



SAVONIA

OPINNÄYTETYÖ - AMMATTIKORKEAKOULUTUTKINTO
TEKNIIKAN JA LIIKENTEEN ALA

UNIFAUN API- INTEGRAATIO TOIMINNANOHJAUSJÄRJESTELMÄÄN

TEKIJÄ/T:

Marko Mustonen

Tiivistelmä

Koulutusala Tekniikan ja liikenteen ala	
Tutkinto-ohjelma Tietotekniikan tutkinto-ohjelma	
Työn tekijä(t) Marko Mustonen	
Työn nimi UNIFAUN API- INTEGRAATIO TOIMINNANOHJAUSJÄRJESTELMÄÄN	
Päiväys 2.6.2021	Sivumäärä/Liitteet
Toimeksiantaja/Yhteistyökumppani(t) Applirent Oy	
<p>Tiivistelmä</p> <p>Opinnäytetyössä tutustuttiin rajapinnan toimintaan ja sen integrointia toiminnanohjausjärjestelmään. Tarkoituksena ja tavoitteena oli saada toimiva käyttöliittymä kuljetustilauksille toiminnanohjausjärjestelmään. Tämä toi olemassa olevaan toiminnanohjausjärjestelmään joustavuutta jokapäiväiseen työskentelyyn järjestelmän käyttäjille.</p> <p>Työ toteutettiin kehitystyönä. Työn tarkoituksena oli tehdä integraatio Applirent TK:n ja Unifaun REST API rajapinnan välille. Näin mahdollistettiin kuljetustilausten tekeminen suoraan Applirent TK- Järjestelmästä kirjautumatta erilliselle web-sivustolle. Työtä tuki Unifaunin tarjoama dokumentointi rajapinnasta ja sen toiminnasta. Kuljetustilaukseen tarvittavia toimintoja oli kuljetuksen tilaus, poisto, muokkaus ja haku. Näillä ominaisuuksilla saatiin toimiva integraatio, jolla voitiin työskennellä täysipäiväisesti.</p> <p>Tuloksena saatiin toimiva tilauskäyttöliittymä kuljetustilauksille. Käyttöliittymässä on vielä hieman puutteita ja tästä syystä käyttäjäkokemusta ei saatu vaikka se kuului osaksi työtä. Työssä ilmenneet ongelmat jäivät ratkottavaksi tulevaisuuteen. Tuotantoon menevä käyttöliittymä tulee, kun ongelmat ovat ratkottu. Työ tuo tulevaisuudessa kuitenkin paljon helpotusta ja joustavuutta Applirent TK:n käyttäjille, kun toimiva käyttöliittymä saadaan käyttäjille.</p>	
Avainsanat	

Field of Study Technology, Communication and Transport	
Degree Programme Degree Programme in Information Technology	
Author(s) Marko Mustonen	
Title of Thesis Integration between Applirent TK and Unifaun API	
Date 2 June 2021	Pages/Appendices
Client Organisation /Partners Applirent Oy	
<p>Abstract</p> <p>The topic of this thesis was an application programming interface and its integration into the ERP system. The purpose and goal were to have a functional interface for transport orders in the ERP system. This brings flexibility to the existing ERP system for everyday work for software users.</p> <p>A more precise purpose of the thesis was to make an integration between the Applirent TK and the Unifaun REST API (application programming interface). This way users can make transport orders directly by Applirent TK software without logging into a separate website. The thesis was supported by documentation by Unifaun about the application programming interface and its operations. Needed functions for transport orders were order, deletion, adaptation, and search functions. With these functions, a working interface was provided for everyday work.</p> <p>The result was a working user interface for transport orders. There are some shortcomings and for that reason user experience was not gained even it was part of the thesis. The problems that occurred in the thesis remained to be resolved in future. The user interface will go into production when the problems are solved.</p> <p>However, the results of the thesis will bring a lot of relief and flexibility to Applirent TK users in the future when the application programming interface is available to users.</p>	
Keywords	

KÄYTETTY SANASTO	6
1 JOHDANTO	7
2 TILAAJA.....	8
3 TYÖN TOTEUTUS JA SEN KUVAUS	9
3.1 Rajapinta.....	9
3.1.1 HTTP.....	10
3.1.2 JSON.....	11
3.1.3 Metodit ja resurssit.....	12
3.1.4 REST API.....	12
3.2 Käyttöliittymä	12
3.2.1 ActiveX.....	13
3.2.2 West-Wind.....	13
4 SUUNNITTELU	14
4.1 Käyttöliittymän suunnittelu	14
4.2 Integraation suunnittelu	14
4.3 Tietokannan suunnittelu	16
5 TOTEUTUS.....	17
5.1 Rajapinnan testaus	17
5.2 Tilauskäyttöliittymä	18
5.3 Tulostusliittymä	23
5.3.1 ActiveX PDFViewer-komponentti	24
5.4 Ylläpito- ja selausliittymä	25
5.4.1 Ylläpito.....	25
5.4.2 Selaus	26
5.5 Virheen käsittely	27
6 JATKOKEHITYS	28
7 POHDINTA.....	29
8 LÄHDELUETTELO.....	30

KUVALUETTELO

Kuva 1. Esimerkki yksinkertaisesta http-kutsusta.....	10
Kuva 2. Esimerkki http-vastauksesta	10
Kuva 3. Esimerkki JSON-muotoisesta tiedosta	11
Kuva 4. Tilaukseyttöliittymä tilauksen tiedot.....	19
Kuva 5. Rahdin tyyppi	19
Kuva 6. Kuljetustilaukseen liittyvä rahti.....	20
Kuva 7. Kuljetusliikkeiden näkymä Applirent TK:ssa.....	20
Kuva 8. Kuljetusliikkeiden palvelujen näkymä Applirent TK:ssa.....	21
Kuva 9. Lisäominaisuuden tietojen kirjaus.....	21
Kuva 10. Kuljetukseen valitut lisäosat	22
Kuva 11. Tulostuskäyttöliittymä.....	23
Kuva 12. Ulkopuolisen ActiveX-komponentin lisääminen	24
Kuva 13. Kuva ylläpitoikkunasta	26
Kuva 14. Kuva selausikkunasta	27

TAULUKKOLUETTELO

Taulukko 1. Yleisimmät HTTP-tilakoodeja (Vartiainen, 2020).....	11
Taulukko 2 Esimerkkejä CRUD operaatioista (Vartiainen, 2020).....	12
Taulukko 3. Tarvittavat resurssit	15
Taulukko 4. Luodut taulut ja niihin liittyvät sisällöt sekä relaatiot.....	16
Taulukko 5. Tarvittavien JSON-objektien sisällöt.....	18
Taulukko 6. PDFViewer-komponentissa käytetyt toiminnot.....	25

KÄYTETTY SANASTO

- API - Ohjelmointirajapinta (engl. Application programming interface, API) on määritelmä, jonka mukaan eri ohjelmat voivat tehdä pyyntöjä ja vaihtaa tietoja eli keskustella keskenään
- Applirent TK – Tilaaajan ERP, johon mainittu integraatio tehdään
- DLL-tiedosto – On koottu kirjasto, joka sisältää menettelyjä tai ohjaimia, joihin Windows-ohjelma viittaa ja suorittaa. Välttämättömiä tiettyjen ohjelmien suorittamiseen.
- Endpoint – URL tai päätepiste minne HTTP-kutsuja tehdään. Jokainen Endpoint on yksilöity tiettyyn toimintoon.
- ERP - Toiminnanohjausjärjestelmä. Yrityksen tietojärjestelmä, joka ohjaa yrityksen toimintaa
- FoxPro - Ohjelma, jolla kehitetään ERP:tä. Lyhennetty Microsoft Visual FoxProsta
- HTTP-kutsu – Siirtoprotokolla, jota käytetään tiedonsiirtoon.
- Microsoft Visual FoxPro – Ohjelma, jolla kehitetään ERP:tä.
- Ohjelmistokehys – eng. Framework. Ohjelmistokehys on ohjelmoinnin apuväline, jonka tarkoituksena on nopeuttaa uusien ohjelmistotuotteiden valmistusta. Ovat myös joskus välttämättömiä ohjelman toiminnassa
- PDF-dokumentti – Ohjelmistoriippumaton siirrettävä tiedostomuoto. Käytetään sähköiseen julkaisemiseen, tulostamiseen ja painamiseen.
- PDFViewer – WiscomSoft nimisen yrityksen tarjoama lisäosa Visual FoxProhon. Tämän avulla esikatsellaan ja tulostetaan PDF-dokumentteja.
- REST - (eng.Representational State Transfer) on HTTP-protokollaan perustuva arkkitehtuurimalli ohjelmointirajapintojen toteuttamiseen.
- TK - Tilaaajan ERP, johon mainittu integraatio tehdään. Lyhennetty Applirent TK:sta.
- Unifaun - Kuljetushallintajärjestelmien toimittaja.
- Unifaun Online – Unifaunin käyttäjälle suunnattu käyttöjärjestelmä, jolla hallinnoidaan kuljetuksia Unifaun Onlinesta.
- Unifaun rajapinta / API – Integroitava rajapinta. Tehdään pyyntöjä TK:sta Unifaun Onlinen rajapinnan kautta.
- ViscomSoft – Yritys, joka tarjoaa ActiveX-komponentin (PDFViewer)
- Uniikki tunniste / ID – Asiaan tai tapahtumaan liittyvä tunniste/id yksilöi asian tai tapahtuman. Näin voidaan hallinnoida useita samankaltaisia asioita ja tapahtumia.

1 JOHDANTO

Opinnäytetyön aiheena on käyttöliittymän tekeminen kuljetustilausta varten sekä siihen tarvittava integraatio REST-rajapintaan. Työssä tutkitaan REST-rajapintojen toimintaa ja tarkoituksena saada toimiva yhteys Applirent TK-järjestelmän sekä Unifaun REST API-rajapinnan välille. Työ pitää sisälleen myös toimivan käyttöliittymän luomisen Applirent TK-järjestelmään mikä on Applirent Oy:n tarjoama toiminnanohjausjärjestelmä. Työn merkitys tilaajalle on suuri, tuoden arvoa ja myyntivalttia Applirentille. Nykyiset asiakkaat ovat toivoneet kyseistä toimintoa järjestelmään.

Vastaavanlaista integraatiota ei ole Applirent TK järjestelmään tehty, mutta samankaltaisia integraatioita on suoritettu erilaisiin rajapintoihin. Nykyinen toiminnanohjausjärjestelmä on luotu 2000-luvun vaihteessa ja sen tekniikat voivat tuoda joitakin ongelmia liittämiseen. Tuote on myös tarkoitus viedä tulevaisuudessa uuteen toiminnanohjausjärjestelmään Applirent RESO:n.

Opinnäytetyö tuo tärkeän lisän Applirentin järjestelmään lisäarvon ja myyntivaltin kohentumiseksi. Applirent TK:n asiakkaille tämä tuo joustavuutta, ketteryyttä ja helpotusta jokapäiväiseen työskentelyyn järjestelmän parissa. Tuleva työni Applirent TK:n parissa antaa minulle paremmat valmiudet työelämään sekä jokapäiväiseen työskentelyyn.

2 TILAAJA

Työn tilaajana toimii Applirent Oy. Applirent kehittää toiminnanohjausjärjestelmää vuokratoimialalle ja kalustonhallintaan. Applirentin asiakaskunta on laaja. Pienistä rakennuskonevuokraamoista suuriin maanrakennuskaluston vuokraajiin ja jopa sote alalle. Pääosin Applirent TK:n asiakkaat ovat vuokraamoita, jotka vuokraavat kalustoa heidän asiakkailleen. Applirent TK:ta käytetään myös paljon sisäiseen kaluston hallintaan.

3 TYÖN TOTEUTUS JA SEN KUVAUS

Aiheesta tiedetään todella kattavasti. Unifaun tarjoaa kattavan API-dokumentoinnin heidän rajapinnastaan. Unifaunin API-dokumentointi antaa hyvän käsityksen mitä työ tulee pitämään sisällään. Tavoitteena on rakentaa toimiva integraatio Applirent TK:hon käyttäen Unifaunin rajapintaa. (Unifaun REST Api Documentation)

Integraatioon tarvittavat toiminnot ja niiden sisällöt

- Kuljetustilaukseen liittyvien tietojen lähetys Unifaunin rajapintaan.
- Kuljetustilaukseen liittyvien tulosteiden nouto Unifaun rajapinnasta.
- Kuljetusliikkeiden ja niiden palvelujen noutaminen Unifaunin rajapinnasta.
- Kuljetustilauksen poistaminen Unifaunin rajapinnasta.

Unifaun tarjoaa resurssit näille toiminnoille, joita tarvitaan toimivaan integraatioon.

Kuljetukseen liittyvät tiedot ovat aina yksilöllisiä ja jokainen kuljetus täytyy tilata yksilöidyillä tiedoilla. Tiedot pitävät sisällään lähettäjän ja vastaanottajan tiedot, rahdin ja kuljetusliikkeen sekä sen palvelut ja lisäominaisuudet. Osa tiedoista ovat valinnaisia ja osa pakollisia tietoja.

Tulosteet ovat myös aina yksilöllisiä ja Unifaun luo jokaiselle kuljetukselle tarvittavat asiakirjat. Näiden nouto tapahtuu uniikin tunnisteiden avulla rajapinnasta. Kuljetustilauksen yhteydessä saadaan uniikki tunniste tulosteisiin liittyen.

Kuljetusliikkeet ja niiden palvelut sekä lisäominaisuudet tulostetaan käyttäjälle suoraan rajapinnasta. Nämä noudetaan aina kuljetustilausta tehdessä rajapinnasta. Näin varmisetaan, että edellämainitut tiedot ovat aina ajantasalla.

Kuljetusten poistaminen tapahtuu kuljetuksen uniikin tunnisteiden avulla.

3.1 Rajapinta

Rajapinta on niin sanottu tiedonvälittäjä, jonka avulla voidaan lähettää tietoa toiseen palveluun sekä vastaanottaa tietoa. Toimivaan integraatioon tarvitaan saumaton keskusteluyhteys kahden palvelun välillä.

Työssä integroitava rajapinta oli tehty REST API-vaatimuksia mukailleen. REST API on suosittu webarkkitehtuurimalli nykypäivän rajapintojen toteuttamiseen. Tämä on suosittu tekniikka nykypäivänä, joka hyödyntää tiedon välityksessä JSON-muotoista tietoa. Myös muita tiedostomuotoja voidaan välittää REST-rajapinnassa. JSON tarkoittaa formaattia, jota käytetään tiedon tallentamiseen ja siirtämiseen.

Luvussa käydään läpi yleisesti kuinka rajapinnat toimivat ja mitä rajapinnan toimintoja kyseisessä työssä tarvittiin.

3.1.1 HTTP

HTTP (HyperText Transfer Protocol) on selainten ja WWW-palvelimien käyttämä siirtoprotokolla tiedonsiirtoon. Tämä on tärkeä asia tietää työtä tehdessä, koska kyseinen rajapinta perustuu http-protokollan ominaisuuksiin.

HTTP toimii lähettämällä ja vastaanottamalla tietoa. Esimerkiksi sovelluksen käyttäjä lähettää pyynnön rajapintaan koskien kuljetusta, josta palvelin vastaa onko pyyntö onnistunut vai epäonnistunut tässä myöskin palautetaan kuljetukseen liittyvää tietoa käyttäjälle.

HTTP-pyyntö rakentuu monesta eri osasta. Yleisimmin pyynnössä on metodi eli verbi (GET, POST, PUT, DELETE) ja URL johon pyyntö tehdään. Näillä voidaan jo saada tehtyä onnistunut kutsu palvelimelle. Pyyntö sisältää myös otsikoita (headers), pyynnön esitysmuoto (Accept) ja palautettavan tiedon kieli (Accept-Language). HTTP-pyyntö sisältää myös monia muita kohtia.

Jos kyseessä on POST tai PUT metodi tarvitaan yleensä kutsuun mukaan kappale (body), joka lähettää tarvittavat tiedot palvelimelle (Vartiainen, 2020, s. 12). Esimerkki http-pyyntöstä kuvassa 1.

```

1 GET /ufoweb-prod-202103040749/rs-extapi/v1/meta/lists/partners?
   favorite=default&language=fi HTTP/1.1
2 Host: www.unifaunonline.se
3 Accept: application/json
4 Authorization: *****

```

Kuva 1. Esimerkki yksinkertaisesta http-kutsusta

HTTP-vastaus sisältää paljon samoja asioita kuin palvelimelle tuleva HTTP-pyyntö. Vastauksessa tärkein kuitenkin on tilakoodi, joka kertoo pyynnön tilan sekä selityksen mitä se tarkoittaa. Vastauksessa on myös otsikko tason tietoa sekä kappale (body) joka palauttaa mahdollisesti tietoa käyttäjälle. HTTP-vastaus esimerkki kuvassa 2.

```

Date: "Mon, 19 Apr 2021 09:24:11 GMT"
Server: "Apache"
Keep-Alive: "timeout=5, max=100"
Connection: "Keep-Alive"
Transfer-Encoding: "chunked"
Content-Type: "application/json;charset=UTF-8"

```

Kuva 2. Esimerkki http-vastauksesta

Yleisimmät tilakoodit sekä niiden tarkoitukset listattu taulukkoon 1. Tilakoodi voi johtua monesta eri tekijästä ja välttämättä käyttäjä ei voi vaikuttaa onko pyyntö ollut onnistunut vai epäonnistunut.

Taulukko 1. Yleisimmät HTTP-tilakoodeja (Vartiainen, 2020)

TILAKOODI	SELITYS	ESIMERKKEJÄ
1XX	Pyyntö kesken	100 continue
2XX	Onnistunut pyyntö	200 OK, 201 Created
3XX	Uudelleenohjaus	302 Found
4XX	Käyttäjävirhe	400 Bad Request, 403 Not Allowed
5XX	Palvelimen virhe	500 Internal Server Error

3.1.2 JSON

JSON (JavaScript Object Notation) on formaatti tiedon tallentamiseen ja siirtämiseen. JSON on yleinen tiedostomuoto datan vaihtamiseen HTTP-palvelimen sekä käyttäjän välillä. JSON-muotoinen data toimii avain-arvo-pareina. Lähetettävässä tiedossa on yleensä määritelty avain, jolle muodostetaan arvo käyttöliittymään syötetyillä arvoilla. Kuvassa 3 esimerkki JSON tiedostosta.

```

"href":"https://www.unifaunonline.se/rs-extapi/v1/shipments/15110099",
"id":"15110099",
"status":"PRINTED",
"shipmentNo":"136686772095",
"orderNo":"Joku tilaus numero varmaankin töhön",
"route":null,
"routingCode":null,
"reference":null,
"serviceId":"KKSTD",
"parcelCount":2,
"sndName":"Jyvöskylö",
"sndZipcode":"40320",
"sndCity":"40320 JYVÖSKYLÖ",
"sndCountry":"FI",
"sndQuickId":"1",
"rcvName":"Kuopio",
"rcvZipcode":"70780",
"rcvCity":"70780 KUOPIO",
"rcvCountry":"FI",
"rcvQuickId":"1",
"created":"2021-04-19T09:30:31.286+0000",
"changed":"2021-04-19T09:30:31.286+0000",
"shipDate":"2021-04-19T09:30:31.000+0000",
"returnShipment":false,
"normalShipment":true,
"consolidated":false,
"deliveryCode":null,
"profileGroup":"Perusprofiiliryhmä",
"parcels":[
  {
    "parcelNo":"373325382912485856",
    "returnParcelNo":null,
    "reference":null,
    "partOrderNo":null
  }
],

```

Kuva 3. Esimerkki JSON-muotoisesta tiedosta

3.1.3 Metodit ja resurssit

Resursseilla tarkoitetaan rajapinnan tarjoamia kohteita, josta saadaan tietoa. Resurssit erotellaan toisistaan URI:n (Uniform Resource Identifier) avulla, joka pitää sisällään resurssin nimen ja osoitteen. Näin tiedot saadaan eroteltua toisistaan.

Rajapinta on aina kuvattu jollakin tavalla ja dokumentoinnista saa tietoa käytettävissä olevista resursseista. Jokaiseen resurssiin on liitetty metodi, jota tarvitaan http-pyyntöä tehdessä. Näitä metodeja on paljon mutta käytetyimpiä ovat GET, POST, PUT, DELETE. Edellä mainitut toiminnot tunnetaan myös CRUD (Create, Read, Update, Delete) operaationa. Taulukossa 2 on kuvattu edellä mainitut metodit sekä niiden käyttökohteita.

Taulukko 2 Esimerkkejä CRUD operaatioista (Vartiainen, 2020)

METODI/CRUD	TOIMINTO	PYYNTÖ
POST/CREATE	Uuden resurssin luonti	POST esim.fi/api/data
GET/READ	Resurssin hakeminen	GET esim.fi/api/data
PUT/UPDATE	Resurssin päivittäminen tai uuden resurssin luominen	PUT esim.fi/api/data/id
DELETE/DELETE	Resurssin poisto	DELETE esim.fi/api/data/id

3.1.4 REST API

REST (Representational State Transfer) on yksinkertainen ja suosittu webarkkitehtuurimalli nykypäivän rajapintojen toteuttamiseen. REST API koostuu kahdesta avainsanasta asiakasohjelma ja resurssi. Asiakasohjelma voi olla esimerkiksi selain, käyttöliittymä tai kehittäjä, joka kutsuu resursseja. Esimerkiksi henkilö menee selaimella Facebook sivustolle, joka kutsuu resurssia mikä palauttaa näkymiä Facebook sivulle. Resursseilla on omat metodit ja toiminnot, jotka ovat määritelty yleensä API-dokumentoinnissa. (Avraham, 2017)

Vartiainen toteaa tekstissään (Vartiainen, 2020, s. 14) että REST:in suosio perustuu sen yksinkertaisuuteen; ”ominaisuudet nähdään käyttäjäystävällisenä, ja ohjelmoijien on helppo ymmärtää sen koodaamista”. REST on joustava tekniikka rajapintojen toteuttamisessa.

3.2 Käyttöliittymä

Käyttöliittymä toteutettiin ERP-järjestelmään nimeltä Appliren TK, joka on yksi tuote Applirentin tuoteperheestä.

TK on julkaistu vuosituhanen vaihteessa ja on kehitetty Microsoft Visual FoxPro 9.0:lla (Microsoft Visual FoxPro 9.0, 2016). Nykypäivänä kehitystyökalu on hieman vanhentunut, mutta integraation valmistumiseen sen ei oleteta vaikuttavan.

FoxPro on käyttöliittymäohjelmointiin soveltuva kehitystyökalu. FoxPro on jo hieman aikaansa jäljessä mutta hyvin toimiva ja tehokas työkalu käyttöliittymäohjelmointiin. FoxProlle tarjotaan myös ulkoisia lisäosia sekä ominaisuuksia. West-Wind (Strahl, ei pvm) tarjoaa paljon lisäosia FoxProlle ja tässä työssä hyödynnettiin kyseisen yrityksen tarjoamia lisäominaisuuksia. FoxPro tukee ActiveX-komponentteja, joita tarjoaa myös ulkopuoliset palveluntarjoajat.

3.2.1 ActiveX

ActiveX on sovelluskomponenttikerros Microsoft Windows-käyttöjärjestelmässä. Tekniikat pohjautuvat COM (Component Object Model) -arkkitehtuuriin. ActiveX on ohjelman komponentti objekti, jota voidaan käyttää ohjelman eri osioissa (Rykov, 2012, s. 20).

FoxPro tarjoaa monenlaisia ActiveX-komponentteja valmiina, jota voidaan lisätä ohjelmaan. Näitä komponentteja ovat esimerkiksi selain-, Windows Media Player- ja WordPad- komponentti. Näitä komponentteja voidaan upottaa käyttöliittymään ja käyttää tarvittaessa suoraan käyttöliittymässä.

3.2.2 West-Wind

West-Wind kehittää ja myy ohjelmistoja, työkaluja, ohjausta ja konsulttipalveluja. West-Wind tarjoaa paljon lisäominaisuuksia FoxPro:lle. Kehitettävään ERP-järjestelmään on hankittu lisäosana West-Wind Web Connection. Web Connection on täydellinen websovelluksen ohjelmistokehys tukemaan rajapintojen kehitystä. Kyseinen ohjelmistokehys tarjoaa myös muita helpotuksia FoxProlla kehittämiseen. Esimerkiksi ohjelmistokehyksestä löytyy funktioita, aikojen muokkaamiseen, JSON-tiedostojen muotoiluun sekä tekstin dekodaukseen ja enkoodaukseen. Näillä tarkoitetaan tekstin muokkaamista salattuun muotoon (enkoodaus) ja sen purkamista normaaliksi tekstiksi (dekoodaus). West-Wind tarjoaa dokumentoinnin kaikista funktioista, jotka kuuluvat Web Connection ohjelmistokehykseen (Strahl, ei pvm).

4 SUUNNITTELU

Työn suunnittelu aloitettiin toimeksiantajan kanssa. Mukana suunnittelussa oli myös Applirent TK:ssa useamman vuoden työskennellyt kollega. Työlle asetettiin selkeät suuntaviivat ja siihen kuului käyttöliittymän ja tietokantojen teko sekä integraatio Unifaun-APIin. Tämä antoi suuntaviivat työn aloitukselle ja edellä mainittujen kohteiden työstämiselle.

4.1 Käyttöliittymän suunnittelu

Käyttöliittymä vaati useita eri ikkunoita. Aluksi työstettiin uudet kuljetustilaus-, ylläpito- ja selausikkunat, jotka noudattavat TK:ssa olevien ikkunoiden toimintaperiaatetta. Ne pyrittiin pitämään samanlaisena muiden TK:n ikkunoiden kanssa.

Kuljetustilausikkunassa oli otettava huomioon kuljetustilauksen vaadittavat tiedot. Näitä olivat lähettäjän, vastaanottajan ja kuljetukseen liittyvät tiedot. Unifaun API-kuvaus määrittää pakolliset tiedot. Lisäksi sieltä löytyy valinnaista tietoa, jota voidaan hyödyntää kuljetusta tehdessä. Tarpeellisia tietoja mitä hyödynnetään yhteydenottoja varten, ovat lähettäjän ja vastaanottajan puhelinnumero ja sähköposti sekä tieto kuljetettavasta tavarasta, muun muassa mitat, sisältö ja viite. Nämä kaikki eivät ole pakollisia, mutta käytännöllisiä.

Kuljetusten ylläpito- ja selausikkunasta nähdään tilatut kuljetukset. Ylläpitovälilehdellä näkyy tietyn kuljetuksen tiedot, ja tätä kautta voidaan tarpeen vaatiessa poistaa kuljetus, tulostaa asiakirjat sekä tarkastella kaikkia tietoja kuljetuksesta. Nämä näkymät perustuvat pelkästään Applirent TK:n omaan tietokantaan ja muutosten tapahtuessa Unifaun Online- käyttöliittymässä tiedot eivät muutu Applirent TK:ssa. Tulevaisuudessa tähän voidaan tehdä joku ratkaisu, mutta alussa oletuksena on, että kuljetuksia hallinnoidaan ainoastaan Applirent TK:n kautta.

4.2 Integraation suunnittelu

Integraation suunnittelu aloitettiin yhdessä Unifaunin edustajan kanssa. Molemmat yritykset olivat tuntemattomia toisilleen, joten molemmat käyttivät aikaa omien yritysten esittelyyn.

Palaverin tarkoituksena olisi kartoittaa Applirent TK:hon tarvittavat käyttökohteet ja tarpeet Unifaunin rajapinnasta sekä kuinka Unifaun pystyy parhaiten nämä edellä mainitut kohdat parhaiten tarjoamaan ja kuinka niitä olisi helpoin hyödyntää.

Unifaun tarjoaa paljon eri resursseja heidän rajapinnastaan, mutta myös paljon sen kaltaisia mitä työssä ei tarvittu käyttää. Applirent TK:n käyttökohteet olivat kuljetustilaukset, asiakirjojen nouto ja tulostus sekä kuljetusliikkeiden, heidän palvelunsa ja lisäominaisuuden tuonti järjestelmään. Tarvittavat resurssit mainittu taulukossa 3.

Taulukko 3. Tarvittavat resurssit

RESURSSI/ENDPOINT	TOIMINTO
/SHIPMENTS (POST)	Uuden kuljetuksen luonti
/SHIPMENTS/SHIPMENTID (DELETE)	Tietyn kuljetuksen poisto
/META/LISTS/PARTNERS (GET)	Kuljetusliikkeiden ja niiden palvelujen listaus
/SHIPMENTS/{SHIP-METID}/PDFS/{PDFID} (GET)	Kuljetukseen liittyvien PDF-dokumenttejen nouto

Unifaun tarjoaa monia erilaisia kuljetustilauksia. Työhön tarvittiin näistä vain reaaliaikaista kuljetustilausta. Tästä saadaan tulostettua kuljetukseen tarvittavat asiakirjat tunnin sisällä. Tiedot kuitenkin tallennetaan omaan järjestelmään, josta ne ovat tulostettavissa myös myöhemmin. Tarvittavat tiedot kootaan kuljetustilaukseen järjestelmästä ja kuljetus tilataan kootuilla tiedoilla. Tästä tallentuu tieto Unifaun Onlineen, joka on käyttöjärjestelmä Unifaun apille. Tämän lisäksi tallentuu tieto myös oman järjestelmämme tietokantaan.

Tietojen poisto onnistuu rajapinnan palauttamasta tunnisteesta, joka on uniikki kaikille kuljetuksille. Tämä tieto menee kuljettavalle yritykselle, joka poistaa kuljetuksen järjestelmästä.

Asiakirjoille saadaan uniikki tunniste. Asiakirjojen noutoon tarvitaan edellä mainittu tunniste sekä kuljetuksen tunniste. Näiden avulla voidaan tunnin aikana noutaa asiakirjat järjestelmästä. Tässä tapauksessa kuitenkin asiakirjat tallennetaan omaan järjestelmään, josta ne ovat noudettavissa myös myöhemmin. Tulostus tapahtuu joko tilauksen yhteydessä tai myöhemmässä vaiheessa.

Unifaunilla on useita eri kuljetusliikettä, jotka ovat liittyneet Unifauniin. Näillä kuljetusliikkeillä on omat palvelut, jotka voidaan noutaa rajapinnasta. Applirent TK:n käyttäjä valitsee omat sopimuskuljetusliikkeensä rekisteröityessään Unifaun Onlineen, näin saadaan käytetyt kuljetusliikkeet listattua.

Kyseisellä listauksella saadaan tuotua käyttäjälle kuljetusliikkeiden palvelut ja lisäominaisuudet. Kaikkia listauksia ei saada julki, koska ne ovat Unifauni Onlinen kirjautumisen takana. Esimerkkejä kuljetusliikkeistä ovat Posti, Kaukokiito, Schenker, USP ja DHL.

Unifaunilla on kuljetusliikkeitä yli 200 (Unifaun kuljetusliikkeet, ei pvm). Kuljetusliikkeillä on omat palvelut ja lisäominaisuudet. Esimerkiksi käytetäänkö Postin rahtipalvelua, Posti Smart-Shippiä tai Schenkerin ulkomaan vai kotimaan rahtia. Lisäominaisuuksia ovat esimerkiksi ennakoilmoitus, kuljetuksen seuranta tai vaarallisten aineiden kuljetus. Täältä saadaan listattua myös kuljetettavan tavaran tyyppi: lava, kolli, häkki tai muu sellainen. Näitä palveluita ja lisäominaisuuksia on paljon riippuen kuljetusliikkeestä.

4.3 Tietokannan suunnittelu

Applirent TK:ssa tietokanta on toteutettu paikallisesti ohjelmatiedoston juureen. Tietokanta on relaatiotietokanta. Tietokantaan mietittiin tarvittavat taulut, joita on viisi kappaletta alustavassa suunnitelmassa. Tauluja lisätään tarpeen tullen. Taulujen nimeäminen on haastavaa koska maksimi merkkimäärä FoxProssa tauluille on 10 (kymmenen) merkkiä (Visual FoxPro System Capacities, 2007). Joissakin tilanteissa FoxPro katkaisee taulujen nimen, jos se ylittää yli 10 (kymmenen) merkkiä. Tämän vuoksi merkkimäärä pidetään mahdollisimman pienenä. Tämä vaikeuttaa taulujen nimeämisen kuvaavasti. Tietokannan taulut selitetty taulukossa 4.

Taulukko 4. Luodut taulut ja niihin liittyvät sisällöt sekä relaatiot

TAULUN NIMI	TAULUN SISÄLTÖ	RELAATIO
KULJTIL	Sisältää otsikkotason tietoa. Kuljetusliike, lähettäjä/vastaanottaja tiedon, kuljetettavan tavaratiedot, lisäominaisuudet	Lähettäjä id ktlsendr tau-luun Vastaanottaja id ktilrecvr tau-luun Kuljetuksen si-sältö ktilparc tauluun
KTILPARC	Sisältää tiedot kuljetettavasta tavarasta. Paino, mitat, viitteet, lisätiedot	Kuljetus ID kuljtil taulusta
KTILRECVR	Sisältää tiedot vastaanottaja osapuolesta. Osoitetiedot, yhteystiedot, vastaanottajan viite.	Kuljetus ID kuljtil taulusta
KTILSENDR	Sisältää tiedot lähettäjä osapuolesta. Osoite- ja yhteystiedot, lähettäjän viite.	Kuljetus ID kuljtil taulusta
KTILPRNTS	Sisältää tiedot tiettyyn kuljetukseen liittyvistä dokumenteista	Kuljetus ID kuljtil taulusta

5 TOTEUTUS

Toteutus kertyi monesta eri osiosta. Toimivaan järjestelmään tarvittiin:

- Toimiva ja visuaalisesti kuvaava ja yksinkertainen käyttöliittymä.
- Oikeamuotoisen datan luonti sekä sen lähetys rajapintaan.
- Tarvittavien tietojen tallennus tietokantaan ja niiden oikea oppinen käyttö käyttöliittymässä.
- Asiakirjojen tulostaminen palvelimelta.

Osoita käsiteltiin ja työstettiin eri näkökulmista ja niitä myös päivitettiin työn edetessä

Käyttöliittymä täytyi pohtia paljon sen pohjalta mihin käyttäjät ovat tottuneet niin TK:ssa jokapäiväisessä työssä niin kuin tehdessä kuljetustilausta joltain web-sivulta. Näiden pohjalta on helppo siirtymä käyttämään kuljetustilausta TK:sta käsin.

Lähetettävän tiedon luonti oli keskeinen asia toimivassa käyttöliittymässä. Käyttöliittymän tarvitsi olla tarpeeksi kuvaava, että se ei jätä tulkinnan varaa, kuinka tietoja täytetään käyttöliittymässä. Tiedon tarkistaminen ja sen muokkaus lähetettävään muotoon pitää myös huolehtia ennen kuljetustilauksen lähettämistä.

Tietokantaan tallennetaan samat asiat mitä viedään rajapintaan. Täältä myös palautuvat uniikit tunnisteet otetaan talteen myöhempää tarvetta varten.

Asiakirjojen tulostus tapahtuu palvelimelta, jonne rajapinnasta haetut tulosteet tallennetaan. Tämä toiminto tehty sen takia että voidaan käyttää tulosteita myös myöhemmin.

5.1 Rajapinnan testaus

Uusi kuljetustilaus toimii lähettämällä JSON-tiedosto rajapintaan tiettyyn endpointtiin. Endpointilla tarkoitetaan tiettyä URL-osoitetta, joka viittaa palvelimeen, jossa Unifaun API sijaitsee. Nämä ovat uniikkeja ja jokaisella on omat toimintonsa ja tietovarastonsa. Aluksi kirjoitettiin JSON-tiedosto, joka lähetettiin Postman-nimisellä ohjelmalla rajapintaan. Postmania käytetään HTTP-kutsujen tekemiseen testauksessa. Tästä sai hyvän käsityksen, kuinka rajapinta toimii ja kuinka Unifaun Online käyttöliittymä näyttää tilatut kuljetukset. Kuljetustilaukseen liittyy paljon kerättävää tietoa. Tiedot ovat joko pakollisia tai valinnaisia. Tarvittavan JSON-tiedoston tulee sisältää tietynlaisia objekteja. Taulukossa 5 kuvattu sanallisesti JSON-objektien rakenteet.

Taulukko 5. Tarvittavien JSON-objektien sisällöt

JSON-OBJEKTI	KUVAUS
LÄHETTÄJÄ	Lähetäjä osapuolen yhteystiedot ja viite.
VASTAANOTTAJA	Vastaanottaja osapuolen yhteystiedot ja viite
RAHTITAVARA	Kuljetettava tavara. Voidaan määritellä mitat, paino, tyyppi (lava, kolli, hikki jne.), kuvaus tavarasta.
PALVELUT	Tähän liitetään kuljetusliikkeen ID mitä käytetään. Tähän myös voidaan määritellä palvelut ja lisäominaisuudet.

Unifaunin API-dokumentoinnista (Unifaun shipments, 2020) löytyy skeema (eng. schema) eli tietomalli joka kertoo mitä tietoa kullekin merkitykselle annetaan sekä esimerkki JSON, joka kuvaa täsmällisemmin mitä tietoja kuljetustilaus täytyy pitää sisällään.

Postmanilla tehdyissä testeissä sai hyvän käsityksen rajapinnan toiminnasta ja kuinka tiedot piirtyivät Unifaun Online-käyttöliittymään. Tästä oli joustava siirtyä luomaan kuljetuksia Applirent TK:n kautta. Jo olemassa oleva testidata toimi moitteettomasti joten sitä käytettiin kuljetustilauksen tekemisessä TK:n kautta. Tähän mennessä kuljetustilaukset olivat toimineet moitteettomasti ja dataa saatiin liikkumaan rajapintaan

Tämä järjestys helpotti seuraavaa vaihetta kuljetustilausten luonnissa. Kun testikuljetustilauksia tehtiin pelkällä tiedostolla tieto karttui ja käyttöliittymä oli helpompi työstää tiedostaen, mitä kaikkia kenttiä mahdollisesti tarvitaan onnistuneeseen kuljetustilaukseen.

5.2 Tilauskäyttöliittymä

Käyttöliittymää ei suunniteltu alustavasti, vaan mietittiin tarvittavat rajapintaa mukailevat ikkunat. Kuljetustilaukseen tarvittiin uusi ikkuna. Ikkunaa oli helppo suunnitella testeissä käytetyn datan perusteella. Se antoi hyvän käsityksen ja mallin mitä kenttiä tarvitaan tiedon keräämiseen.

Lähetäjälle ja vastaanottajalle tehtiin täytettävät tekstikentät tarvittaville tiedoille. Kuljetusten lähettäjät ja vastaanottajat pohjautuvat suurimassa osassa tapauksista jo olemassa oleviin asiakkaisiin ja toimipisteisiin, joten kenttien täyttö on automatisoitu. Molempien osapuolien kentät voidaan valita ylälaidassa olevilla pudotusvalikoilla toimipisteet ja "..."-painikkeesta avautuu asiakas -haku, josta voidaan täyttää automaattisesti myös asiakastiedot.

Kuvasta 4 käy ilmi lähettäjä/vastaanottaja osapuolten täytettävät kentät. Kuva on testausvaiheesta, joten kentät ja muotoilu voivat muuttua tuotantoon mennessä. Maksaja voidaan määritellä manuaalisesti asiakasnumerolla. Jos käyttäjä on rekisteröitynyt Unifaun Onlineen, niin maksajakenttä täyttyy automaattisesti kuljetusliikkeen asiakasnumerolla. Tämä voidaan noutaa myös paikallisesta tietokannasta, josta kuljetusliikkeen asiakasnumero voi löytyä esimerkiksi asiakkaalta.

Lähettäjä		Vastaanottaja	
Vp/asiakas	50 Joensuu	Vp/asiakas	44892
Kuljetusliike			
Palvelu			
Maksaja			
Asiakas/Vp	Joensuu	MARKO MUSTONEN	
Osoite	Pamilonkatu 22 b	ILMARISENTIE 8A59	
Postinumero	80100	70500	
Postitoimipaikka	Joensuu	KUOPIO	
Puhelinro	013 129 555	0449758519	
Sähköposti			
Viite			
Tekstiä			

Kuva 4. Tilaukseyttöliittymä tilauksen tiedot

Kuljetustilaukseen on määritelty pakolliseksi vähintään yksi kappale kuljetettavia tavaroita. Oletettavasti on turha tilata kuljetusta, jos kuljetettavaa tavaraa ei ole. Tämä on kuitenkin estetty Uni-faunin puolesta ja kuljetusta ei voi tilata ilman yhtään kuljetettavaa kappaletta.

Kuljetettavan tavaran sisällölle määritellään tyyppi, paino, kuvaus ja määrä. Tyyppi listataan valitun kuljetusliikkeen mukaan. Jokaisella on yksilöidyt tyytit kuljetettavalle tavaralle, jotka esiintyvät kuvassa 5. Tämä kuvaa tarkoituksen selkeästi. Kaikissa tapauksissa kuljetusliike ei tarjoa pakettityyppejä. Näissä tilanteissa tyyppikenttä jätetään tyhjäksi.

Paketin tyyppi	P2
Määrä	NE Pakkaamator
Paino	RO Rulla (RO)
Kuvaus	CW Rullakko (CW)
	JB Suursäkki (JB)
	BG Säkki (BG)
	P2 TEHO-lava (f)
	BA Tynnyri (BA)

Kuva 5. Rahdin tyyppi

Lisätyt rahdit listataan taulukkoon, joka näkyy kuljetustilausikkunassa. Virheen tapahtuessa muokaus toimintoa ei ole mutta voidaan poistaa ja lisätä uusi rahtitieto listaan tarvittaessa. Kuvassa 6 nähdään lisätyt rahdit.

Paketin tyyppi	P2	Kuvaus	Tyyppin kuvaus	Tyyppi	Paino	Määrä
Määrä	1	Tavaran sisältöä	TEHO-lava (P2)	P2	1	100
Paino	100					
Kuvaus	Tavaran sisältöä					
Lisää rivi						
		Poista rivi				

Kuva 6. Kuljetustilaukseen liittyvä rahti

Kuljetusliikkeet haetaan rajapinnasta. Tämä listaa kaikki kuljetusliikkeet mitä käyttäjälle Unifaun Onlineen on listattu. Nämä kirjataan kirjautumisvaiheessa ja niitä voidaan lisätä ja poistaa järjestelmästä tarpeen tullen. Edellä mainitusta syystä päädyttiin ratkaisuun, joka listaa kaikki reaaliajassa eli ne haetaan rajapinnan kautta pudotusvalikkoon.

Tilanteessa, jossa listattaisiin omasta tietokannasta kuljetusliikkeet ja palvelut ei voida taata reaaliaikaista listausta. Käyttäjän tulisi muistaa päivittää Unifaun Online kuljetusliikkeet, TK järjestelmän kuljetusliikkeet ja palvelut. Tätä ei ole toteutettu TK:n järjestelmässä aiemmin mutta toimivuus todettiin hyväksi ja valinta pysyi.

Ongelmakohtana huomattiin, että liikkeitä ja palveluja ei saada listattua, kun Unifaun API ei ole toiminnassa. Tähän ongelmaan ei puututtu, koska kuljetusten tilaaminenkin on mahdotonta, jos rajapinta ei ole toiminnassa.

Kuvasta 7 ja 8 huomataan kuinka kuljetusliikkeet ja valitun kuljetusliikkeen palvelut listautuvat pudotusvalikkoon. Palvelut ovat tietyn kuljetusliikkeen tarjoamia palveluita. Palvelut listautuvat kuljetusliikkeen perusteella ja näyttävät mitä palveluita on saatavilla.

Nämä kaikki ovat yksilöityjä valintoja ja tietoja mitkä jokainen kuljetusliike määrittelee.

Kuljetusliike	KK	
Palvelu	SBTL	DB Schenker
	ITELLA	Itella, Kotimaan rahti
Maksaja	KK	Kaukokiito
	MH	Matkahuolto
Asiakas/Vp	ITELLALO	Posti Oy, Kotimaan rahti
Osoite	Pamilonkatu 22 b	ILMARISEN
Postinumero	80100	70500
Postitoimipaikka	Joensuu	KUOPIO
Puhelinro	013 129 555	044975851

Kuva 7. Kuljetusliikkeiden näkymä Applirent TK:ssa

Vp/asiakas	50 Joensuu	Vp/asiakas	
Kuljetusliike	MH		
Palvelu	KKSTD		
Maksaja	MH10 Matkahuolto Bussipaketti		
Asiakas/Vp	MH3050 Matkahuolto Jakopaketti/Paikallispaketti		ONEN
Osoite	MH34 Matkahuolto Kotijakelu		8A59
Postinumero	MH57 Matkahuolto Lavarahiti		
Postitoimipaikka	MH81 Matkahuolto Lähellä-paketin palautus	Joensuu	KUOPIO
Puhelinno	MH80 Matkahuolto Lähellä-paketti	013 129 555	0449758519
Sähköposti	MH74 Matkahuolto Mannermaan Paketti		

Kuva 8. Kuljetusliikkeiden palvelujen näkymä Applirent TK:ssa

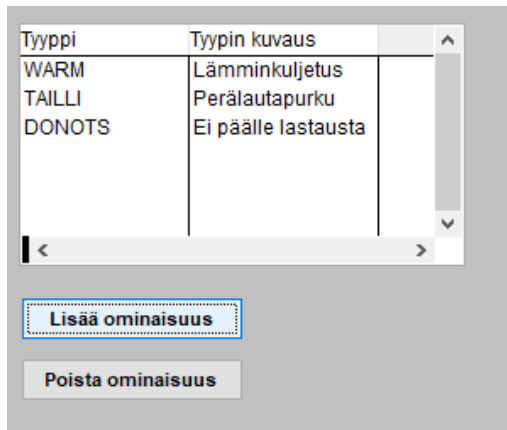
Kuljetusten lisäominaisuuksien valinta tapahtuu listaamalla ominaisuudet samalla tavalla kuin kuljetusliikkeet ja heidän palvelunsa. Lisäominaisuuksien listaus ja niiden näyttäminen tuottivat ongelmia käyttöliittymän muotoiluun. Lisäominaisuuksilla voi olla erilaisia täytettäviä kenttiä. Tähän tarvittiin automaatiikka, joka listaa tarvittavat kentät tekstille, listatuille arvoille ja valinta ruutu kyllä/ei mahdollisuuksille. Esimerkki lisäominaisuuksista kuvassa 9.

Mistä	<input type="text"/>
Mihin	<input type="text"/>
Kopio	<input type="text"/>
Piilokopio	<input type="text"/>
Virhe	<input type="text"/>
Viesti	<input type="text"/>
Kieli	<input type="text"/>
Lähetä sähköpost	<input checked="" type="checkbox"/> Lähetä sähköpost
Malli	<input type="text"/>

Kuva 9. Lisäominaisuuden tietojen kirjaus

Kuvassa 9 on valittu jokin ominaisuus kuljetukselle, johon täytetään tarvittavat tiedot. Lisäominaisuuksia on paljon ja niille annettavat tiedot ovat erilaisia. Tiedot voivat olla myös pelkästään otsikkotasolla eikä mitään lisätietoja ominaisuuteen anneta.

Ominaisuuksia voi olla useampi yhteen kuljetukseen liittyen ja ne lisätään painikkeella. Listassa näkyy kaikki valitut lisäominaisuudet. Tässä myöskään ei ole huomioitu muokkaamista vaan se tapahtuu poistamalla ja lisäämällä uudestaan haluttu lisäpalvelu. Kuvassa 10 on esimerkki käyttöliittymästä, jolla voidaan lisätä ja poistaa lisäominaisuuksia sekä nähdään kuinka lisäominaisuudet listautuvat taulukkoon.



Kuva 10. Kuljetukseen valitut lisäosat

Lisäominaisuudet tuottivat useita eri ongelmia ja mietinnän kohteita kuljetuksen tilaukseen. Unifaun ei tarjoa palvelussaan kenttien pakollisuutta ja näitä tietoja on vaikea arvata. Toisin sanoen ohjelmassa saadaan virhe pelkästään itse todentamalla se. Tämä tuo myös jatkossa ongelmia, koska ohjelman käyttäjä ei välttämättä tunnista Unifaunista tulevaa virhetekstiä eikä osaa tästä syystä päätellä kuinka toimia.

Kyseiseen ongelmaan ei tässä vaiheessa suuremmalti puututtu vaan jatkettiin tekemistä. Tähän todennäköisesti tulee myöhemmin asiakaskohtaiset ratkaisut, jossa pohditaan asiaa asiakkaan näkökulmasta. Ongelma oli tiedossa ja jotain ratkaisuja tähänkin mietittiin.

Kuljetustilaus tapahtuu, kun aiemmat vaihtoehdot ovat täytetty. Pakollisia tietoja tässä ovat vastaanottaja ja lähettäjäosapuolet, niiden tiedot ja vähintään yksi rivi rahtia. Näillä tiedoilla kuljetus saadaan tilattua.

Lisäominaisuudet ovat vaihtoehtoisia, mutta niitä valitessa on täytettävä pakolliset kentät. Kuljetustilaus lähtee painamalla "tilaa kuljetus" painiketta, joka sijaitsee käyttöliittymän alalaidassa. Ohjelma ilmoittaa, mikäli pakollisten kenttien tiedot ovat puutteelliset ja vaativat täydentämistä.

Kuljetustilaus lähtee Unifauniin ja sieltä tulee tieto käyttäjälle onnistuneesta tilauksesta. Virheen tapahtuessa tulee ilmoitus, että kuljetusta ei ole saatu tilattua ja mahdollisuuksien mukaan pyritään tiedottamaan käyttäjää, mistä virhe johtui. Ilmoittaminen on rajallista, koska normaali käyttäjä ei saata tunnistaa kaikkia virhekoodeja tai -tekstejä mitä tilauksen yhteydessä voi ilmaantua.

Onnistuneen tilauksen jälkeen ohjelma ohjaa käyttäjän siirtymään tulostusikkunaan tai vaihtoehtoisesti käyttäjä voi tulostaa tarvittavat asiakirjat myöhemmässä vaiheessa. Onnistuneesta tilauksesta tehdään päivitys tietokantaan. Muussa tapauksessa tietokantaa ei päivitetä. Virhetilanteen sattuessa virheet kirjataan tekstitiedostoon.

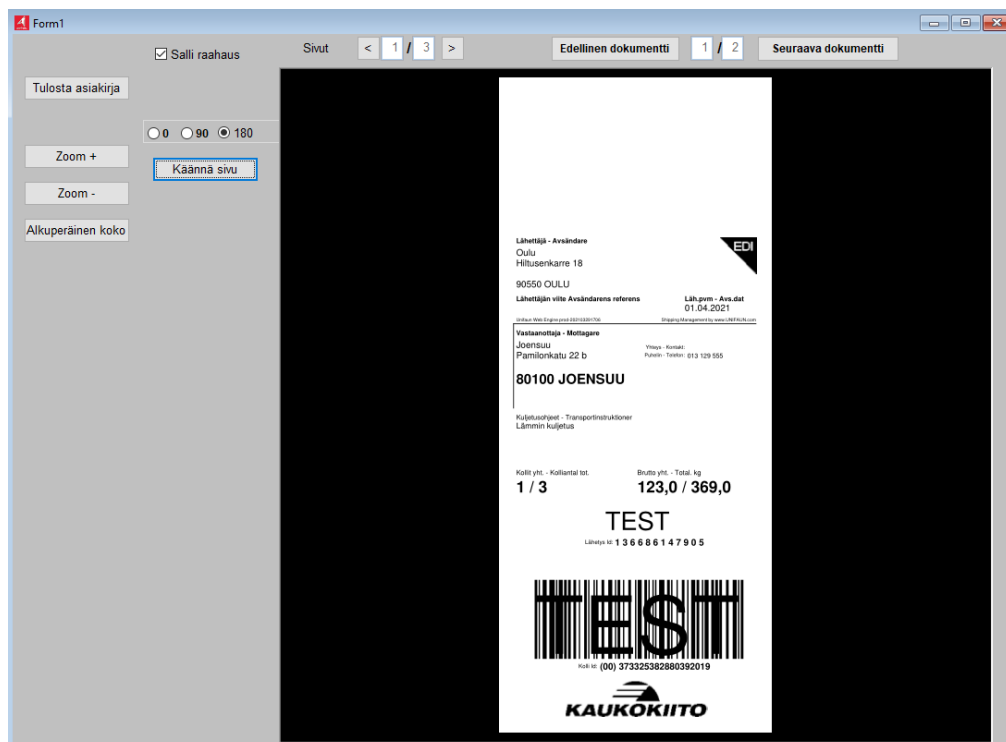
5.3 Tulostusliittymä

Tulostuksen kanssa ilmeni ongelmia palvelimella sijaitsevista PDF-dokumenteista, jotka pitäisi saada tulostettua ohjelman kautta. Ongelmana on se, että ohjelma ja PDF-tiedostot sijaitsevat samalla palvelimella. Käyttäjällä ei ole oikeutta avata tai selata näitä tiedostoja, joten tähän tarvittiin ohjelman kautta pääsy kuljetustilaukseen liittyviin asiakirjoihin.

Käyttäjä ei myöskään saanut tiedostoja paikalliseen ympäristöön, jota kautta olisi tulostuksen voinut suorittaa. Applirent TK:ssa tätä toimintoa ei ole aikaisemmin ollut. Tähän toimintoon tarvittiin ulkopuoliselta toimittajalta lisäosa millä mahdollistettiin tulosteiden esikatselu ja tulostus suoraan ohjelmasta. Tässä vaiheessa ei tiedetty mistä löytyisi toimittaja tarvittavalle lisäosalle.

Tarjolla oli hyvin niukasti tarpeeseen sopivaa lisäosaa tarjolla ja sen löytäminen vei aikaa. Viscom-Softin (WiscomSoft, ei pvm) tarjoama ActiveX-komponentti, jolla tiedoston tulostaminen ja esikatselu on mahdollista käyttöliittymässä ilman erillistä ohjelmaa. Tämä tuo lisäominaisuuksia ohjelmaan tulevaisuudessa.

Kyseisestä komponentista lähdettiin työstämään tulostukselle suunnattua omaa käyttöliittymää. Kuvassa 11 valmis tulostukseen tarkoitettu käyttöliittymä.



Kuva 11. Tulostuskäyttöliittymä

5.3.1 ActiveX PDFViewer-komponentti

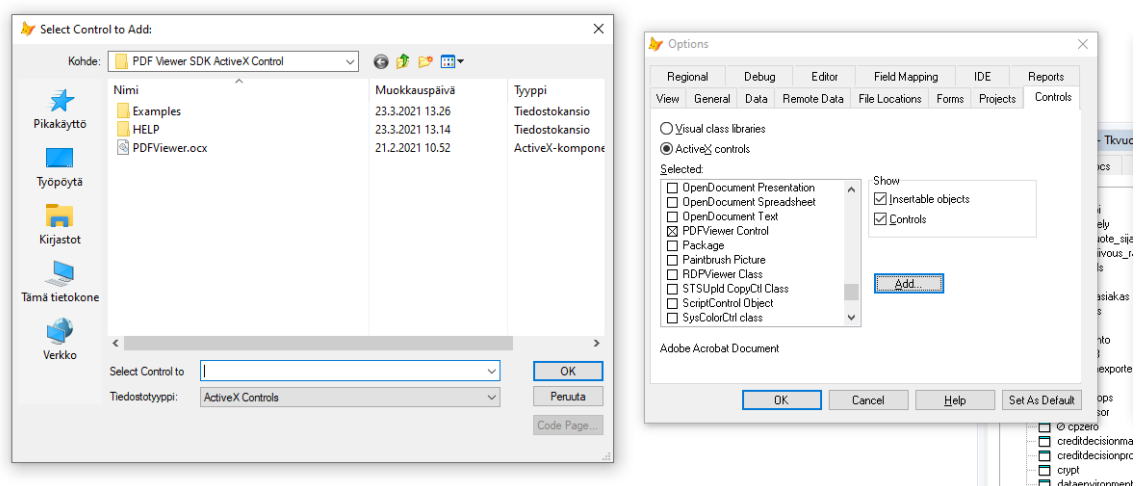
Komponentista on tarjolla vapaasti testattava versio, joka piti sisällään tarvittavat toiminnot mitä tähän työhön on tarve käyttää. Jatkossa tullaan todennäköisesti päivittämään tuetumpaan maksulliseen versioon. FoxPro tarjoaa paljon erilaisia ActiveX-komponentteja, mutta kyseistä komponenttia ei löydy valmiina ohjelmasta. Tämä tuotiin lisäosana järjestelmään.

ViscomSoft tarjoaa hyvät ohjeistukset PDFViewerin asennukseen ja kuinka sitä käytetään. Asennukseen tarvittiin lisäosat ohjelmaan sekä yhden DLL-tiedoston asennus koneeseen. PDFViewer-komponentti lisätään käyttöliittymään, joka toimii PDF-tiedoston esikatseluikkunana. ViscomSoft tuo ladattavan tiedoston yhteydessä esimerkkikäyttöliittymän. Tästä oli joustava muokata ja käyttää tarvittavia toimintoja omaan käyttöliittymään. Toiminnot ovat listattu selkeästi ViscomSoftin tarjoamassa dokumentissa (PDF Viewer SDK ActiveX).

Kyseinen komponentti on asennettava ja otettava käyttöön erikseen. Asennukseen vaadittiin yhden DLL-tiedoston asennus, joka mahdollisti PDFViewerin esikatseluikkunan käytön. Projektin tiedoston juureen oli myös sisällytettävä viisi kappaletta tarvittavia DLL-tiedostoja, jotka mahdollistivat tulostuksen kyseisellä komponentilla.

Asennuksen ja käyttöönoton jälkeen FoxProhon tuli lisätä ActiveX-komponentti "options"-valikosta ja uuden komponentin lisäys tapahtuu "Add"-painikkeella ja valitsemalla oikea tiedosto.

Valitsemalla PDFViewer Control -komponentti listataan ikkunoiden suunnittelu työkaluihin, josta PDFViewer lisätään haluttuun ikkunaan. Kuvassa 12 esitetty uuden ActiveX-komponentin lisääminen.



Kuva 12. Ulkopuolisen ActiveX-komponentin lisääminen

Taulukko 6 näyttää toiminnot mitä PDFViewer-komponentilla voidaan tehdä ja se on tehty pelkään ajatellen kuljetustilauksen tekoa. Muitakin toimintoja on luotu PDF-käyttöliittymään, mutta painikkeet ovat piilotettu tai ne eivät ole tässä käytettävissä. Tällä tavoin voidaan muokata PDF-käyttöliittymää tarpeiden mukaan.

Taulukko 6. PDFViewer-komponentissa käytetyt toiminnot

TOIMINTO	KUVAUS
LOADPDFFILE	Lataa näkymälle pdf tiedoston esikatselun. Annetaan polku mistä tiedosto noudetaan esikatseltavaksi
PDFPRINT	Tulostetaan asiakirja. Annetaan polku mistä tiedosto noudetaan tulostettavaksi
PDFZOOMOUT/IN	Voidaan tarkentaa asiakirjaa
ROTATEANGLE	Voidaan kääntää asiakirjaa haluttuun kulmaan
GOTOPAGE	Voidaan siirtyä halutulle sivulle. Käytetään edellinen/seuraava painikkeen avulla

Kokonaisuudessaan PDFViewer on erinomainen lisäosa ohjelmaan ja se tuo valtavasti lisämahdollisuuksia tulevaisuudessa PDF-tiedostojen käsittelyyn.

5.4 Ylläpito- ja selausliittymä

Ylläpito ja selaus toimivat samalla ikkunalla eri välilehdissä ja ovat kuljetustilauksen päänäkymä. Kun kuljetustilaus avataan, avautuu myös tämä ikkuna. Tämä on luotu Applirent TK:n niin sanottujen määräysten mukaisesti. TK:ssa on tehty suurin osa ikkunoista kyseisellä tyyllillä ja tulevaisuudessa tähän voidaan helposti lisätä välilehtiä.

5.4.1 Ylläpito

Ylläpitovälilehti pitää sisällään tiedot kuljetukseen liittyen: Vastaanottaja-, lähettäjä- sekä rahtiin liittyvät tiedot, kuljetusliikkeiden palvelut/lisäominaisuudet ja tulostusvaihtoehdot. Kyseisellä välilehdellä voidaan tarvittaessa myös poistaa kuljetus.

Poistamistoiminnon käytön kannattavuutta on mietittävä tarkoin. Ilmoitus menee kuljettavalle yritykselle, mutta varmuutta ei ole, saavuttaako tieto kuljettajan. Poiston tulisi tapahtua huomattavasti ennen kuin kuljettava yritys on noutamassa tavaroita. Tämä oli Unifaunin antama huomio, joka vaatii jatkossa tarkempaa tarkastelua.

Tulostus voidaan tehdä myös ylläpitoikkunasta. Tämä vaihtoehto on laitettu lisänä tähän, jos aikaisemmassa vaiheessa oleva tulostusvaihtoehto ei ole käytettävissä tai halutaan tulostaa myöhemmin tarvittavat asiakirjat. Painike avaa tulostusikkunan sen kuljetuksen tiedoilla, joka on valittuna ylläpitoikkunassa.

Unifaun tarjoaa myös kuljetuksen muokkaamistoimintoa, mutta tätäkin pidettiin Unifaunin mielestä harkinnanvaraisena. Unifaun tarjoaa myös erilaisia kuljetustilauksia, jossa muokkaaminen on hyödyllisempää kuin meidän tavassamme toimia. Tähän todettiin, että suurempi ja helpompi tapa on poistaa edellinen kuljetus ja tehdä uusi kuljetus tilalle. Näin saadaan varmuus, että uudet tiedot

pitävät varmasti paikkansa. Tässä on myös hyvä arvioida, poistetaanko kuljetustilaus vai ollaanko yhteydessä suoraan kuljettavaan yritykseen esimerkiksi puhelimitse. Kuvasta 13 näkyy ylläpitoikkuna.

Form1

Ylläpito Selaus

Tilausnumero 36

Lähetäjä Vastaanottaja

Asiakas/Vp Joensuu Marko Mustonen

Osoite Pamilonkatu 22 b Testtie 8 a 5

Postinumero 80100 99999

Postitoimipaikka Joensuu Kuopio

Puhelinro 013 129 555 05000055

Sähköposti Marko.mustonen@testi.fi

Lähetäjän viite Lähetäjän viite Vastaanottajan viite

Tekstiä Lähetäjän viite

Rahti

Kuvaus	Tyyppin kuvaus	Tyyppi	Paino	Määrä
Tervehdys	EUR-lava (EP)	EP	100	1
Tervehdys	EUR-lava (EP)	EP	100	1

Lisäominaisuudet

Tyyppi	Kuvaus
WARM	Lämminkuljetus
TAILLI	Perälautalastaus

Poista kuljetus Tulosta dokumentit

Kuva 13. Kuva ylläpitoikkunasta

5.4.2 Selaus

Selausikkuna listaa kaikki kuljetukset mitä omassa järjestelmässä on muistissa. Tämä toiminto ei listaa kuljetuksia mitkä on tehty Unifaun Onlinen kautta. Listausta voidaan järjestää monella eri tavalla. Listausta voidaan myös suodattaa lähettäjä/vastaanottaja tiedoilla sekä kuljetustilaukseen liittyvillä tiedoilla esim. tilausnumero sekä kuljettava yritys. Näihin tulee varmasti lisäystä asiakkaiden tarpeiden mukaan. TK:ssa on myös mahdollista käyttää hakupaneelia, jossa suodatus mahdollisuuksia saadaan lisää helposti. Tässä vaiheessa siihen ei vielä enempää paneuduttu kuljetustilausten määrän ollessa vähäinen. Kuvasta 14 näkyy selausikkuna.

Tilaus	Palveluid	Palvelu	Lähtetäjä	Vastaanottaja	Lähtetäjä osoite	Vastaanottaja osoite	Customer
2	KKSTD		Jyväskylä	Kuopio	Ahjokatu 24	Kartanonkatu 9	1
3	KKSTD		Joensuu	Kuopio	Pamilonkatu 22 b	Kartanonkatu 9	1
4	KKSTD	Kaukokiito	Joensuu		Pamilonkatu 22 b		1
5	MH57	Matkahuolto					1
6	KKSTD	Kaukokiito	Joensuu	Nurmes	Pamilonkatu 22 b	Teollisuustie 5	1
8	KKSTD	Kaukokiito	Joensuu	Joensuu	Pamilonkatu 22 b	Pamilonkatu 22 b	1
9	KKSTD	Kaukokiito	Jyväskylä	Jyväskylä	Ahjokatu 24	Ahjokatu 24	1
10	KKSTD	Kaukokiito	Joensuu	Jyväskylä	Pamilonkatu 22 b	Ahjokatu 24	1
11	POF1	Posti Oy, Kotima	Joensuu	Jyväskylä	Pamilonkatu 22 b	Ahjokatu 24	1
12	POF1	Itella, Kotimaan	Kuopio	Jyväskylä	Kartanonkatu 9	Ahjokatu 24	1
13	MH57	Matkahuolto	Joensuu	Kuopio	Pamilonkatu 22 b	Kartanonkatu 9	1
14	POF1	Posti Oy, Kotima	Jyväskylä	Joensuu	Ahjokatu 24	Pamilonkatu 22 b	1
15	POF1	Posti Oy, Kotima	Joensuu	Oulu	Pamilonkatu 22 b	Hiltusenkarre 18	1
16	KKSTD	Kaukokiito	Oulu	Joensuu	Hiltusenkarre 18	Pamilonkatu 22 b	1
17	KKSTD	Kaukokiito	Jyväskylä	Joensuu	Ahjokatu 24	Pamilonkatu 22 b	1
18	KKSTD	Kaukokiito	Joensuu	Kuopio	Pamilonkatu 22 b	Kartanonkatu 9	1
19	KKSTD	Kaukokiito	Kuopio	Joensuu	Kartanonkatu 9	Pamilonkatu 22 b	1
20	KKSTD	Kaukokiito	Kuopio	Joensuu	Kartanonkatu 9	Pamilonkatu 22 b	1
21	KKSTD	Kaukokiito	Joensuu	Nurmes	Pamilonkatu 22 b	Teollisuustie 5	1
22	KKSTD	Kaukokiito	Kuopio	Jyväskylä	Kartanonkatu 9	Ahjokatu 24	1
23	KKSTD	Kaukokiito	Joensuu	Nurmes	Pamilonkatu 22 b	Teollisuustie 5	1
24	KKSTD	Kaukokiito	Joensuu	Kuopio	Pamilonkatu 22 b	Kartanonkatu 9	1
26	KKSTD	Kaukokiito	Joensuu	Kuopio	Pamilonkatu 22 b	Kartanonkatu 9	1
27	KKSTD	Kaukokiito	Joensuu	Joensuu	Pamilonkatu 22 b	Pamilonkatu 22 b	1
29	KLGRP	DB SCHENKER	Jyväskylä	Oulu	Ahjokatu 24	Hiltusenkarre 18	1
30	MH57	Matkahuolto	Joensuu	Joensuu	Pamilonkatu 22 b	Pamilonkatu 22 b	1
32	KKSTD	Kaukokiito	Joensuu	Jyväskylä	Pamilonkatu 22 b	Ahjokatu 24	1
33	KKSTD	Kaukokiito	Joensuu	Marko Mustonen	Pamilonkatu 22 b	Testitie 8 a 5	1
34			Jyväskylä		Ahjokatu 24		1
35							1
36							1

Kuva 14. Kuva selausikkunasta

5.5 Virheen käsittely

Työssä eteen tulleiden virheiden käsittely nousi tärkeään rooliin. Virheen käsittelyssä oli mietittävä, kuinka käyttäjälle saadaan mahdollisimman kuvaava tieto virheestä ja kuinka virhe voidaan korjata. Tämä tuotti ongelmia, koska rajapinnasta saatu virhekoodi saattoi olla sellaista tekstiä, josta TK:n käyttäjä ei välttämättä osaa tulkita mitä virheessä tarkoitetaan. Tähän tehtiin aluksi ratkaisu, joka ilmoittaa vain virhekoodin käyttäjälle ja tästä tulisi tulkita kuinka toimia. Asiaan paneudutaan tulevaisuudessa ja yhdessä Unifaunin kanssa mietitään, kuinka jatkossa toimitaan.

Virheenkorjausta käsiteltiin myös kehittäjätasolla ja lokitukset tehtiin kaikista tarvittavista kohdista, jossa oletettiin virheiden tulevan. Lokituksella tarkoitetaan selkokielen tekstin kirjoittamista tekstiedostoon, kun koodia suoritetaan. Käytännössä virheitä voisi tulla vain kutsuja tehdessä, johon emme voi koodillisesti aina vaikuttaa. Tämä lokitetaan joka kerta, kun tilausta tehdään, oli se onnistunut tai virheellinen. Tällä tavoin saadaan kontrolloitua ja vertailtua onnistuneita ja virheellisiä tilauksia.

Optimaalinen tilanne on, että näiltä virheiltä säästytäisiin. Työn edetessä tuli esille tilanteita, joita on otettava jatkossa huomioon. Esimerkiksi tilauksia tehdessä virheellisten ja puutteellisten tietojen kanssa. Tämä oli ongelma, jota emme voineet korjata muulla tavalla, kuin näyttämällä käyttäjälle tai kehittäjälle saapunut virheteksti.

6 JATKOKEHITYS

Jatkokehitysideoita tuli työtä tehdessä valtava määrä. Osa ideoista toteutettiin työtä tehdessä. Ideoita kehitettäväksi ja toteutettavaksi jäi vielä paljon.

Yhtenä kehitysideana nousi esiin kuljetustilaus muualtakin TK:sta kuin kuljetustilausikkunasta. Kuljetuksien tilauksia voisi hyödyntää varastonsiirroissa, sopimusta tehdessä ja huoltoon siirrettäessä. Tämä on yksinkertainen lisäys, joka melko suurella varmuudella tullaan tulevaisuudessa tekemään.

Edellä mainittuun viitaten voitaisiin hyödyntää tuotetietokantaa sekä niihin liittyviä tuotteen tietoja (esimerkiksi paino, pituus ja selite). Tavaraa lähettäessä näitä tietoja voitaisiin automaattisesti käyttää rahtitietona. Tätä olisi tarkoitus lähteä työstämään jatkossa.

Applirent tarjoaa asiakkailleen myös mobiilisovelluksen. Mobiilisovellus ei varsinaisesti kuulu Applirent TK:hon mutta se kuuluu Applirentin tuoteperheeseen. Tämä olisi todella hyödyllinen rajapinta myös siihen ja tullaan tulevaisuudessa liittämään mobiilisovellukseen.

7 POHDINTA

Työssä käsiteltiin rajapinnan liittämistä ERP-järjestelmään. Asetettuihin tavoitteisiin päästiin osittain. Tavoitteena oli saada toimiva integraatio tuotantoon asti mutta tuotannon vaihe jäi vajaavaiseksi. Tuote on tuotantokuntoinen, mutta se toimisi vain rajoitetusti. Tilaajan taholta integraatio jätettiin kehitysvaiheeseen, koska tuotantoon haluttiin täysin toimiva integraatio.

Työssä tuli esille useita ratkaistavia kohteita ja ongelmia, joihin paneuduttiin. Kyseisten asioiden tunnistaminen oli suunnitteluvaiheessa vaikeaa. Ratkaisut löydettiin ongelmiin, joihin pystyttiin itse vaikuttamaan. Jotkin ongelmat olivat riippumattomia meidän laitteistoistamme ja näitä ratkotaan tulevaisuudessa.

Yhtenä suurena ongelmana oli PDF- tiedostojen lukeminen ja tulostaminen. Tämän ongelman ratkettua saatiin lisäarvoa myös Applirent TK-järjestelmään. Tätä voidaan jatkossa käyttää myös muualla järjestelmässä kuin pelkästään kuljetustilauksissa.

Opinnäyteyössä tähdättiin Applirentin asiakkaiden jokapäiväisen työn helpottamiseen ja lisäarvon tuomiseen Applirent TK-järjestelmään. Arvioita siitä, kuinka ohjelma toimii ja onko järjestelmästä ollut hyötyä asiakkaille tai lisäarvoa Applirentille, saadaan tulevaisuudessa.

8 LÄHDELUETTELO

- Avraham, S. B. (5. 10 2017). *What is REST*. Haettu 29. Huhtikuu 2021 osoitteesta Medium:
<https://medium.com/extend/what-is-rest-a-simple-explanation-for-beginners-part-1-introduction-b4a072f8740f>
- Microsoft Visual FoxPro 9.0*. (29. 08 2016). Haettu Maaliskuu 2021 osoitteesta Microsoft:
[https://docs.microsoft.com/en-us/previous-versions/visualstudio/foxpro/mt490117\(v=msdn.10\)?redirectedfrom=MSDN](https://docs.microsoft.com/en-us/previous-versions/visualstudio/foxpro/mt490117(v=msdn.10)?redirectedfrom=MSDN)
- PDF Viewer SDK ActiveX*. (ei pvm). Haettu 15. Huhtikuu 2021 osoitteesta ViscomSoft:
<http://www.viscomsoft.com/onlinehelp/pdfviewer/Introduction.html>
- Rykov, D. (2012). *THREE DIMENSIONAL INTERACTIVE DENTAL CHARTING*. Haettu 01. Toukokuu 2021 osoitteesta Theseus:
https://www.theseus.fi/bitstream/handle/10024/42783/Rykov_Dmitry.pdf?sequence=1&isAllowed=y
- Strahl, R. (ei pvm). *West Wind*. Haettu 05. 05 2021 osoitteesta Web Connection: <https://webconnection.westwind.com/>
- Unifaun kuljetusliikkeet*. (ei pvm). Haettu 01. Maaliskuu 2021 osoitteesta Unifaun:
<https://www.unifaun.com/fi/kuljetusliikkeet/kuljetusliikehakemisto/>
- Unifaun REST Api Documentation*. (ei pvm). Haettu 01. Maaliskuu 2021 osoitteesta Unifaun:
<https://www.unifaunonline.se/rs-docs/>
- Unifaun shipments*. (15. 03 2020). Haettu Maaliskuu 2021 osoitteesta Unifaun shipments:
<https://www.unifaunonline.se/rs-docs/##shipments>
- Vartiainen, S. (7. Syyskuu 2020). *Theseus*. Haettu 15. Huhtikuu 2021 osoitteesta Avoin web-rajapinta sensoridatalle:
https://www.theseus.fi/bitstream/handle/10024/345462/Vartiainen_Samu.pdf?sequence=2&isAllowed=y
- Visual FoxPro System Capacities*. (07. 09 2007). Haettu Helmikuu 2021 osoitteesta Microsoft:
[https://docs.microsoft.com/en-us/previous-versions/visualstudio/foxpro/3kfd3hw9\(v=vs.80\)?redirectedfrom=MSDN](https://docs.microsoft.com/en-us/previous-versions/visualstudio/foxpro/3kfd3hw9(v=vs.80)?redirectedfrom=MSDN)
- WiscomSoft*. (ei pvm). Haettu 15. Huhtikuu 2021 osoitteesta WiscomSoft:
<http://www.viscomsoft.com/products/pdfviewer>