

# RPA-PROSESSI SAP:N ERÄHALLINTADATAN KORJAAMISEKSI



Ammattikorkeakoulututkinnon opinnäytetyö

Tietojenkäsittelyn koulutus, Hämeenlinnan korkeakoulukeskus  
kevät, 2021

Mika Björkholm

## TIIVISTELMÄ

Tämän opinnäytetyön tavoitteena oli suunnitella ja ohjelmoida ohjelmistorobotiikka-automaatio Konecranes Oy:n toiminnanohjausjärjestelmässä havaittuun vikaan. Vika toiminnanohjausjärjestelmässä liittyi puuttuvaan tietoon yrityksen erähallintadatassa. Korjaava työvaihe suoritettiin manuaalisesti syöttämällä toimitusnumero selainpohjaiseen järjestelmään, johon oli rakennettu toiminnanohjausjärjestelmän oma automaatio korjaamaan puuttuva data. Yritys etsi vaihtoehtoja SAP:n toiminnallisuuksista, mutta päätyi ohjelmistorobotiikkaan ihmisen manuaalisesti tekemän työprosessin automatisoinniksi.

Suunnittelusta ja ohjelmoinnista tehtiin yksityiskohtainen dokumentaatio yrityksen omaan käyttöön. Ohjelmistorobotiikka-alustana toimi yrityksen käyttämä UiPath, joka on myös yksi maailman suurimmista ohjelmistorobotiikan palveluntarjoajista. Ohjelmointikielenä oli Visual Basic. Opinnäytetyön käytännön osuudessa käytettiin yrityksen dokumentaatiopohjia sekä ohjelmistorobotiikkatyökaluja.

Teorian tietolähdepohjana käytettiin pääosin UiPath-dokumenttiportaalia, missä ohjelmointialustan funktioiden ja metodien selitykset oli kerrottu. Ongelmatilanteiden selvittämiseen käytettiin UiPath:n keskustelupalstoja, mistä löytyi laaja-alaisesti muiden UiPath-käyttäjien ongelmia sekä ratkaisuja. Ohjelmointialustaan tutustuminen suoritettiin verkkokoulussa, UiPath Academyssä, missä oli mahdollista opiskella ja suorittaa sertifikaatteja UiPath:n käyttöliittymän toimintoihin.

Tämän toiminnallisen opinnäytetyön tuloksena syntyi Konecranes Oy:n tilaama automaatio. Automaation rakentaminen tutustutti opinnäytetyöntekijän ohjelmistorobotiikkaan, joka on tällä hetkellä yleistynyt trendi yrityksissä.

Avainsanat Ohjelmistorobotiikka, RPA, UiPath, SAP, FIORI,

Sivut 38 sivua ja liitteitä 1 sivu.

---

Author       Mika Björkholm

Year 2021

Subject       Batch data correction for auto goods receipt

Supervisors  Lasse Seppänen

---

#### ABSTRACT

The aim of this thesis was to design and develop robotic process automation for a bug in Konecranes Oy enterprise resource planning (ERP) system SAP. The bug is related to missing batch data in the goods receipts process in SAP that the company is using. Workflow that is made by a human was to set manually an inbound delivery number in a browser based system related to company ERP. The system has built in automation for fixing a bug in SAP. Company has decided to use robotic process automation (RPA) to automate manually done workflow.

Detailed documentation of the design and programming was made for the company's own use and automation was used to replace the manmade manual work process. The platform for the robotic process automation development is one of the world's largest RPA service providers called UiPath. It is used by the Konecranes Oy and the programming language was Visual Basic. The company's documentation templates and RPA tools were used in the practical part of the thesis.

Theoretical part is mainly based on UiPath-documents web portal that contains explanations of the functions and methods for programming platform. Moreover, UiPath forums, where other users have discussed related problems and solutions, were used to solve problems. Introduction to the programming platform was conducted at an online school UiPath Academy where it was possible to study and complete certifications for UiPath user interface functions and roles. The result of this functional thesis is an automated process ordered by Konecranes. The implementation of this process allowed the author of this thesis to gain expertise in robotic process automation that is a rapidly rising trend in companies.

Keywords Robotic Process Automation, RPA, UiPath, SAP, FIORI

Pages 38 pages and appendices 1 page

## Sanasto

RPA	Robotic Process Automation (ohjelmistorobotiikka)
CoE	A Center of Excellence (yrityksen organisaatioyksikkö, joka valvoo ja koordinoi RPA-kehityksiä)
SAP	Systeme, Anwendungen und Produkte in der Datenverarbeitung) (toiminnanohjausjärjestelmän nimi)
ERP	Enterprise Resource Planning (liiketoimintaprosessien hallintaohjelmisto)
Transaktio-koodi	Koodi, jolla ajetaan SAP:n funktioita tai ohjelmia
FIORI	SAP:n selainpohjainen yksinkertaistetumpi käyttöliittymä. FIORI tarjoaa joukon sovelluksia, joita käytetään yrityksen tavallisissa liiketoiminnoissa, kuten työnhyväksynnöissä, taloussovelluksissa, laskentasovelluksissa ja erilaiset itsepalvelusovelluksissa
UiPath	Ohjelmistorobotiikan alusta
Dispatcher	RPA-prosessin vaihe, jossa kerätään data työjonon luomiseksi
Performer	RPA-prosessin vaihe, jossa prosessoidaan työjono
AS-IS	Nykyinen työnkulku kuvattuna step-by-step
TO-BE	Robotin työnkulku kuvattuna step-by-step
BPMN	Business Process Modelling Notation (liiketoimintaprosessin kuvaaminen graafisesti korkealla tasolla)
Business Exception	Työnkulussa tapahtuva robotille asetettujen säännöistä poikkeaminen
Application Exception	Ohjelmistossa (kuten SAP:ssa) tapahtuva odottamaton virhe (esimerkiksi palvelimen verkkoyhteys katkeaa)

## Sisällys

1	Johdanto .....	1
2	Opinnäytetyöntilaaja .....	2
3	Toimintaympäristö .....	3
3.1	SAP .....	3
3.2	SAP FIORI .....	5
4	Ohjelmistorobotiikka (RPA) .....	6
4.1	UiPath .....	8
4.2	Modularisointi .....	9
4.3	UiPath Orchestrator .....	9
4.4	KC-Framework, Konecranes Oy:n RPA-rakenne .....	10
5	RPA - Dokumentaatio .....	11
5.1	PDD-Dokumentaatio .....	11
5.1.1	AS-IS .....	11
5.1.2	TO-BE .....	12
5.2	SDD-dokumentaatio .....	13
6	RPA-ohjelmointi erähallintadatan korjaamiseksi .....	14
6.1	Erähallintadata ja sen ongelmat .....	14
6.2	UiPathin alustaminen robotille .....	15
6.3	Dispatcher-ohjelmointi SAP:ssa .....	16
6.4	FIORIn avaaminen selaimessa .....	19
6.5	Performer-ohjelmointi FIORI:ssa .....	24
7	Johtopäätökset ja pohdinta .....	27
8	Yhteenveto .....	28
	Lähteet .....	29

Kuva 1, SAP:n transaktio VL06i, jota käytetään robotin työnkulussa toimitusnumeroiden hakuun .....	4
Kuva 2. FIORIn etusivun käyttäjäkohtainen näkymä .....	5
Kuva 3. Työtehtävän automatisoinnin potentiaalikuvaaja (Asatiani & Penttinen s.6, 2016). .....	7
Kuva 4. KC-Framework, ohjelman tilat (States) ja niiden relaatiot .....	10
Kuva 5. AS-IS työnkulku kuvattuna .....	12
Kuva 6. Korkean tason BPMN-mallinnus automaatiosta .....	14
Kuva 7. Työnkulku, jossa annetaan SAP:n parametrit käyttämällä varianttia .....	16
Kuva 8. Väliaikaisen tiedostosijainnin tallentaminen muuttujaan.....	17
Kuva 9. Dispatcher-prosessi.....	18
Kuva 10. Robotti saa Orchestratorista asetti arvona selaimen nimen, jonka perusteella se valitsee työnkulun selaimen valitsemiseksi prosessissa .....	20
Kuva 11. Robotin työnkulku, kun se saa asetti-arvona "chrome" .....	20
Kuva 12. Robotti avaa selaimen ja kirjautuu FIORlin .....	21
Kuva 13. Orchestratorista haettu salasana kryptataan kirjautuessa FIORlin.....	22
Kuva 14. Robotti tarkastaa onko selain auki. Jos selain ei ole auki robotti avaa sen. ....	23
Kuva 15. Robotti vie toimitusnumeron FIORlin.....	25
Kuva 16. Toimitusnumero on hyväksytty FIORlin Goods Receipt -näkymän avautuessa robotille. ....	26
Taulukko 1 - Kuvassa verrataan ihmisen ja robotin työnkulkua ajallisesti.....	12

## Liitteet

Liite 1	Aineistonhallintasuunnitelma
---------	------------------------------

# 1 Johdanto

Nykypäivän dataohjautuvassa yritysmaailmassa ehjä data on liiketoimen ja sisäisen tiedon jakamisen kannalta tärkeä osa prosessiketjua. Lopputoimitukselle tärkeät tiedot voivat muuttua prosessin eri vaiheissa tai jopa puuttua kokonaan. Esimerkiksi yksittäisen materiaalin tai materiaalikuvauksen muuttunut tuotetieto vaikuttaa koko prosessin sujuvaan toimivuuteen aina tilauksen tekemisestä lopputuotteen toimitukseen.

Tämän toiminnallisen opinnäytetyön lähtökohtana toimi Konecranes Finland Oy:n toiminnanohjausjärjestelmässä havaittu vika materiaalin määrittämisprosessin järjestelmässä. Vika ilmenee, kun materiaalin määrittämisprosessin järjestelmää käytetään yhdessä erähallintadatan kanssa. Järjestelmävian korjaaminen SAP:ssa tarkoittaisi muutosta tuotannonohjausjärjestelmän perustoiminnoissa. Se olisi huomattavan riskialtista ja aiheuttaisi suuren määrän testaamista jokaisen SAP-järjestelmäpäivityksen yhteydessä. SAP:n FIORI on selainpohjainen yksinkertaistempi käyttöliittymä, johon Konecranesin SAP-osasto on kehittänyt automaation virheellisen erähallintadatan korjaamiseksi. Datan syöttäminen FIORIn automaatioon on toistuva ihmisen tekemä työtehtävä ja yritys on päättänyt tehdä ohjelmistorobotiikkaa hyödyntäen ratkaisun työnkulun automatisoinniksi.

Tämän opinnäytetyön tarkoituksena oli suunnitella ja ohjelmoida ohjelmistorobotti suorittamaan SAP:n BATCH-datan korjausprosessi käyttämällä UiPath-nimistä ohjelmistorobotiikka-alustaa. Ohjelman suunnittelussa käytiin yksityiskohtaisesti läpi työprosessin nykytilan kuvaus asiakkaan näkökulmasta ja opinnäytetyöllä oli tarkoitus korvata ihmisen tekemä työnkulku. Työn kulusta tehtiin graafinen esitys liiketoimintaprosessien määrittelemiseksi ja prosessiin liittyvät yksityiskohdat dokumentoitiin yrityksen standardien mukaisesti. Opinnäytetyössä asiakkaana toimi Konecranes Finland Oy:n Global Parts, Data & Processes -liiketoimintayksikkö.

Opinnäytetyössä etsittiin ratkaisuja seuraaviin kysymyksiin:

- Miten robotti käsittelee ongelmatilanteet FIORI:ssa?
- Miten robotti reagoi muutoksiin FIORI:ssa?
- Miten optimoidaan robotin tehokkuus eri prosessivaiheissa?
- Syntyykö automaatioprosessissa muuta hyödyllistä dataa, jota voidaan välittää asiakkaalle?

## 2 Opinnäytetyön tilaaja

Tämän opinnäytetyöntilaaja Konecranes Oy on maailman johtaviin nosturilaittevalmistajiin kuuluva yritys, jonka toimenkuva on asiakasyritysten toimintaa tehostavien nostoratkaisuiden ja huoltopalveluiden toimitus kaikille nosturimerkeille. Yrityksen liikevaihto oli vuonna 2020 yli 3 miljardia euroa.

Konecranes Finland Oy, Global Parts, Data & Processes on yrityksen oma organisaatioyksikkö, jonka vastuulla on globaalin varaosamyynnin tuotetiedon ylläpito ja prosessien kehitys. Se vastaa esimerkiksi kansainvälisten myynti- ja ostoryhmien tuotetiedosta sekä uusien tuotenimikkeiden luonnista yrityksen omaan verkkokauppaan.

Eri liiketoimintayksiköiden alaisuuteen on perustettu ohjelmistorobotiikkaosastoja, RPA-hubeja. Ne vastaavat osaltaan liiketoimintayksiköissä tapahtuvista ohjelmistorobottikehityksistä (Robotic Process Automation, RPA). Kehitysten ylläpidosta vastaa yksikön RPA-spesialisti. RPA-spesialistin toimenkuvaan kuuluu uusien RPA-kehitysten suunnittelu ja ohjelmointi, vanhojen kehitysten muokkaukset sekä robottien ylläpito ja ohjelmistovianetsintä. Yrityksessä toimii ohjelmistorobotiikkayksikkö RPA-Center of Excellence. Sen tehtävänä päättää yrityksen ohjelmistorobotiikan uusista kehityksistä ja toimia tukena liiketoimintayksiköiden omille RPA-hubeille.

### 3 Toimintaympäristö

Opinnäytetyössä robotilla käytetty päätoiminen toimintaympäristö oli SAP:n graafinen käyttöliittymä ja selainpohjainen SAP:n Fiori. Tämän luvun tarkoituksena on avata SAP:n toimintaperiaatteita yleisesti ja opinnäytetyön kannalta merkittävässä toiminnoissa toiminnanohjausjärjestelmässä.

#### 3.1 SAP

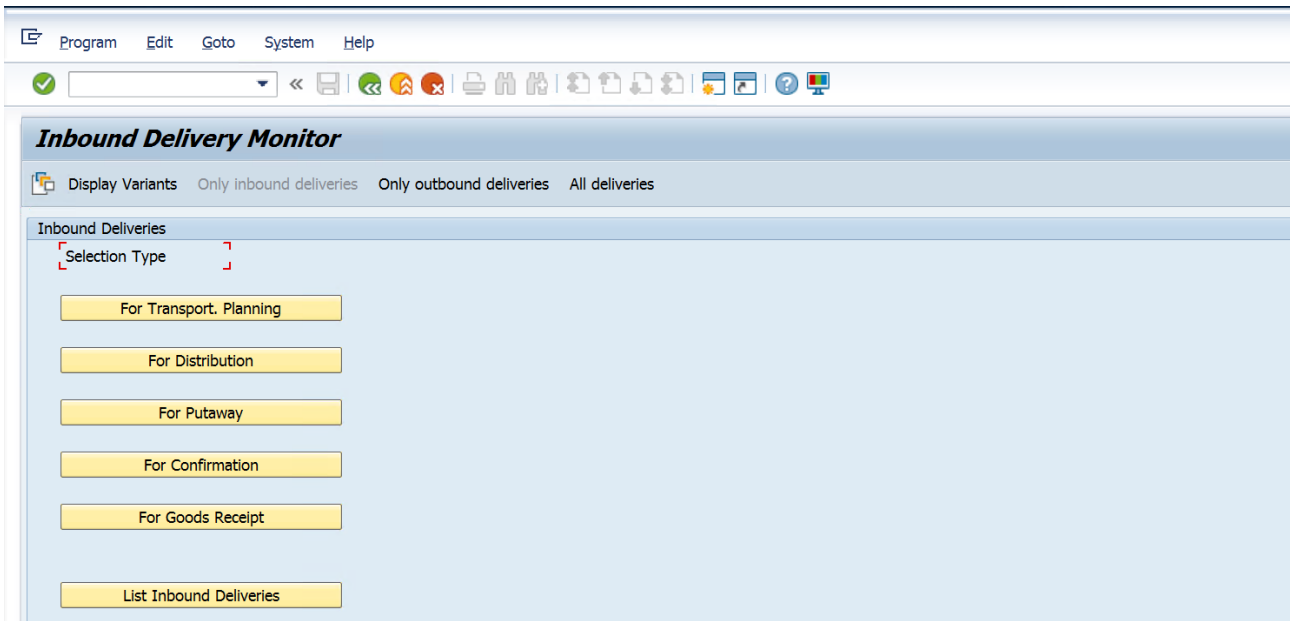
SAP on moniulotteinen toiminnanohjausohjelmisto, jolla ohjataan yrityksen liiketoiminnan osia. Yrityksen useat eri organisaatioyksiköt on liitetty SAP:n avulla toisiinsa. Keskitetyn järjestelmän ansiosta tiedon ja datan hallinta on mahdollista reaaliaikaisesti eri osastojen välillä.

SAP sisältää useita toimintoihin rakennettuja funktioita ja ohjelmia, joita voidaan suorittaa SAP:n graafisesta käyttöliittymästä käyttämällä niille tarkoitettuja transaktiokodeja. Loppukäyttäjille on annettu SAP-oikeuksien myötä roolit SAP-ympäristöön, jolla voidaan rajata pääsy vain tiettyihin transaktioihin. Näin hallinnoidaan käyttäjän toimintaa järjestelmässä.

Käyttäjän on myös mahdollista luoda käyttämiinsä transaktioihin haluamiaan ulkoasuja ja variantteja. Ulkoasuilla voidaan esimerkiksi muokata monisarakkeista näkymää ja poistaa sieltä tietoa, joka on käyttäjälle tarpeetonta. Varianteilla käyttäjän on esimerkiksi mahdollista rakentaa valmiita esitetyjä kenttiä, joilla tietoa haetaan transaktioista.

SAP:n perustoimintoihin kuten tilausprosessiin, myyntilauksen käsittelyyn, tuotannon ohjaukseen sekä varaston vaiheisiin keräilyssä ja pakkauksessa saatiin tätä opinnäytetyötä tehdessä kattava perehdytys ja käytännön kokemus (kuva 1.). Perehdytyksen avulla oli mahdollista toistaa ihmisen tekemä prosessi ja rakentaa testitapauksia RPA-automaation testaamista varten. Konecranes Oy:n SAP-järjestelmää ylläpitää ja kehittää yrityksen oma SAP-organisaatioyksikkö, jonka vastuulla on toiminnanohjausjärjestelmän ylläpito sekä yrityksen sisäiset SAP-kehitykset. Yritys käyttää myös SAP-konsultteja, joille ulkoistetaan kehitykseen ja ylläpitoon liittyviä tehtäviä. (Using SAP ERP, Schulz, 2017).

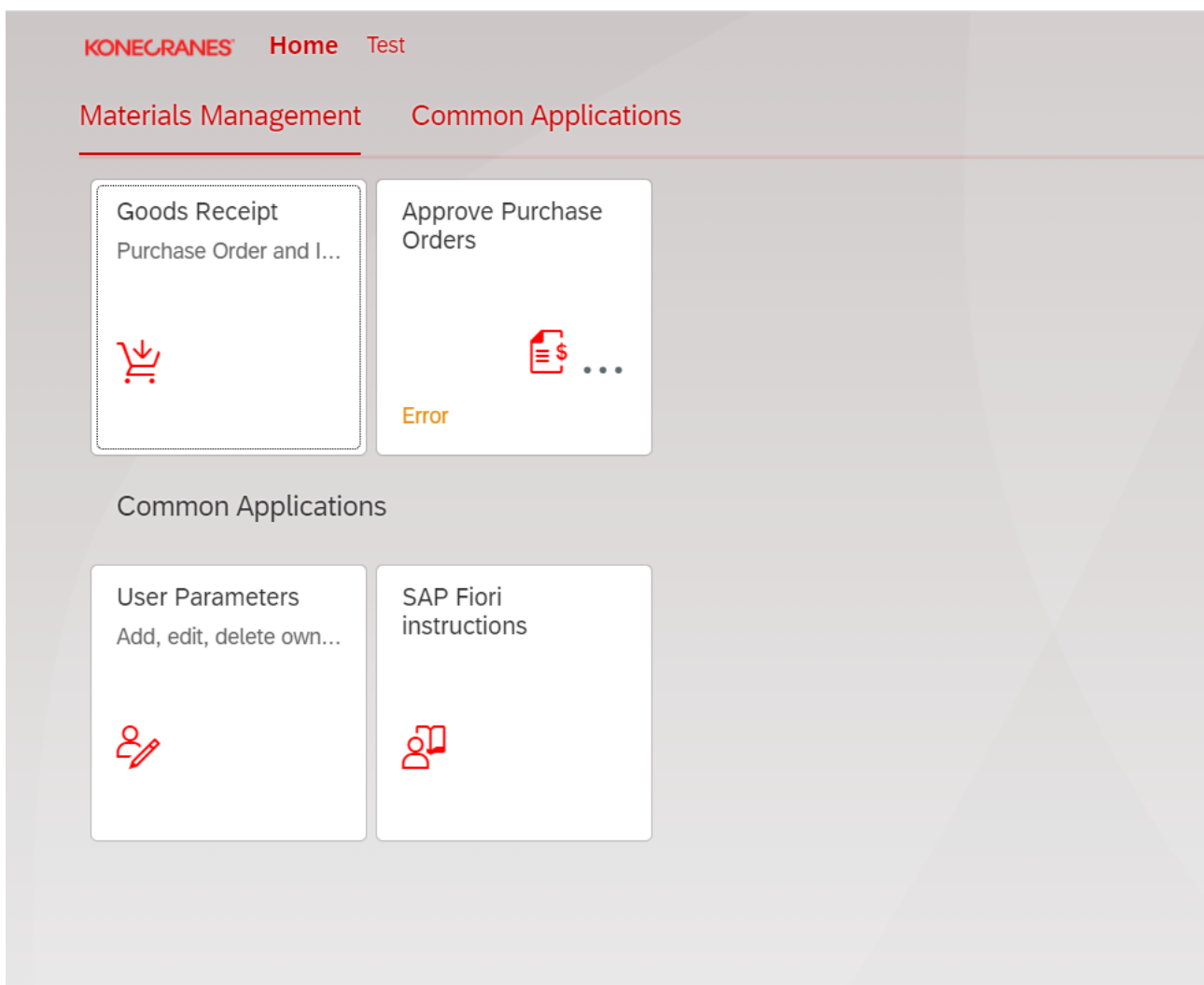
Kuva 1, SAP:n transaktio VL06i, jota käytetään robotin työnkulussa toimitusnumeroiden hakuun



## 3.2 SAP FIORI

SAP:n selainpohjainen FIORI on yksinkertaistetumpi käyttöliittymä (Kuva2.) SAP:n yleisimmin käytettyihin toimintoihin, kuten tapahtumien kirjaukseen ja järjestelmän tietojen katseluun. FIORIa ei ole kuitenkaan tarkoitettu korvaamaan SAP:n graafista käyttöliittymää vaan ainoastaan helpottaa tiettyjen SAP-toimintojen ja -tietojen saatavuutta ja käytettävyyttä. FIORI parantaa SAP:n käytettävyyttä huomattavasti esimerkiksi mobiililaitteille suunnatun käyttöliittymän ansiosta ja sitä käytetään Konecranes Oy:n tuotanto-osastoilla jatkuvasti tuotannolle oleellisten tietojen keräämiseksi.

Kuva 2. FIORIn etusivun käyttäjäkohtainen näkymä

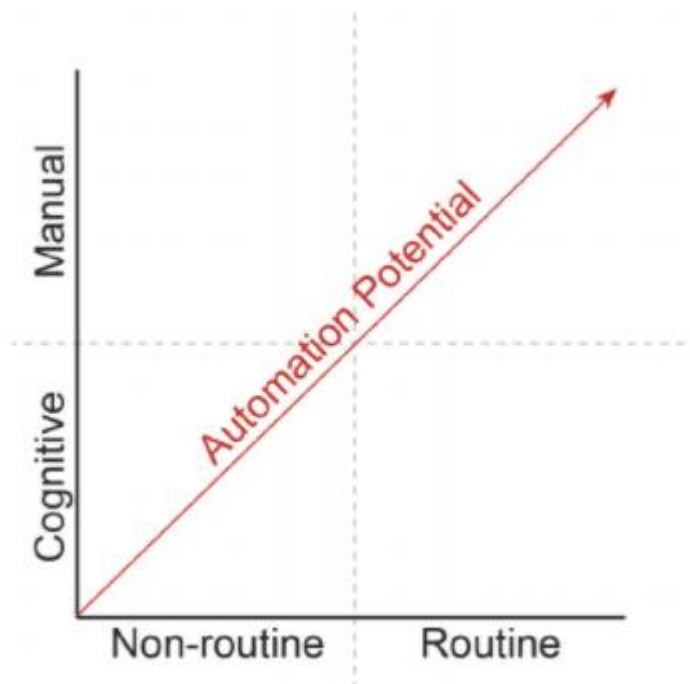


## 4 Ohjelmistorobotiikka (RPA)

Ohjelmistorobotiikka tarjoaa ratkaisuja yritysten tarpeisiin automatisoida toistuvia liiketoiminnan työtehtäviä. Käytännössä robotilla voidaan automatisoida joko yksinkertainen tai hyvin monimutkainen työnkulku. Verrattuna ihmisen vastaavaan työsuoritukseen robotin etu on nopeus ja tarkkuus. Tästä syntyy suuri hyöty yritykselle. Van der Aalst, Bichler ja Heinzl (2018) ovat tutkineet automaation tarpeellisuutta sekä työtehtävien kannattavaa jakoa robottien ja ihmisten välillä yrityksen näkökulmasta. Usein ohjelmistorobottien ajatellaan korvaavan ihmisten tekemät työt ja aiheuttavan irtisanomisia työpaikoilla. Vaikka ohjelmistorobotit suorittavatkin tarkkaa ja väsymätöntä työtä, monesti niiden tarkoitus on helpottaa ihmisen tekemän toistuvan työn kuormitusta. Kun roboti tekee toistuvat työtehtävät, vapautuu yrityksen ihmisresursseja esimerkiksi päätöksenkäyttöä vaativiin työtehtäviin. Tekoälyn sekä koneoppimisen myötä myös Konecranes Oy on alkanut tutkia mahdollisuuksia hyperautomaation, tekoälyn ja ohjelmistorobotiikan työkalujen yhdistämiseen.

Kun tarkastellaan työtehtävien automatisointia ja sopivuutta ohjelmistorobotiikalle on tärkeää määrittellä, saadanko kaikki prosessin vaiheet kuvatuiksi tietyllä tarkkuudella. Tässä on otettava useita tekijöitä huomioon. Tärkeintä on manuaalisten ja rutiininomaisten tai ei rutiininomaisten ja kognitiivista ajattelua vaativien työtehtävien kuvaaminen yksityiskohtaisesti (Kuva3.). Lisäksi on tärkeää identifioida ne ihmisen tekemät rutiininomaiset työvaiheet, joita ei monen aikaisemman toiston takia usein huomata tekevän päivittäisissä työtehtävissä.

Kuva 3. Työtehtävän automatisoinnin potentiaalikuvaaja (Asatiani & Penttinen s.6, 2016).



UiPath:n suurimpiin kilpailijoihin kuuluvat Automation Anywhere ja Blue Prism ovat myös pitäneet oman paikkansa kärkikaartissa ohjelmistorobotiikan markkinoilla. Ohjelmistorobotiikka toimialana on suhteellisen nuori. Edellä mainittujen yritysten toiminta on alkanut vasta 2000-luvun alussa Blue Prismin ensimmäisillä tuotteilla. Tällä hetkellä räjähdysmäisessä kasvussa oleva ala tarjoaa nykyään jatkuvasti uusia teknologioita, joiden avulla yritykset pääsevät rakentamaan ohjelmistorobottiarmeijaa liiketoiminnan tehostamiseksi. Yritysten valikoidessa palvelun tarjoajaa tulevatkin ohjelmistojen teknisten toteutusten määrittelyt ja automaatiotyökalujen käyttöönoton helppous usein ensimmäisinä valintakriteereinä. Tässäkin UiPath tarjoaa monipuolisen käyttöliittymän ohjelmistorobotiikan työkulkujen automatisointiin ja jatkuvan järjestelmäkehityksen. (AiMultiple 2021).

## 4.1 UiPath

UiPath ohjelmistorobotiikka-alusta tarjoaa ratkaisuja yritysten tarpeisiin automatisoida toistuvia liiketoiminnan työtehtäviä. Se on monipuolinen ohjelmointiympäristö, johon on rakennettu työkaluja erilaisten ohjelmointia helpottavien aktiviteettien muodossa. Aktiviteetteja on mahdollista vetää ja/tai pudottaa ohjelman sisälle sekä käyttää niitä funktiona. Ohjelman sekvenssit, eli jaksottaiset ohjelmiston osat koostuvat UiPath-aktiviteeteista, jotka ovat funktioita tai pieniä ohjelmia UiPath:n sisällä. Ohjelmoinnissa pyritään modularisoimaan jokaisen yksittäisen ohjelman aktiviteeteista koostuva sekvenssi. Mahdollisuus liikutella valmiita moduuleita yksinkertaisesti vetämällä niitä sekvenssien sisälle helpottaa kehitystyötä. Yleensä tietoa välitetään argumenteilla moduulien välisesti ja virheellisen tiedon ilmentyessä ohjelma kertoo moduulin nimen vian paikantamiseksi. Argumentit eli alustetut muuttujat mahdollistavat tiedon muokkaamisen moduuleissa. Ulkoisilla muuttujien arvoilla, kuten esimerkiksi salasanat tai internet-osoitteet voidaan lähettää Orchestratorista suoraan robotin ohjelmalle argumenttien avulla. (<https://docs.uipath.com/>). Vianetsintään saattaa myös usein löytyä vastaus UiPath:n keskustelupalstoilta eli foorumeilta. Siellä muut UiPath-käyttäjät vaihtavat tietoa ongelmien ratkaisemiseksi (<https://forum.uipath.com/c/build/activities/168> ).

UiPath:lle ominaisiin tekniikoihin kuuluu myös monimutkaisten sääntötaulukoiden rakentaminen Microsoft Exceliin robotin hallitsemiseksi. Excel-tiedostoilla voidaan ohjata robottia loppukäyttäjän toimesta koskematta robotin ohjelmakoodiin. Taulussa voidaan säilyttää yksinkertaista tietoa, kuten asiakasnumeroita, toimituspaikkoja sekä yksinkertaisesti sääntörivejä, jotka aktivoidaan loppukäyttäjän toimesta ”kyllä” tai ”ei” tekstillä. Robotti lukee sääntötaulun ja tekee sille annettujen sääntöjen mukaan töitä. Tässä opinnäytetyössä tutustuttiin tekniikkaan ja funktioihin, mutta varsinainen ohjelmointityö ei tarvinnut erillistä sääntö-Exceliä (<https://docs.microsoft.com/en-us/dotnet/visual-basic/> ).

UiPath:ssa on myös työkalu tiedonharavointia varten. Käytännössä tiedonharavointi tarkoittaa tietojen lukemista käyttöliittymästä tai websivulta. Se on yksi tärkeä työkalu tiedon keräämiseksi tehokkaasti, koska sillä saadaan tekstimuotoista tietoa tallennettua suoraan datatauluun, jota on mahdollista käyttää ohjelmoinnin osana.

UiPath tarjoaa myös työkaluja enenevässä määrin, joihin on rakennettu tekoälyä ohjelmistorobotiikan tueksi. Koneoppimista ja liiketoiminnan analytiikkaa sekoittamalla UiPath on päässyt uudenlaisille automaation osa-alueille, joita myös Konecranes Oy:ssä on aloitettu tutkimaan.

## **4.2 Modularisointi**

Ohjelmistojen rakenne koostuu sekvensseistä, joista voidaan rakentaa yksittäisiä ohjelmanosia eli moduuleja. Moduuleista voidaan koostaa ohjelmistorakennevuokaavio, joka mahdollistaa esimerkiksi eri työkulkujen päätöksenteon robotille. Käytännössä tämä tarkoittaa kyllä/ei - polkujen rakentamista. Jos toinen ehto ei täyty, robotti seuraa toista polkua päästäkseen sille toivottuun tavoitteeseen. Kun moduulit on rakennettu ohjelman alussa, niiden käyttäminen uudestaan on tehokasta. Esimerkiksi skriptin suorittaminen on usein toistuva prosessi robotille, joten tämän prosessin moduulia voidaan käyttää saman ohjelman sisällä useita kertoja. Moduulit saavat argumentteina käytettävät arvot ja niitä voidaan liikuttaa ohjelmiston eri tasoilla.

## **4.3 UiPath Orchestrator**

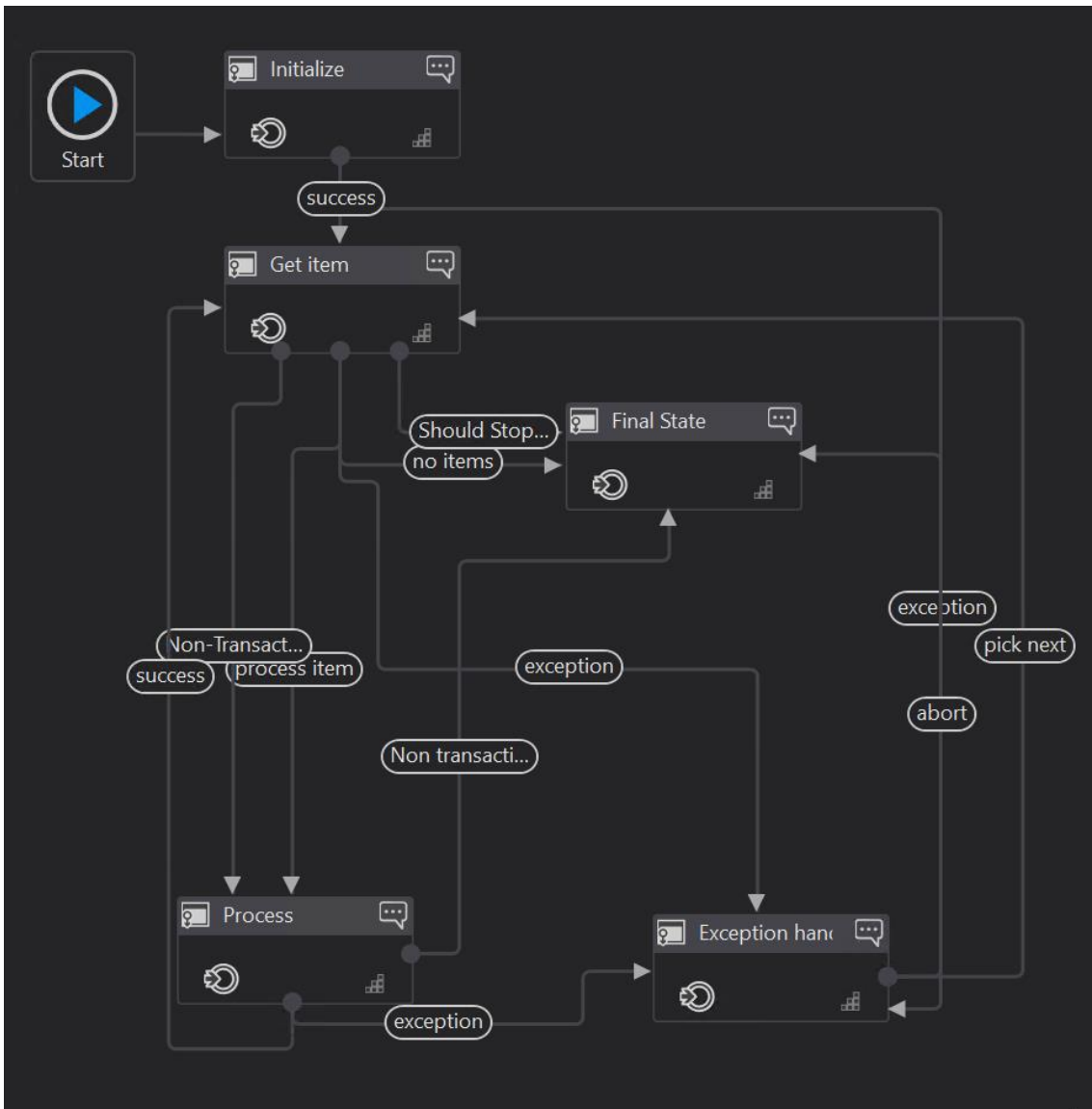
UiPath Orchestrator on selainpohjainen toimintaympäristö robottien monitoroimiselle ja hallinnoinnille. Roboteille voidaan luoda Orchestratoriin työjono, josta robotti yksitellen hakee esimerkiksi prosessoitavan nimikkeen ja antaa ilmoituksen työn lopputuloksesta. Orchestratorissa voidaan myös monitoroida robottien prosesseja sekä ylläpitää robottien eri automaatioissa käytettäviä arvoja. Ohjelmistorobottien arvojen muutokset ovat mahdollisia muuttujien avulla ilman tarvetta muuttaa ohjelmakoodia. Robotin käyttämien sekä jatkuvasti vaihtuvien järjestelmäsälasanojen hallinta on tehokasta ja nopeaa Orchestratorissa ylläpidettävien jaettujen muuttujien eli asettien ansiosta.

Automaation ajastettu toiminta on mahdollista Orchestratoriin luotujen triggereiden avulla. Triggerillä tarkoitetaan robotille annettua käskyä, joka antaa luvan käynnistää automaatioprosessi. Käskyt voivat olla robotille ajastettuja, mikä käynnistää suoraan pääprosessin tai robotin sivuprosessin. Sivuprosessi monitoroi esimerkiksi saapuvia ostotilauksia ja aktivoi robotin pääprosessin saadessaan tiedon uudesta tilauksesta.

#### 4.4 KC-Framework, Konecranes Oy:n RPA-rakenne

Konecranes Oy:ssä on rakennettu RPA-CoE:n toimesta oma UiPath-ohjelmointialustapohja (Kuva 4.), johon on rakennettu toiminnallisuuksia Konecranesin omien RPA-kehityksiin sopiviksi. Näitä kutsutaan generisiksi aktiviteeteiksi. Toiminnallisuudet on rakennettu niin, että ne ovat otettavissa nopeasti käyttöön ja sopivat Konecranes Oy:n käytössä olevien järjestelmien RPA-automaatiokehityksiin. Esimerkiksi SAP:n transaktioista on tehty aktiviteetti, johon annetaan arvona transaktiokoodi. Tämän UiPath-aktiviteetin ansiosta robotti osaa navigoida SAP:n sisällä haluttuun transaktioon. Sähköpostien lähettämiseen on myös oma geneerinen aktiviteetti. Se pitää sisällään sähköpostin rungon, johon on liitetty viestin sisällölle olennaista tietoa.

Kuva 4. KC-Framework, ohjelman tilat (States) ja niiden relaatiot



## **5 RPA - Dokumentaatio**

Tässä opinnäytetyöstä syntynyt dokumentaatio on tuotettu Konecranes Oy:n asiakirjamalleihin. Kappaleessa kuvataan dokumentaatioiden sisällöt sekä niiden tarkoitus ohjelman elinkaaren kannalta.

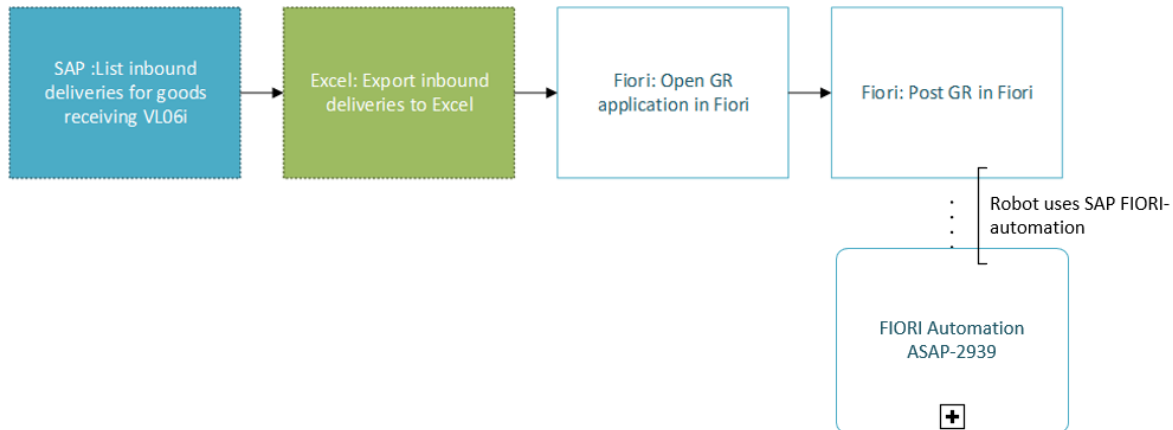
### **5.1 PDD-Dokumentaatio**

PDD eli Process Definition Documentation on automaatiosta syntyvä dokumentti, johon on kuvattu työnkulku ihmisen ja robotin tekemänä. Kuvaus kirjoitettiin niin, että lukija saa käsityksen prosessin eri vaiheista sekä myös robotin tekemästä työnkulusta. Tämä dokumentaatio on tärkeä prosessi-informaation tallennuspaikka koko ohjelman elinkaaren ajan suunnittelusta toteutukseen.

#### **5.1.1 AS-IS**

Nykyinen, ihmisen tekemä työnkulku kuvattiin yksityiskohtaisesti mallintamalla nykyisen työnkulun rakenne mahdollisimman yksityiskohtaisesti. Tätä kutsutaan dokumentaatioissa AS-IS vaiheeksi, nykytilan kuvaamiseksi. Käytännön työosuuksista tehdään Step-By-Step-kuvakaappaukset, mihin numeroidaan ja kirjoitetaan kuvaus eri työvaiheiden merkityksistä. Step by step-kuvasarjalla pystytään toistamaan työnkulku RPA-ohjelmoijan toimesta, kun suunnitellaan robotin ohjelmointia. Työnkulku käydään huolellisesti läpi ja siitä voidaan tapauskohtaisesti esimerkiksi tehdä videonauhoite, jolla voidaan toistaa ihmisen tekemä työnkulku. Työnvaihe on olennainen RPA-ohjelmoijalle nykyisen työnkulun ymmärtämiseksi. Lisäksi työnkulusta tehtiin graafinen mallinnus.

Kuva 5. AS-IS työnkulku kuvattuna



AS IS-vaiheessa kerättiin myös liiketoiminnan näkökulmasta tärkeitä tietoja prosessille. Olennaisia tietoja yrityksen RPA-kehityksen hyväksynnälle ovat automaation tuomat hyödyt. Robotin taloudellista tarvetta tarkastellessa tutkitaan ihmisen tekemän työnkulun toistuvuuden ja siihen käytetyn työajan vertaamista RPA-robotin vastaavaan suoritukseen.

Taulukko 1. Kuvassa verrataan ihmisen ja robotin työnkulkua ajallisesti

	Aika(min)	Määrä(kpl)	Yhteensä(tuntia)
<b>Ihmisen tekemä työnkulku:</b>	15	1400	350
<b>Robotin tekemä työnkulku:</b>	0,5	1400	11,7

### 5.1.2 TO-BE

Kuvaus nykyisestä työnkulusta hyväksytään CoE:n toimesta, kun AS IS-prosessi on valmis ja yksityiskohtaisesti kirjoitettu. Sen jälkeen tehdään TO BE-työnkulun mallinnus ja suunnittelu. Käytännössä TO BE-työnkulku tarkoittaa rakennettavan automaation ja sen suunnittelun dokumentointia ja käytännön ohjelmointia. Mallinnuksessa käytettiin Microsoft Visio -nimistä piirustusohjelmaa, joka on osa Microsoft Officea. Piirustusohjelmalla kuvataan graafisesti työnkulun vaiheita eri tasoilla (Business Process Modelling Notation, BPMN). Mitä syvemmillä tasoilla ohjelmaa pystytään mallintamaan, sitä sujuvammin käytännön ohjelmointia pystytään toteuttamaan.

Mallinnuksen suunnittelun tukena käytettiin AS IS-vaiheessa syntyneen step by step-työnkulun yksityiskohtaista kuvausta ja sen avulla työnkulku toistettiin manuaalisesti UiPath:ssa olevalla

UiPath Explorer-työkalulla. Työkalulla tutkittiin ja testattiin robotin näkökulmasta tapahtuvia toimintoja SAP:ssa ja verkkoselaimessa. Jokainen yksittäinen klikkaus käyttöliittymässä tai esimerkiksi robotin kyky nähdä verkkosivun elementti tuli huomioida ohjelmoinnin suunnittelussa.

Dokumentaatioon kirjatattiin robotin toiminta ohjelmassa tapahtuvien poikkeustilanteiden (exceptions) hallitsemiseksi. Automaatiolle mahdollisesti annettujen sääntöjen (Business Rules) nimet ja merkitykset sekä muutokset automaatiossa kirjattiin niille varattuun taulukkoon. Näitä ohjelmarakenteellisesti tärkeitä kirjauksia käytetään ohjelmaa suunnitellessa, mutta myös esimerkiksi valmiin ohjelman vianetsinnässä, kun ohjelma on siirretty testiympäristöstä tuotantoympäristöön.

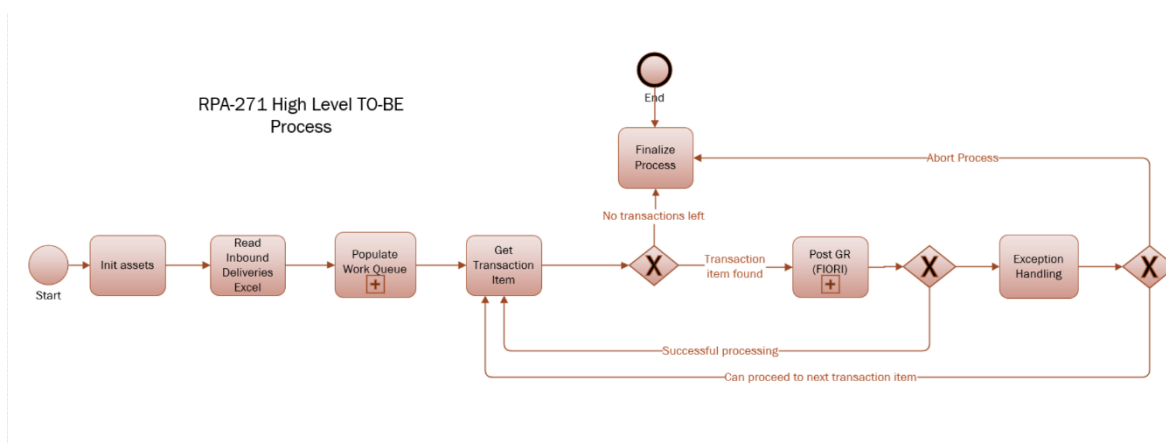
## **5.2 SDD-dokumentaatio**

Software Definition Document (SDD) on RPA-kehityksen tekninen dokumentaatio. Siinä kuvataan ohjelman toteutus yksityiskohtaisesti ja selitetään UiPath:ssa käytettyjen xaml-tiedostojen toimintaperiaatteet ohjelmarakenteessa. Tiedostot kirjoitettiin dokumentaatioon selitteineen ja niihin liittyvien muuttujien sekä argumenttien tarkoitukset kuvattiin yksityiskohtaisesti.

## 6 RPA-ohjelmointi erähallintadatan korjaamiseksi

Tämän RPA-ohjelman kehityksessä on käytetty KC-Frameworkkia UiPath:lle. Tässä luvussa käydään läpi erähallintadataan liittyvä ongelma, UiPath-ohjelmointialustan käyttöönotto sekä ohjelmistorobotiikan ohjelmointi. Ohjelmitava prosessi on transaktuaalinen, mikä tarkoittaa, että se on jaettu kahteen erilliseen ohjelmitavaan osaan. Ensimmäinen ohjelmitava osa on Dispatcher-prosessi, jossa robotti kerää automaatioissa käytettävän datan SAP:sta, jota se käyttää työjonon luomiseen Orchestratorissa. Toinen ohjelmitava osa on Performer-prosessi. Siinä robotti hakee työjonosta transaktion ja prosessoi sen viemällä toimitusnumeron SAP:n FIORIiin (Kuva 6.).

Kuva 6. Korkean tason BPMN-mallinnus automaatiosta



### 6.1 Erähallintadata ja sen ongelmat

Konecranes Oy käyttää erähallintadataa parantaakseen tuotteiden seurattavuutta järjestelmässä. Tietystä valmistuserästä toimitetuille tuotteille ja toimituksille jää järjestelmään eränumero (BATCH), jonka avulla voidaan esimerkiksi jäljittää alkuperäinen tuotantoerä. Tietyissä tilaustoimitusprosesseissa tuotteen tunniste vaihtuu tilausprosessin aikana, minkä takia eränumero ei kopioitu pakollisiin kenttiin SAP:ssa. Seurauksena on, että tuotteiden vastaanotto sisäisellä asiakkaalla ei ole järjestelmässä mahdollinen. Tämä edelleen estää sisäisen laskun prosessoinnin. Manuaaliseen vastaanottoprosessiin on tehty kehitys FIORI-sovelluksen taakse, jonka avulla työntekijät pystyvät tekemään tuotteen vastaanoton järjestelmässä. Ongelma on

kuitenkin suoratoimitusasiakkaiden kanssa käytössä oleva automaattinen vastaanotto, joka epäonnistuu taustalla ja joka huomataan vasta, kun laskutuksen kanssa tulee ongelma.

## 6.2 UiPathin alustaminen robotille

Ohjelmointi aloitetaan alustamalla robotille arvot, joiden avulla se pystyy käyttämään Orchestratoria ja toiminnanohjausjärjestelmää prosien läpiviemiseksi. Orchestratoriin on asetettu työjonoa ja asetteja (ulkoisia muuttujia) varten nimi mitä ohjelma käyttää. Tämän ohjelmoinnin työnimi on BDC271. Tätä nimitystä kutsutaan prefixiksi eli lyhenteeksi. Sitä käytetään myös Orchestratorissa ja sillä saadaan yhteys kaikkiin asetteihin sekä työjonoon. Robotti saa myös oletusarvona boolean-tyyppisen muuttujan eli "tosi" tai "epätosi" arvon. Sillä määritellään, suoritetaanko robotin työnkulku tuotanto- vai testiympäristössä.

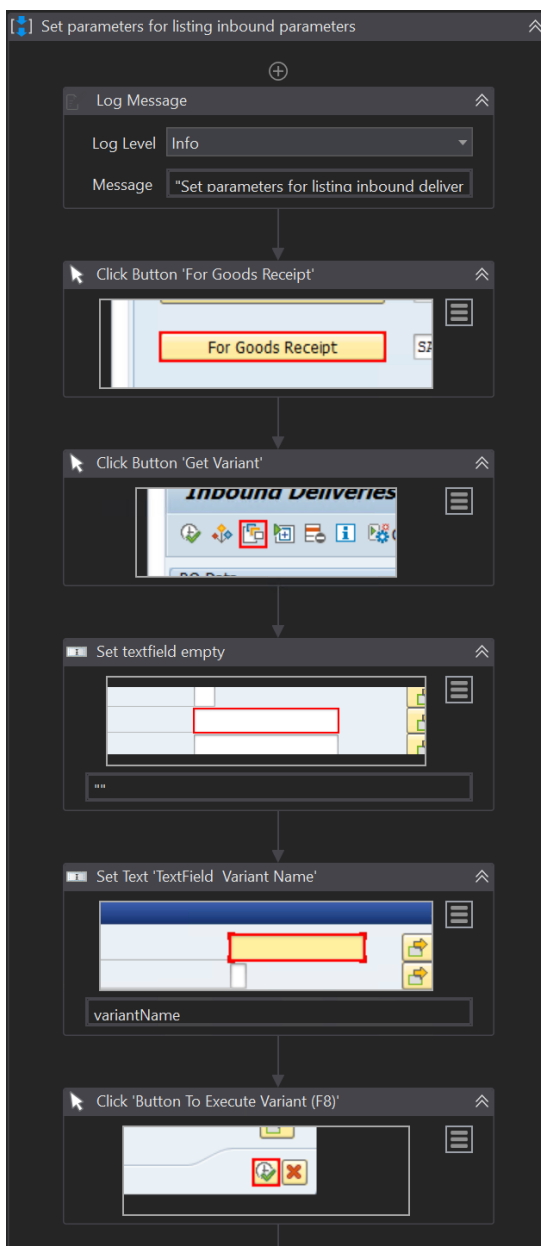
Konecranesilla SAP:n tuotantoympäristö on nimeltään OE8 ja testiympäristö on nimeltään OEQ. Asetti arvona valitaan käyttääkö työnkulku Microsoft Edge -selainta vai Googlen Chrome -selainta. Robotti käyttää annettua selainmääritystä FIORIn avaamiseen työnkulussa.

Koska robotti käyttää omia Windows käyttäjätunnuksia, myös vian hallintaa (exception handling) varten tulee asettaa sähköpostiosoite, johon robotti lähettää viestin virheen ilmaantuessa. Ohjelmassa on määritetty sähköpostiosoite liiketoimintayksikölle, mihin lähetetään viesti aina kun robotille tulee vastaan toimitusnumero, jota se ei saa käsitellä.

### 6.3 Dispatcher-ohjelmointi SAP:ssa

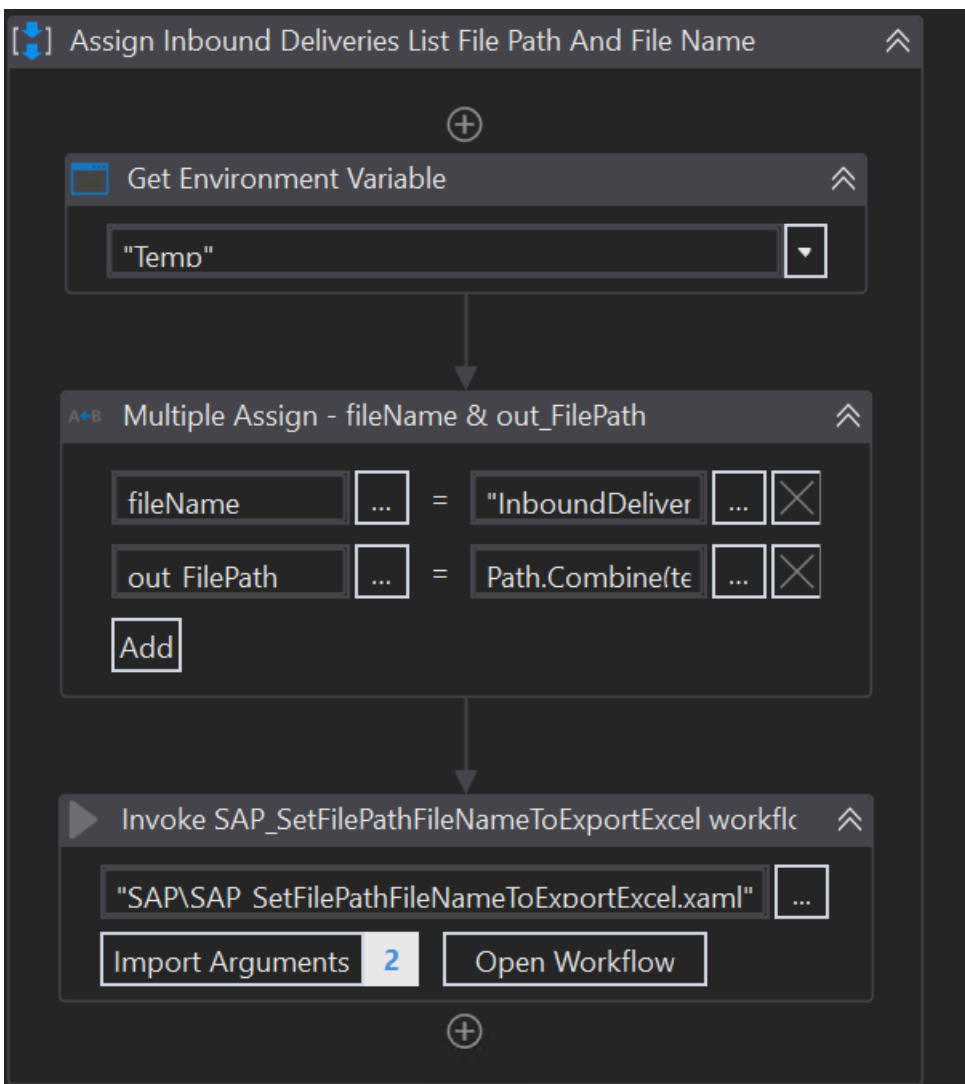
Ensimmäinen ohjelmitava osuus on Dispatcher-prosessi. Robotti ensin sulkee SAP:n, jos se on auki ja sen jälkeen avaa sen uudestaan. Tässä prosessissa robotti kirjautuu SAP-järjestelmään käyttäen Orchestratorissa sille annettuja SAP-tunnuksia. Onnistuneen kirjautumisen jälkeen robotti syöttää SAP:n transaktiokoodin "VL06i" käyttäen Set Text -aktiiviteettia, ja klikkaa "Execute" -painiketta SAP:ssa, mikä avaa Inbound Delivery Monitor -näkyvän.

Kuva 7. Työnkulku, jossa annetaan SAP:n parametrit käyttämällä varianttia



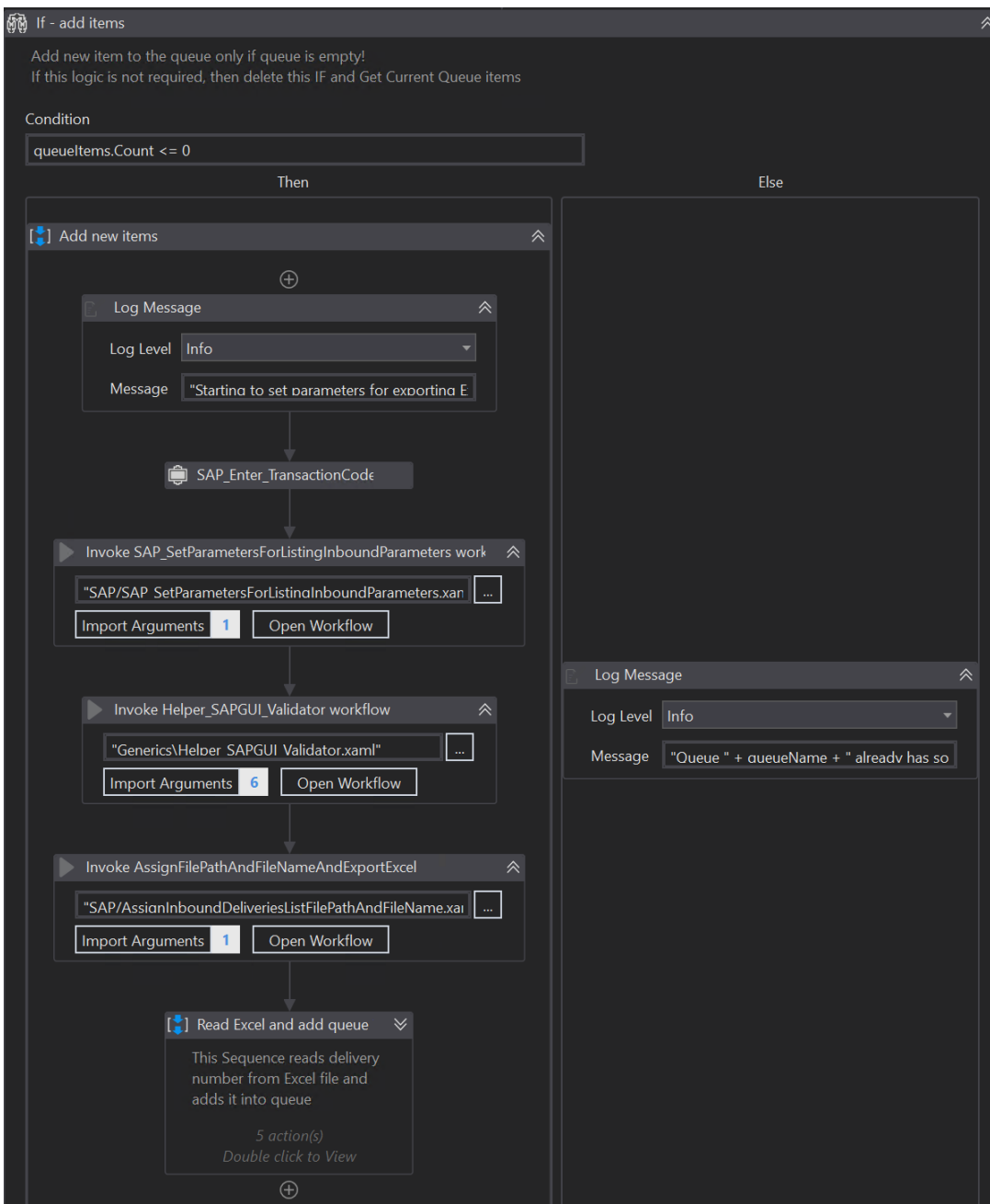
Robotti klikkaa "Inbound Deliveries" -painiketta ja siirtyy seuraavaan näkymään, johon on listattu ne toimitusnumerot, jotka robotin on tarkoitus syöttää FIORlin. Robotille on tehty skripti Excel-raportin ulosajoa varten käyttäen SAP:n omaa skriptaustyökalua. Robotti suorittaa skriptin, jonka lopputuloksena aukeaa Excel-tiedosto, johon on viety SAP:n toimitusnumerot. Excel-tiedostoa varten käytetään ympäristömuuttujaa, joka mahdollistaa UiPath:n aktiviteetin kautta tiedoston väliaikaisen tallentamisen Temp-nimiseen kansioon (Kuva 8.).

Kuva 8. Väliaikaisen tiedostosijainnin tallentaminen muuttujaan



Robotti sulkee Excel-tiedoston ja lukee sen käyttäen omaa aktiviteettia "Read Workbook". Tiedosto luetaan suoraan datatauluksi luomalla ulostulomuuttuja "inboundDeliveriesDT". Datataulua prosessoidaan For Each loopissa, kun luodaan työjonoa Orcehstratoriin. Looppi käy datataulun jokaisen rivin kerrallaan ja tarkastaa onko työjonossa prosessoitavaa. Jos löytyy prosessoitavaa, robotti syöttää yksitellen datataulun jokaisen toimitusnumeron Orchestratorin työjonoon käyttäen aktiviteettia "Add Queue Item". Kun kaikki prosessoitavat on lähetetty Orchestratoriin, robotti lopettaa Dispatcher-vaiheen (Kuva 9.) ja siirtyy tarkastamaan onko FIORI auki.

Kuva 9. Dispatcher-prosessi

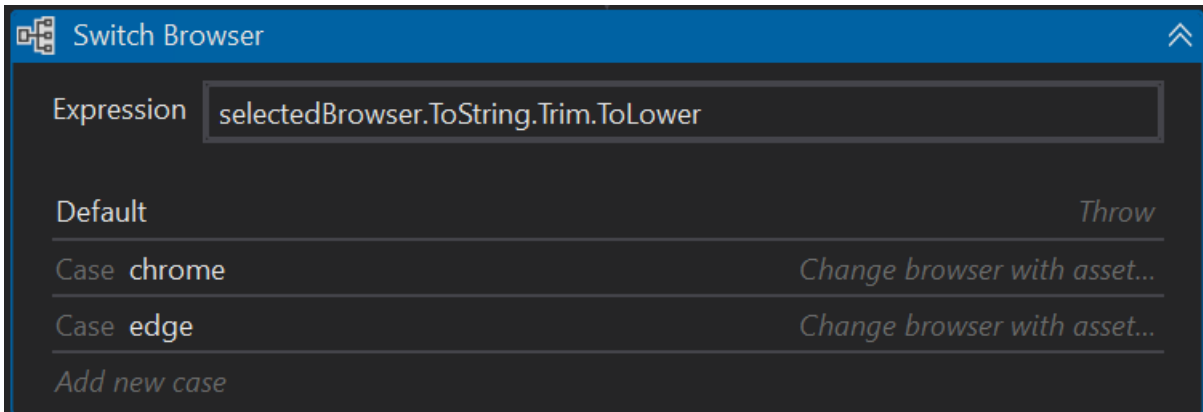


## 6.4 FIORIn avaaminen selaimessa

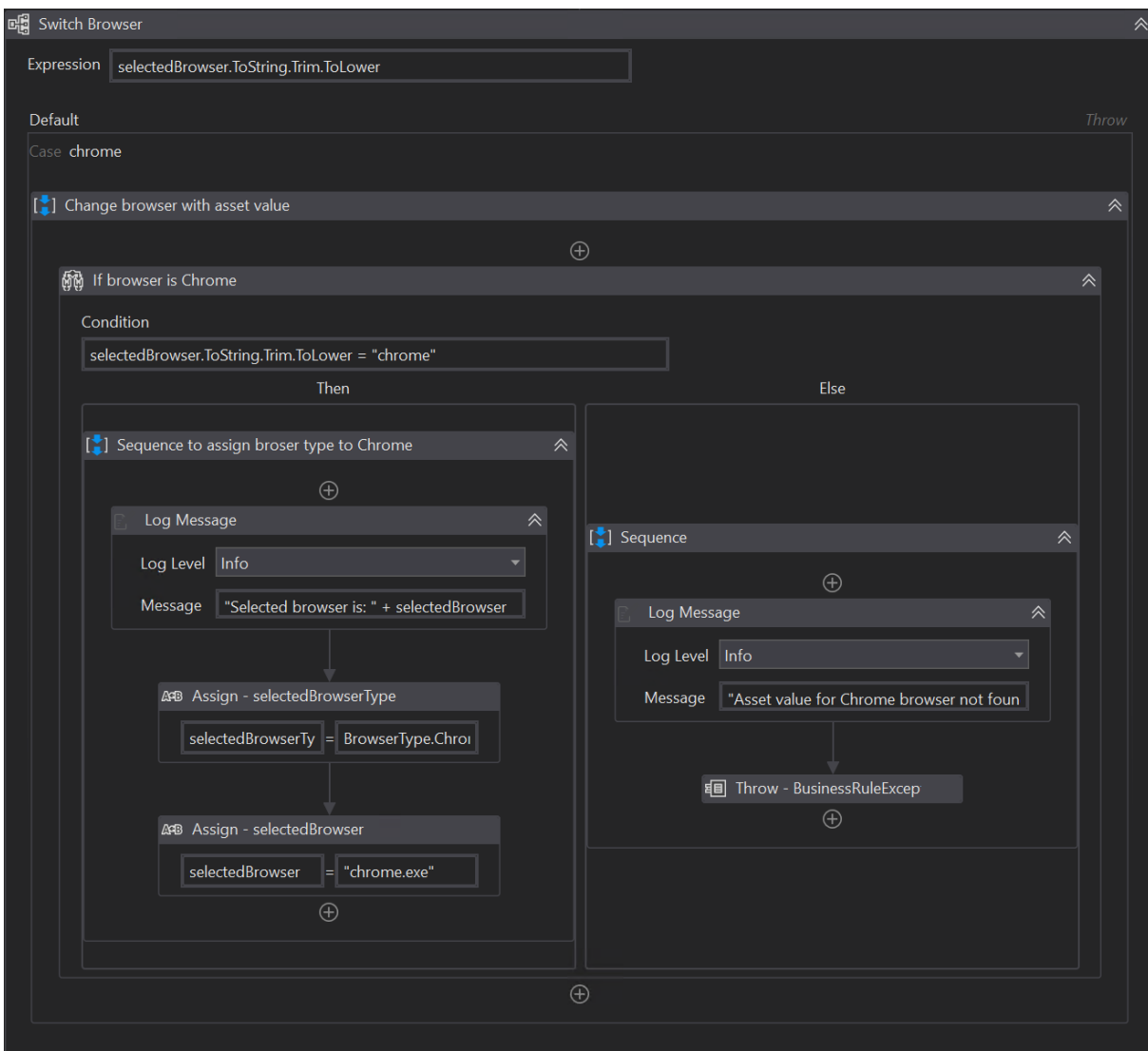
Robotti aloittaa tarkastamalla Orchestratorin työjonosta onko prosessoitavaa toimitusnumeroa. Työjonon ollessa tyhjä ja toimitusnumerot on käsitelty, robotti lopettaa ohjelman. Jos työjonossa on prosessoitavaa, aloittaa robotti kirjautumisprosessin. Kirjautumisprosessiin on tehty tunnistus, mihin on käytetty UiPathin aktiviteettia "PickBranch". Aktiviteetilla voidaan rakentaa robotille vaihtoehtoisia työpolkua, jolla käytännössä tarkoitetaan tiettyjen ehtojen täyttymistä robotin tutkiessa sopivaa työpolkua. Tässä ohjelmoinnissa käytettiin kahta eri selainta Microsoft Edgeä ja Google Chromea. Niitä varten rakennettiin Switch-aktiviteettia käyttäen vaihtoehtoinen työnkulku robotin käyttämän selaimen perusteella (kuva 11). Robotti saa assetti-arvon "edge" tai "chrome" Orchestratorista (kuva 12). Microsoft Edgellä kirjautuminen oli robotille yksinkertainen koska se käyttää Windowsin järjestelmäsalausanaa myös FIORlin, joten erillistä tunnusten asettamista ei tarvittu robotille. Google Chromeen kirjautuessa oli otettava huomioon, kirjautuuko robotti ensimmäistä kertaa vai ei. Robotin kirjautuessa ensimmäistä kertaa, tarvitsee FIORlin syöttää sähköpostiosoite käyttäjätunnukseksi ja sen jälkeen Orchestratorista haettu salattu salasana. Jos taas kirjautuminen on robotilla jo kertaalleen tehty, riittää että robotti valitsee käyttäjän listalta ja syöttää salasan.

Ennen kuin robotti voi ruveta käsittelemään Orchestratorin työjonoa on sen varmistuttava, että FIORI ei ole auki. Sitä varten UiPathissa on "Element Exists" -aktiviteetti, jolla voidaan tarkistaa graafisen käyttöliittymän elementtejä. Elementeistä saadaan "tosi" tai "epätosi" arvo, jota varten luodaan boolean-muuttuja. Kun robotti on tarkastanut avoimet selaimet se saa Orchestratorista assetti-arvona valitun selaimen (Kuva 10.), Edge tai Chrome, jolla se avaa FIORIn kirjautumissivun. Assetti-arvo on tallennettu ohjelman alussa alustettuun muuttujaan "selectedBrowser".

Kuva 10. Robotti saa Orchestratorista assetti arvona selaimen nimen, jonka perusteella se valitsee työnkulun selaimen valitsemiseksi prosessissa (Kuva 11.).

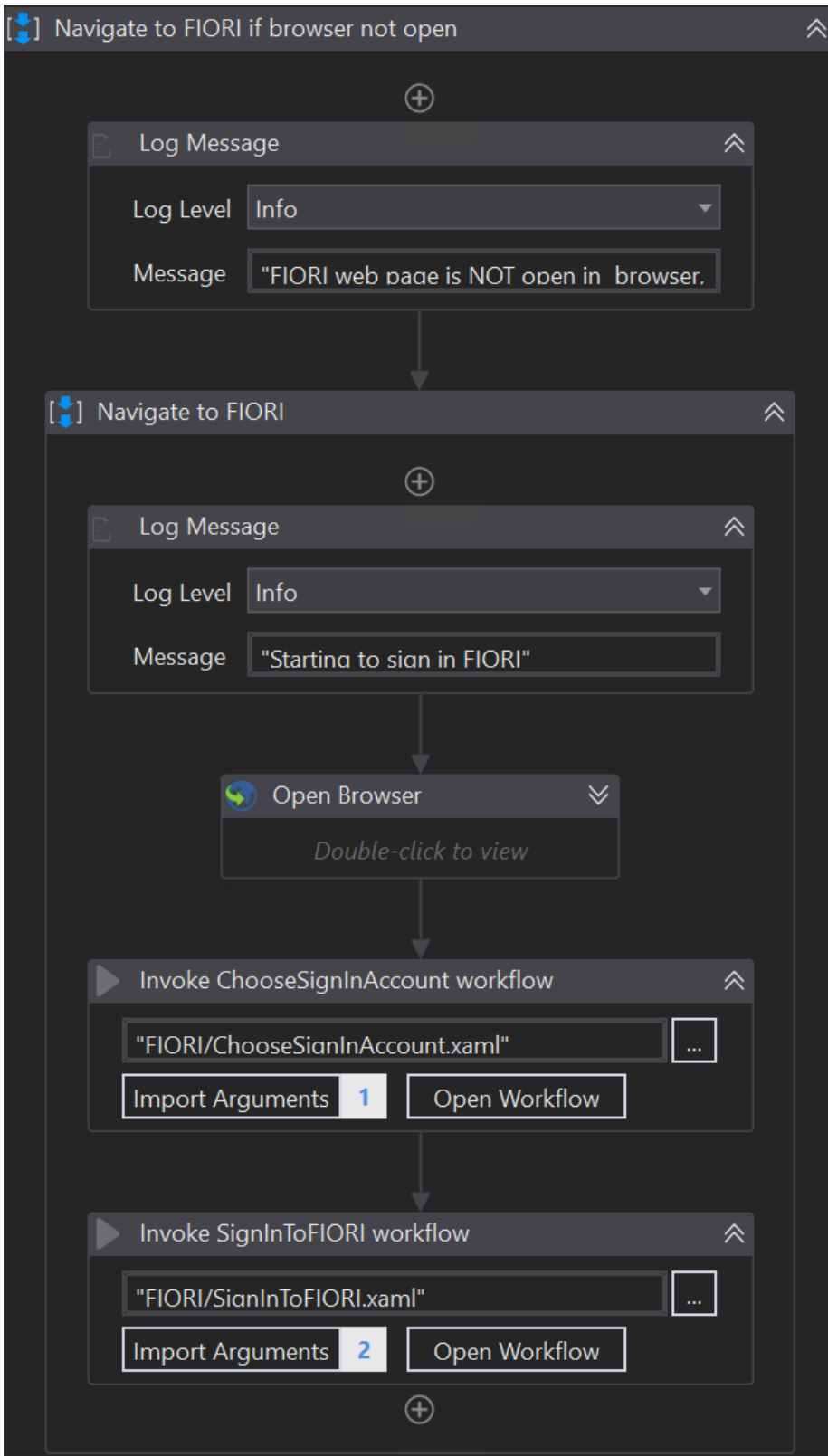


Kuva 11. Robotin työnkulku, kun se saa assetti-arvona "chrome"



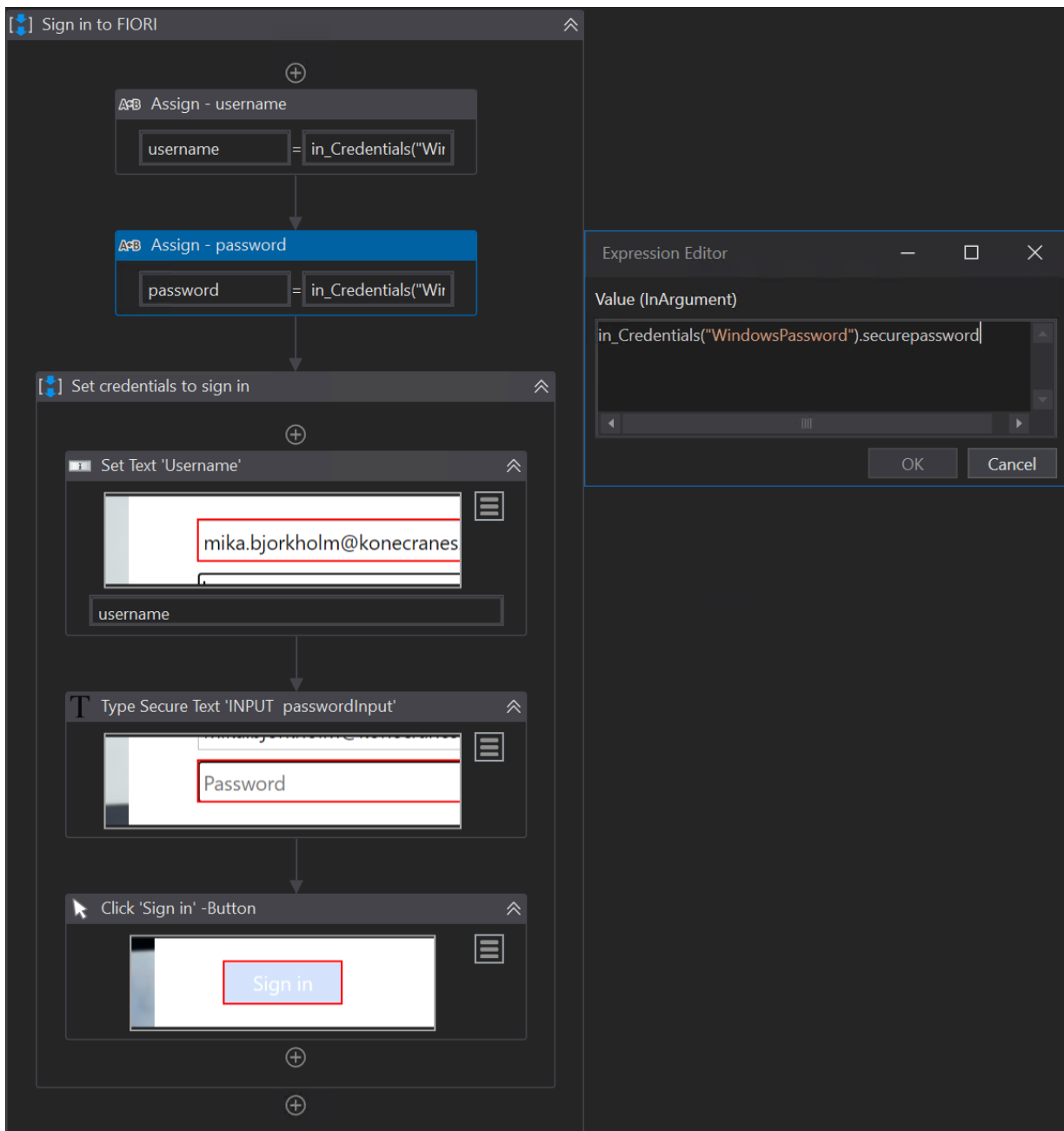
Robotti avaa assetilla saadulla arvolla halutun selaimen ja käyttää Orchestratorista saamiaan kirjautumistunnuksia sisäänkirjautumisessa (Kuva12.).

Kuva 12. Robotti avaa selaimen ja kirjautuu FIORLin



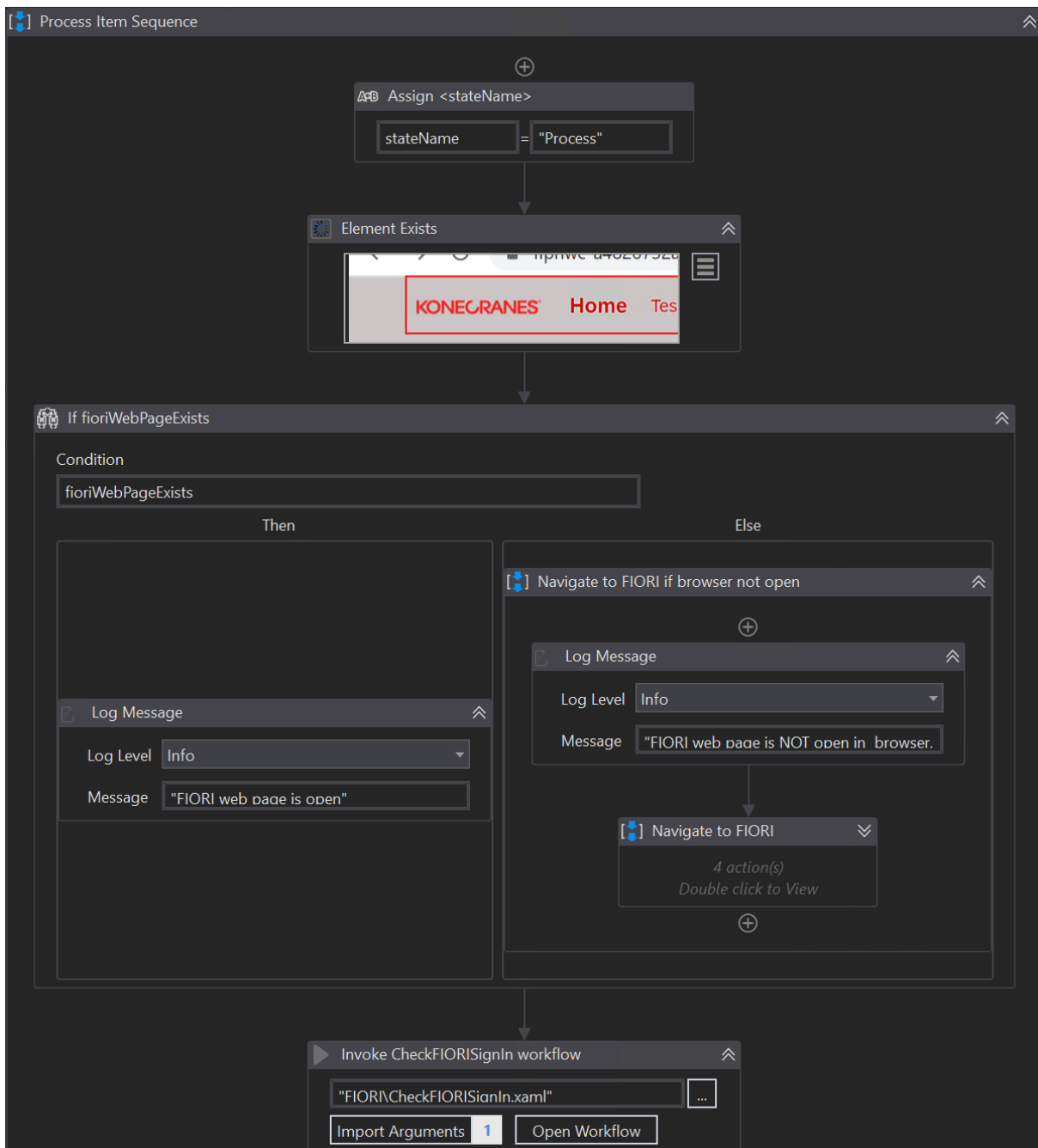
Salasana kryptataan käyttäen UiPath:n aktiviteettiä Set Secure Text (Kuva 13.). Aktiviteetille on ominaista lähettää Orchestratorista saatu salasana turvallisena SecureString-tyyppisenä muuttuja kirjautumissivustolle.

Kuva 13. Orchestratorista haettu salasana kryptataan kirjautuessa FIORlin



Kun kirjautuminen on onnistunut, robotti varmistaa onnistuneen kirjautumisen lukemalla Element Exists-aktiviteetilla näytöltä näkyykö selaimen oikeassa yläkulmassa Sign In-logo (Kuva 14.). Kirjautumisen epäonnistuessa lähettää UiPath siitä sähköpostin, johon on kirjattu UiPath:n generoima viesti järjestelmävirheestä.

Kuva 14. Robotti tarkastaa onko selain auki. Jos selain ei ole auki robotti avaa sen.



Kirjautumisen onnistuessa robotti jatkaa FIORIn etusivulle ja klikkaa "Inbound Deliveries"-elementtiä ja etenee seuraavalle sivulle mihin syötetään toimitusnumero.

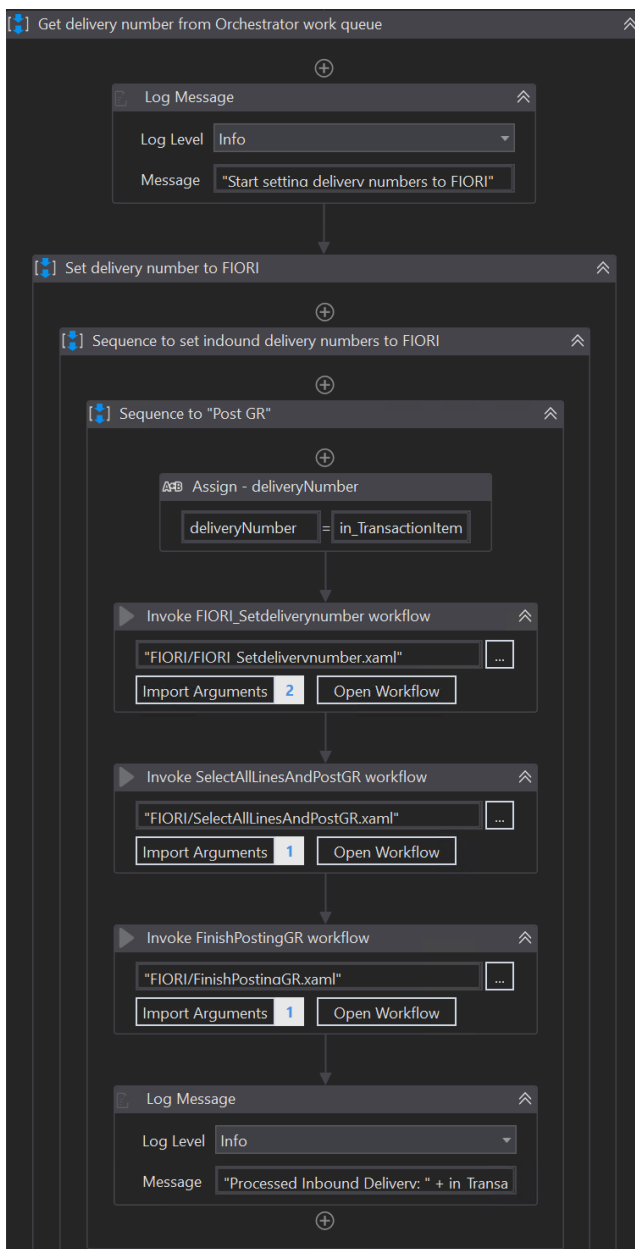
## 6.5 Performer-ohjelmointi FIORI:ssa

Onnistuneen kirjautumisen jälkeen robotti aloittaa työstämään työjonoa Orchestratorista. UiPath käyttää nimitystä Item, kun se vie työjonoon prosessoitavaa. Kun robotti aloittaa prosessoinnin, puhutaan robotin transaktioista. Robotti käyttää hyväkseen KC-Frameworkin Get Item-tilaa, johon on ohjelmoitu robotille ominaisuus hakea yksitellen työjonosta transaktioita (Kuva 15.).

Orchestratorista haettu toimitusnumero tallennetaan muuttujaan ja se viedään FIORI:n käyttäen Set Text-aktiviteettia. Robotti asettaa toimitusnumeron sille määrättyyn kenttään FIORI:ssa ja käyttää aktiviteettiä Hot Key, eli painaa enter-painiketta (kuva 16).

Jos toimitusnumero on järjestelmän hyväksymä, päästään prosessoimaan FIORI:n automaatiolla toimitettavien tavaroiden kuitti (Goods Receipt). Robotti tarkastaa näkykö sivulla kirjautumisesta indikoiva kuvake (Signed In) sivun oikeassa yläkulmassa. Robotti klikkaa Post GR-painiketta suorittaakseen erähallintadatan korjauksen.

Kuva 15. Robotti vie toimitusnumeron FIORlin.



Kuva 16. Toimitusnumero on hyväksytty FIORlin Goods Receipt -näkymän avautuessa robotille.

FinishPostingGR > Finish Posting GR

Goods Receipt

flpncw-a4826732a.dispatcher.hana.ondemand.com/sites/fiori#ZWMPOInboundDeliveryGRSemObj-display&/Overview/9180908025

KONECRANES Goods Receipt Test

GR for Deliv 9180908025 Find material

Vendor: 1255967 - Konecranes Finland Oy / Parts

Document date: 16.02.2021

Posting date: 16.02.2021

Delivery Note: [Empty]

Bill of Lading: [Empty]

Contents

<input type="checkbox"/>	PO I...	Material	Quantity			
<input checked="" type="checkbox"/>	10	<b>52678223</b> ROPE 24-32X7R-B-LHO-551.4	1,000	M	Batch managed item	PO Item texts exist Account Assignment: S - Siebel AAC Item Category: S - Third-party Sales order: 280529713 - 10 Delivery date: 19.02.2021

Add file Post GR

## 7 Johtopäätökset ja pohdinta

Työtehtävien automatisointi on tulevaisuudessa yksi yritysmaailman tärkeimmistä työkaluista. Tässä toiminnallisessa opinnäytetyössä kehitettiin automaatiota Konecranes Oy:ssä. Opinnäytetyötä tehdessä opitut työmenetelmät tukevat aiemmista tutkimuksista ja kehityksistä syntynyttä käsitystä siitä, että työtehtäviä voidaan automatisoida monimuotoisesti. Automatisoinnilla yritys voi saavuttaa taloudellista hyötyä ja ajallista säästöä liiketoiminnassa. Myös tämän opinnäytetyön lopputulos antaa enemmän aikaa ihmisten työajankäytölle työtehtäviin, mitkä ovat robotille haastavia.

Käytännössä robotille rakennettiin kyky suoriutua annetusta tehtävästä ihmisen työnkulkua vastaavalla tavalla. Robotti pystyy reagoimaan niihin ongelmakohtiin, jotka tunnistettiin työnkulun suunnittelun ja testauksen aikana. Kuitenkin muuttuva selainpohjainen graafinen käyttöliittymä todennäköisesti antaa robotin koodiin ylläpitotehtäviä ja päivityksiä, jotka kuuluvat osaltaan työnkuvaan. Robotilla automatisoitu työnkulku on haavoittuvainen mahdollisille muutoksille prosessissa käytetyille ohjelmille. Hyötydatan osalta automaatioon rakennettiin mahdollisuus irrottaa prosessissa syntyvää dataa enemmänkin robotin käytettäväksi, jos asiakas joskus niin haluaa. Tässä opinnäytetyössä ylimääräiselle datalle kuitenkin ei ollut käyttöä, joten sitä ei otettu talteen.

Kehitystyössä ajoittain ilmenneet ongelmat johtuivat lähinnä opinnäytetyöntekijän aluksi vähäisestä kokemuksesta toiminnanohjausjärjestelmän käytössä. Opinnäytetyön edetessä opinnäytetyöntekijä saavutti kuitenkin laaja-alaisen käytännön kokemuksen järjestelmästä. Tästä kokemuksesta on jatkossa varmasti hyötyä Konecranes Oy:n ohjelmistorobotiikkakehitysten kohdistuessa suurimmaksi osin SAP:n tuotannonohjausjärjestelmään. UiPath:n osalta oppimisprosessi oli mahtava kokemus ja opittujen menetelmien sekä käytäntöjen osalta opinnäytetyö antoi kauaskantoiset eväät jatkuvaan itseopiskeluun ohjelmistorobotiikan saralla. Ohjelmistorobotiikka on suhteellisen nuori ja voimakkaasti kasvava ala. Sen tehokkuus liiketoiminnassa sekä työllistävä vaikutus tekijöille luovat hyvät tulevaisuuden näkymät kehittyä osaajaksi ohjelmistorobotiikan moniulotteiseen maailmaan.

## 8 Yhteenveto

Opinnäytetyössä käytettyjen työmenetelmien sekä ohjelmistorobotiikan eri vaiheista voi tehdä päätelmän, että automatisoitujen työtehtävien tulevaisuus on kasvamassa yrity maailman yhdeksi tärkeimmistä työkaluista. Sen monimuotoisuus ja sopivuus eri työtehtävien automatisointiin on suuri etu yrityksille, kun mietitään taloudellisia sekä työtehtävien ajallisia säästöjä liiketoiminnassa. Ajallisesti ihmisten työtehokkuutta voidaan resursoida enemmän ajattelua vaativiin tai muuten monimutkaisempiin työtehtäviin kuin yksinkertaisiin ja toistuviin työtehtäviin. Arvioidessa liiketoimintaa taloudellisesta näkökulmasta on tärkeää myös huomioida, että ohjelmistorobotiikan hyötyjä mitataan laskennallisesti ja säästöjä usein verrataan ihmistyövuosiin. Väistämättä se vaikuttaa henkilöstön vahvuuden uudelleen laskentaan ja näin myös mahdollisiin henkilöstövähennyksiin. On myös huomioitava ohjelmistorobotiikan ollessa monessa yrityksessä nouseva trendi, tulee myös siihen tarvittavien resurssien suunnittelu olla tärkeässä roolissa (Wibbenmeyer 2018).

Tutkimuskysymyksiin vastaaminen toteutui hyvin ja opinnäytetyön lopputulos oli onnistunut. Ohjelmointi eli käytännön osuus valmistui aikataulussa ja sen tarkastus aloitettiin 1.3.2021 Konecranes Oy:n RPA CoE:n toimesta. Ohjelmointi hyväksyttiin tuotantoon 5.3.2021.

## Lähteet

Asatiani, A., & Penttinen, E. (2016). Turning robotic process automation into commercial success – Case OpusCapita. *Journal of Information Technology Teaching Cases*, 6(2), 67–74. <https://doi.org/10.1057/jittc.2016.5>

Schultz O. (2017). *Using SAP ERP. An Introduction for Beginners and Business Users*

van der Aalst, W.M.P., Bichler, M. & Heinzl, A. (2018). Robotic Process Automation. *Business & Information Systems Engineering*, 60, 269-272.

Wibbenmeyer, K. (2018). *The Simple Implementation Guide to Robotic Process Automation (RPA): How to Best Implement RPA in an Organization*. iUniverse: Bloomington.

AI Multiple (2021). Käyttöastetutkimus ohjelmistorobotiikka yrityksistä. Haettu 21.3.2021

<https://research.aimultiple.com/best-rpa-software/>

UiPath (2021). UiPath-dokumentaatio. Haettu 27.2.2021

<https://docs.uipath.com/>

UiPath (2021). UiPath-keskustelufoorumi. Haettu 2.2.2021

<https://forum.uipath.com/c/build/activities/168>

Microsoft (2021). Microsoft Oy:n dokumentaatio Visual Basic. Haettu 2.4.2021

<https://docs.microsoft.com/en-us/dotnet/visual-basic/>

## **Liite 1: Aineistonhallintasuunnitelma**

Tämän opinnäytetyön aineistot säilytetään Konecranes Finland Oy:n palvelimilla sekä yrityksen Sharepoint ympäristössä. Opinnäytetyö ei sisällä salassa pidettävää materiaalia. Kehitysprojektin aikana tehdään PDD (Process Definition Documentation) ja SDD (Software Definition Document), johon kerätään teknistä tietoa projektista. Tämä tieto analysoidaan opinnäytetyötä varten opinnäytetyölle sopivaan muotoon. PDD ja SDD dokumentaatio rakennetaan Konecranes Oy RPA CoE:n SharePoint ympäristöön, mihin on tehty kansiorakenne prosessissa tuotettaville materiaaleille kuten dokumentaatioille ja mallinnetuille prosessikaavioille. Lisäksi PDD ja SDD-dokumentaatio tallennetaan yrityksen käyttämään JIRA Tracker -nimiseen pilvipalveluun, missä seurataan prosessin kehitystä, elinkaarta sekä ylläpidetään valmiita dokumentaatioita. Ohjelmistot sijaitsevat yrityksen kehityspalvelimella, missä myös ohjelmoidaan automaatioprosessi. Valmis ja hyväksytty ohjelma viedään tuotantoserverille.