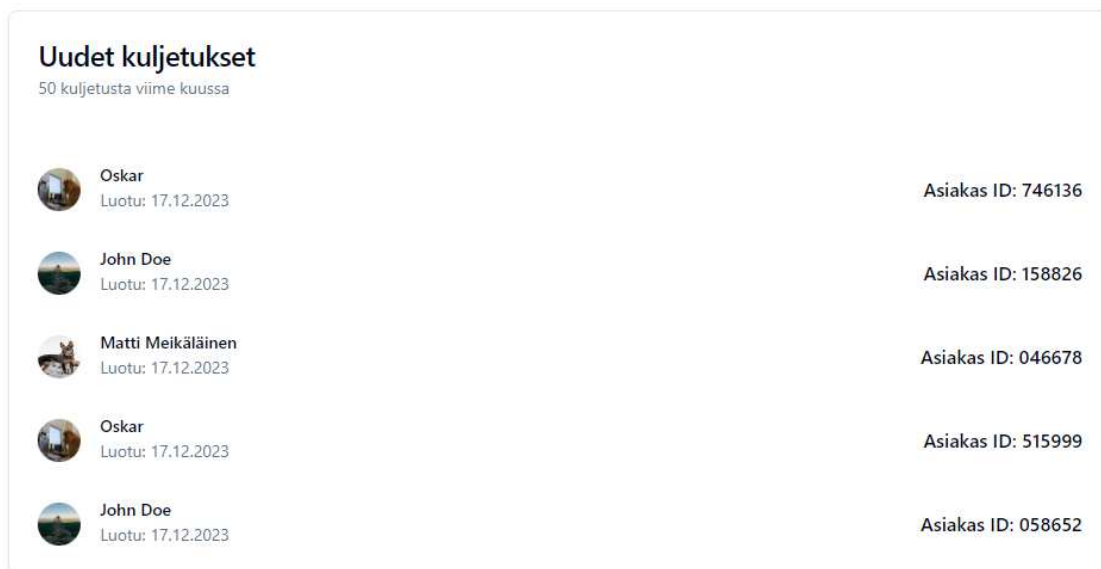
Kuvassa 5 on yrityksen datan analytiikkapaneeli, joka tarjoaa käyttäjille yleiskatsauksen kuljetusten tilastoihin. Käyttäjät voivat nähdä eri kuljetusten määrän, uudet kuljetukset, käsittelyssä olevat ja toimitetut kuljetukset sekä niiden päivittäisen muutoksen. Pylväskaavio esittää kuukausittain toimitettujen kuljetusten määrän, ja kuvassa 4 on myös osio uusimmille kuljetuksille, joka näyttää viimeisimmät lisätyt kuljetukset asiakas-ID:n kera. Lisäksi käyttäjillä on mahdollisuus ladata raportti kuljetuksista valitsemaltaan aikaväliltä.



KUVA 5 Analytiikan yleisnäkymä



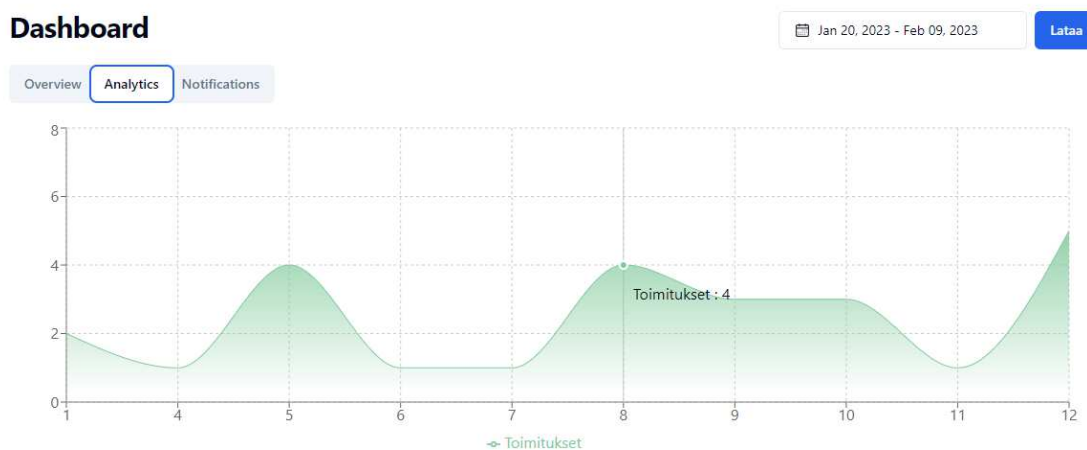
Alla oleva kuva, eli kuva 6, tarjoaa yksityiskohtaisemman näkymän uusimpien kuljetusten tiedoista data-analytiikkapaneelista.



KUVA 6 Uusimmat kuljetukset

### 5.3 Analytiikkakaavio

Tämä analytiikkakaavio (kuva 7) kuvaa kuljetusten määrän vaihtelua kuukauden aikana. Kaaviossa näkyy, kuinka toimitusten määrä on vaihdellut päivittäin, esimerkiksi saavuttaen huippunsa keskivaiheilla ja lopussa. Tämä visuaalinen esitys auttaa hahmottamaan toimitusmäärien trendejä ja huippukohtia, tarjoten arvokasta tietoa toimitusprosessien ja resurssien suunnittelun tueksi. Käyttäjät voivat ladata yksityiskohtaisen raportin kaaviosta valitsemallaan aikavälillä.



KUVA 7 Analytiikkakaavio

Kuva 8 kuvaa komponenttia, jossa luodaan interaktiivinen aluekaavio, joka visualisoi kuukausittaisten toimitusten tietoja.

```

return (
  <ResponsiveContainer width='100%' height={350}>
    <AreaChart
      data={monthlyDeliveries}
      margin={{ top: 10, right: 30, left: 0, bottom: 0 }}
    >
      <defs>
        <linearGradient id='colorTotalDeliveries' x1='0' y1='0' x2='0' y2='1'>
          <stop offset='5%' stopColor='■ #82ca9d' stopOpacity={0.8} />
          <stop offset='95%' stopColor='■ #82ca9d' stopOpacity={0} />
        </linearGradient>
      </defs>
      <XAxis dataKey='month' />
      <YAxis />
      <CartesianGrid strokeDasharray='3 3' />
      <Tooltip content={<CustomTooltip />} />
      <Legend />
      <Area
        type='monotone'
        name='Toimitukset'
        dataKey='total_deliveries'
        stroke='■ #82ca9d'
        fillOpacity={1}
        fill='url(#colorTotalDeliveries)'
      />
    </AreaChart>
  </ResponsiveContainer>
)

```

KUVA 8 Analytiikkakaavion JSX-rakenne

Ensimmäisenä huomion kohteena on `<ResponsiveContainer>`-komponentti, joka takaa, että aluekaavio mukautuu dynaamisesti eri näyttökokoihin. Tämä on elintärkeää nykyisessä monipäätelaiteympäristössä, missä käyttäjät voivat käyttää sovellusta eri laitteilta. Vastuullisen suunnittelun ansiosta kaavio säilyttää toimivuutensa ja visuaalisen selkeytensä riippumatta laitteen näytön koosta.

`<AreaChart>`-elementti on koodin ydin, jossa itse kaavion muoto ja sisältö määritellään. Data, joka syötetään kaavioon (`monthlyDeliveries`), visualisoidaan alueena, joka antaa visuaalisen kuvan toimitusten volyymin muutoksista ajan kuluessa. `Margin`-attribuutin käyttö osoittaa huolellisuutta visuaalisen tilan käytössä, parantaen kaavion luettavuutta ja esteettistä viehätysvoimaa.

Kaavion värittäminen ja ulkoasu tuodaan esiin `<defs>`-elementin ja sisällä määritellyn lineaarisen gradientin avulla. Tämä gradientti, joka määritellään väreillä ja niiden opasiteetilla, luo visuaalisen syvyyden ja erottuvuuden kaavioon, tehostaen käyttäjän kykyä tulkita tietoja visuaalisesti.

X- ja Y-akselien (`<XAxis>` ja `<YAxis>`) määrittely on olennaista datan asianmukaiselle esittämiselle, sillä ne tarjoavat tärkeän viitekehyksen, joka auttaa käyttäjiä ymmärtämään kaavion tietoja

selkeämmin. <CartesianGrid> lisää kaavioon ruudukon, mikä helpottaa tarkkaa lukemista ja tekee kaavion ulkonäöstä ammattimaisemman.

Tooltip-elementin käyttö on erinomainen esimerkki käyttäjäystävällisyydestä. <Tooltip>-komponentti, joka sisältää <CustomTooltip>-elementin, tarjoaa lisätietoja, kun käyttäjä vie hiiren kaavion päälle. Tämä interaktiivinen ominaisuus parantaa käyttökokemusta tarjoamalla kontekstisidonnaista tietoa datasta.

Lopuksi, <Legend>-elementti ja itse <Area>-elementti, joka määrittelee aluekaavion ulkoasun ja käyttäytymisen, viimeistelevät kaavion. Aluekaavio esittää toimitusten määrän visuaalisesti houkuttelevalla ja helposti ymmärrettävällä tavalla, korostaen datan tärkeitä trendejä ja malleja.

## 5.4 Dashboard

Kuva 9 esittää Dashboardin ilmoitusosion, joka näyttää ajantasaisen listan uusista kuljetuspyynnöistä. Jokainen ilmoitus sisältää tunnusteen "Uusi kuljetus!" ja tarjoaa yksityiskohtaiset tiedot, kuten tilaus ID:n, lisäyspäivämäärän ja toimituspäivän tilan, joka on tässä tapauksessa merkitty "Ei sovittu". Tämä on osa käyttäjäystävällistä käyttöliittymää, joka auttaa logistiikan hallinnoinnissa ja mahdollistaa nopean reagoinnin uusiin tilauksiin.

Order ID	Date Added	Delivery Status	Customer ID
661	17.12.2023	Ei sovittu	746136
51	17.12.2023	Ei sovittu	158826
912	17.12.2023	Ei sovittu	046678
207	17.12.2023	Ei sovittu	515999
765	17.12.2023	Ei sovittu	058652
740	17.12.2023	Ei sovittu	974530

KUVA 9 Ilmoituslista

Kuvassa 10 komponentin koodi alkaa <div>-elementillä, jossa on määritelty CSS-luokka 'space-y-8', joka luo tasaisen pystysuoran välin jokaisen listan elementin välille. Tämä suunnitteluratkaisu parantaa elementtien erottuvuutta ja lisää käyttöliittymän visuaalista selkeyttä.

Tilautustietoja sisältävä tilaustaulukko käsitellään pilkkomismetodeilla, jotka rajaavat näytettävien tilausten määrän kymmeneen ja iteroivat niiden yli. Jokaiselle tilaukselle luodaan <Alert>-komponentti, joka toimii visuaalisena indikaattorina tärkeille tilaustiedoille. Jokainen hälytys sisältää oman yksilöllisen avaimen, mikä on tärkeää tehokkaan renderöinnin ja komponenttien hallinnan kannalta Reactissa.

Hälytyskomponentin sisällä käytetään <Avatar>-elementtiä, joka tuo esille visuaalisen käyttäjäprofiilin kuvan. Tämän elementin avulla käyttäjät voivat nopeasti tunnistaa tilaukseen liittyvän henkilön tai yksikön. Avatar-elementin sisäinen rakenne, joka sisältää sekä <AvatarImage> että <AvatarFallback>-komponentit, varmistaa, että käyttäjäprofiilin kuva näytetään johdonmukaisesti ja varavaihtoehto on tarjolla, mikäli pääkuva ei ole saatavilla.

Seuraavaksi, erillinen <div>-elementti sisältää tilauksen tärkeät tiedot, kuten myyjän nimen, tilausnumeron ja luontipäivämäärän. Näitä tietoja korostetaan käyttämällä <AlertTitle>- ja <AlertDescription>-komponentteja, jotka parantavat tekstin luettavuutta ja hierarkiaa. Tässä esitellään tärkeää tietoa, kuten tilausnumero ja arvioitu toimituspäivämäärä, mikä on keskeistä liiketoiminnan prosesseissa.

```
return (
  <div className='space-y-8'>
    {orders.slice(0, 10).map((order, index) => (
      <Alert
        key={index}
        className='flex items-center p-4 border border-gray-200 rounded-lg max-w-7xl mx-left'
      >
        <Avatar className='h-9 w-9'>
          <AvatarImage src='/avatars/01.png' alt='Avatar' />
          <AvatarFallback>OM</AvatarFallback>
        </Avatar>
        <div className='ml-4'>
          <AlertTitle className='text-sm font-medium leading-none'>
            {order.seller} - Uusi kuljetus!
          </AlertTitle>
          <AlertDescription className='text-sm'>
            Tilaus ID: {order.requestId || Math.floor(Math.random() * 1000)} |
            Lisätty: {formatDateToFinnish(order.createdAt)} | Delivery Date: {' '}
            {order.estimatedDeliveryDate || 'Ei sovittu'}
          </AlertDescription>
        </div>
        <div className='ml-auto font-medium'>
          Asiakas ID: {order.clientNumber}
        </div>
      </Alert>
    ))}
  </div>
)
```

KUVA 10 Ilmoituslistan JSX-rakenne

## 6 KEHITYSKOhteET JA JATKOKEHITYS

Kehityskohteet ovat tärkeitä projektin jatkuvassa parantamisessa ja ne auttavat ohjaamaan tulevaa kehitystyötä. Tässä opinnäytetyössä erityisesti projektin etusivun viimeistely nousi esille kehityskohteena. Demoversioon etusivua ei vielä viimeistely, sillä tuotteistamisvaihetta ei ollut tässä vaiheessa suunniteltu saavutettavaksi. Myös sivun komponenttien yhtenäisyyden parantaminen on tunnistettu kehityskohteeksi, mikä edistää käyttökokemuksen johdonmukaisuutta ja esteettistä ilmettä.

Jatkokehityksen kohteet keskittyvät asiakaskokemuksen parantamiseen. Tulevaisuudessa suunnitelmissa on parantaa asiakkaiden saavutettavuutta, mikä voi sisältää esimerkiksi käyttöliittymän suunnittelun uudelleenarvioinnin tai esteettömyysominaisuuksien lisäämisen. Aikataulujen varaamisen helppous ja toimitusprosessin sujuvuus ovat myös avainalueita, jotka edistävät asiakkaan ja yrityksen välistä kanssakäymistä. Lisäksi varastonhallinnan kehittäminen on keskeistä, jotta voidaan varmistaa tuotteiden saatavuus ja toimitusten tehokkuus. Nämä kehitystoimet auttavat varmistamaan, että palvelu vastaa jatkuvasti asiakkaiden odotuksia ja liiketoiminnan tarpeita.

Data-analyysin jatkokehitys käynnistyy, kun demoversion testauksesta saadaan tuloksia ja päästään määrittelemään tarvittavat työkalut datan esittämiseen ja analysointiin. Tämä mahdollistaa tarkemman datan hyödyntämisen päätöksenteossa ja liiketoiminnan kehittämisessä.

## 7 YHTEENVETO

Tavoitteiden saavuttaminen on olennainen tekijä kaikissa projekteissa, mutta se korostuu erityisesti tuotteistamisprojekteissa. Projektin tavoitteet on määriteltävä siten, että ne ovat selkeät, mitattavat ja realistisesti saavutettavissa, mikä mahdollistaa projektin edistymisen seurannan ja lopputuloksen arvioinnin. Tiimityö on tavoitteiden saavuttamisessa avainasemassa, ja jokaisen tiimin jäsenen panosta arvostetaan. Tavoitteiden saavuttamiseksi vaaditaan myös jatkuvaa kommunikaatiota ja yhteistyötä kaikkien sidosryhmien kesken. Tavoitteet voivat liittyä palvelun laadun parantamiseen, asiakastyytyväisyyden kasvattamiseen, markkinaosuuden kasvuun tai sisäisten prosessien tehostamiseen. Onnistuneen projektin myötä organisaatio voi oppia uusia taitoja ja kehittää toimintatapaansa. Tämän opinnäytetyön myötä asetetut tavoitteet saavutettiin aikataulun mukaisesti, ja projektin kehittyessä tavoitteita tarkistettiin ja päivitettiin tarpeen mukaan. Projektin onnistuminen ilmeni tilaajan tyytyväisyydessä lopputulokseen. Projektissa saavutettu kehitys päättyi demoversion viimeistelyyn, ja on tarkoitus jatkaa kehitystä sisäisten testien päätyttyä. Opinnäytetyö onnistui parantamaan tilaajan yrityksen toimintatapoja, teki niistä selkeämpiä ja suoraviivaisempia, ja tämä saavutettiin miettimällä ja toteuttamalla projektin aikana uusia toimintoja.

Opinnäytetyössä oman oppimisen kehitys oli merkittävässä roolissa, erityisesti uusien haasteiden kohtaamisessa ja niiden ratkaisemisessa. Projektin aikana tuli vastaan uutta teknologiaa ja koodauskieliä, jotka herättivät suurta kiinnostusta ja tarjosivat mahdollisuuksia oppia ja soveltaa näitä uusia taitoja projektin selkeyttämiseksi ja sen kehittämisen vahvistamiseksi. Tämä oppimiskokemus ei ainoastaan lisännyt teknistä osaamista, vaan myös vahvisti kykyä soveltaa tietoa käytännön ongelmien ratkaisemiseksi.

Lisäksi projektin aikana kehittyi kyky järjestää ja hallita projektin kehitysaikatauluja tehokkaasti. Tämä sisälsi tilaajan toiveiden kuuntelemisen ja tavoitteiden muokkaamisen niiden mukaisesti. Kyky mukautua ja reagoida tilaajan tarpeisiin osoittautui arvokkaaksi taidoksi, joka auttoi ohjaamaan projektin suuntaa ja varmistamaan, että lopputulos vastasi sekä tilaajan että projektin asettamia odotuksia. Tämä oppimiskokemus on osoittanut, että joustavuus, sopeutumiskyky ja tehokas viestintä ovat avainelementtejä projektin onnistumisessa.

## LÄHTEET

- Fintraffic. (16. 11. 2023). *Sähköinen rahtikirja: Fintraffic*. Haettu 7. 12. 2023 osoitteesta Fintraffic: <https://www.fintraffic.fi/fi/uutiset/sahkoisista-rahtikirjoista-tehoa-kuljetuksiin-koko-euroopassa>
- Github. (2023). *Understanding the GitHub flow" - GitHub Docs*. Haettu 16. 12. 2023 osoitteesta GitHub Docs: <https://docs.github.com/en/get-started/quickstart/github-flow>
- Haikala, I.;& Mikkonen, T. (2011). *Ohjelmistotuotannon käytännöt*. Helsinki: Talentum.
- Logistiikan maailma. (2023). *Logistiikan historiaa: Logistiikan maailma*. Haettu 7. 12. 2023 osoitteesta Logistiikan maailma: <https://www.logistiikanmaailma.fi/aineistot/logistiikka-lukiolaisille/mita-on-logistiikka/>
- Logistiikan maailma. (2023). *Älylogistiikka: Logistiikan maailma*. Haettu 7. 12. 2023 osoitteesta Logistiikan maailma: <https://www.logistiikanmaailma.fi/logistiikka/digitalisaatio/alylogistiikka/>
- Microsoft. (26. 10. 2023). *Tietokannan suunnittelun perusteet: Microsoft*. Haettu 7. 12. 2023 osoitteesta Microsoft support: <https://support.microsoft.com/fi-fi/topic/tietokannan-suunnittelun-perusteet-eb2159cf-1e30-401a-8084-bd4f9c9ca1f5>
- Microsoft. (28. 6. 2023). *What is Azure Boards: Microsoft*. Haettu 16. 12. 2023 osoitteesta Learn microsoft: <https://learn.microsoft.com/en-us/azure/devops/boards/get-started/what-is-azure-boards?view=azure-devops>
- Radix. (2023). *Dokumenttaatio: Radix-ui verkkosivusto*. Haettu 5. 12. 2023 osoitteesta Radix verkkosivusto: <https://www.radix-ui.com/themes/docs/overview/getting-started>
- Rusanen, L. (23. 11. 2023). *Tekoäly tulee – onko organisaatiosi valmis?: Software Finland*. Haettu 7. 12. 2023 osoitteesta Software Finland: <https://softwarefinland.fi/blogi/tekoaly-tulee-onko-organisaatiosi-valmis/>
- Tuominen, T.;Järvi, K.;Lehtonen, M. H.;Valtanan , J.;& Martinsuo, M. (2015). *Aaltodoc: Palvelujen tuotteistamisen käsikirja*. Haettu 19. 12. 2023 osoitteesta Aalto University: <https://aaltodoc.aalto.fi/server/api/core/bitstreams/e28a24d7-a962-4cc2-a147-fe286705f936/content>
- Vercel. (2023). *Deploying: Nextjs*. Haettu 16. 12. 2023 osoitteesta Nextjs Web site: <https://nextjs.org/docs/app/building-your-application/deploying>
- Vercel. (2023). *Next.js verkkosivu*. Haettu 5. 12. 2023 osoitteesta Vercel: <https://nextjs.org/docs>
- Vercel. (2023). *Optimizations: Nextjs*. Haettu 16. 12. 2023 osoitteesta Vercel: <https://nextjs.org/docs/app/building-your-application/optimizing>