

# Tilausprosessin toteuttaminen ohjelmistoro- botiikalla

## Tiivistelmä

Tekijä(t) Korhonen, Matti	Julkaisun laji Opinnäytetyö, AMK	Valmistumisaika 2022
	Sivumäärä 28	
Työn nimi <b>Tilausprosessin toteuttaminen ohjelmistorobotiikalla</b>		
Tutkinto ja koulutusala Insinööri (AMK), Tieto- ja viestintätekniikka		
Toimeksiantajaorganisaatio (jos opinnäytetyöllä on toimeksiantaja) Telia Finland Oyj		
Tiivistelmä <p>Tämän opinnäytetyön tavoitteena oli perehtyä ohjelmistorobotiikkaan ja automatisoida uusien mobiililaitte tilauksien prosessi ohjelmistorobotiikka työkalulla. Työssä käytettävä ohjelmistorobotiikka työkalu oli nimeltään Blue Prism. Työn aihe syntyi toimeksiantajan tarpeesta saada laitetilauksien käsittelyprosessia nopeammaksi ja edullisemmaksi.</p> <p>Työssä esitellään yleisesti ohjelmistorobotiikka tekniikkaa. Tarkoituksena oli perehtyä ohjelmistorobotiikan käyttökohteisiin ja sen tuomiin hyötyihin. Tämän lisäksi perehdytään Blue Prism ohjelmistorobotiikka työkaluun ja sen toiminnallisuuksiin. Blue prism on ohjelmistorobotiikan markkinajohtaja, joka tarjoaa low code -työkalun automatisointi ratkaisujen rakentamiseen.</p> <p>Käytännön toteutuksessa automatisoitiin tilausprosessi, jossa mallinnettiin kahta eri järjestelmää ja käytettiin hyväksi API rajapintaa. Prosessissa käsiteltävät tilaukset haettiin Telian omasta tilausjärjestelmästä.</p> <p>Lopputuloksena voidaan todeta, että automatisoitava tilausprosessi on suoraviivaista tietojen siirtämistä järjestelmästä toiseen. Opinnäytetyössä käsitellään tällä hetkellä vain tietyntyyppisiä laitetilauksia. Tulevaisuudessa olisi tarkoitus saada kaikki Telian lähtevät laitteet kulkemaan saman prosessin kautta. Opinnäytetyö on saatu vietyä onnistuneesti tuotantoon.</p>		
Asiasanat Blue Prism, Ohjelmistorobotiikka, RPA, automatisointi, prosessi		

## Abstract

Author(s) Matti Korhonen	Type of Publication Thesis	Published 2022
	Number of Pages 28	
Title of Publication <b>Implementing order process with RPA</b>		
Degree, Field of Study Bachelor of Engineering, Information and communications technology		
Organisation of the client (if the thesis work is commissioned by another party) Telia Finland Oyj		
Abstract <p>The goal of this thesis was to learn about the RPA and automate process of new mobile device orders with RPA tool. RPA tool that was used at this work is called Blue Prism. Subject of the work arose from the client's need to make the processing of device orders faster and more affordable.</p> <p>Work introduces RPA technology in general. Purpose was to become familiar with the application of RPA and the benefits it brings. In addition to this, we will orientate ourselves with the Blue Prism software and its functionality. Blue Prism is the market leader in RPA, offering a low code software for building automation solutions.</p> <p>In the practical implementation, ordering process was automated, where two different systems were modeled, and the API interface was used. In the process, orders were retrieved from Telia's own ordering system.</p> <p>As a result, it can be stated that the automated ordering process is a straightforward transfer of data from one system to another. The thesis currently deals only with certain types of device orders. In the future, the plan would be to make all Telia's outgoing devices go through the same process. The thesis has been successfully put into production.</p>		
Keywords RPA, Blue Prism, automation, process		

## Sisällys

1	Johdanto.....	1
2	Ohjelmistorobotiikka .....	2
2.1	Ohjelmistorobotiikka eli RPA.....	2
2.2	Historia .....	2
2.3	Käyttökohteet.....	3
2.4	Teknologiat .....	4
2.5	Screen scraping.....	5
2.6	Hyödyt .....	6
2.7	RPA:n elinkaari.....	8
3	Blue prism.....	10
3.1	Historia .....	10
3.2	Käyttöliittymä .....	10
3.3	Process Studio .....	11
3.4	Objektit .....	12
3.5	Application modeller .....	13
3.6	Työjono.....	15
3.7	Poikkeuskäsittely .....	15
4	Prosessin käsittely ohjelmistorobotiikalla .....	16
4.1	Työn aloitus .....	16
4.2	Projektissa käytettävät menetelmät .....	16
4.3	Prosessin luonti .....	17
4.4	Järjestelmien mallinnus .....	18
4.4.1	NGSF .....	18
4.4.2	SAP ERP .....	19
4.5	Tilauksen hakeminen ja työjonon täyttö .....	20
4.6	Tilautustietojen hakeminen .....	22
4.7	Tilauksen luonti.....	23
4.8	Tilauksen lähetys ja sulkeminen .....	26
4.9	Manuaalikäsittely .....	27
5	Yhteenveto ja pohdinta .....	28
	Lähteet .....	29

## 1 Johdanto

Ohjelmistorobotiikka tarjoaa käyttäjälleen työkalun liiketoimintaprosessien automatisointiin. Työkalulla pystytään automatisoimaan säännöllisiä ja toistuvia työtehtäviä ilman ohjelmointi kokemusta. Ohjelmistorobotiikalla rakennetut virtuaalityöntekijät, eli robotit, toimivat käyttöliittymätasolla. Robottien avulla pystytään parantamaan työntekijöiden tehokkuutta ja työtaakkaa sekä vähentämään yrityksen kustannuksia.

Tämän opinnäytetyön toimeksiantaja on Telia Finland Oyj, jonka pääkonttori sijaitsee Helsingissä. Vuonna 2002 ruotsalaisen Telian ja suomalaisen Soneran fuusioitumisen seurauksena syntyi Telia Company. Telia Finland Oyj kuuluu Telia Company -konserniin Suomen maayhtiönä. Telia on teleoperaattori, joka tarjoaa tietoliikennepalveluja sekä kiinteän laajakaistan yritys- ja kuluttaja-asiakkaille. Yrityksen palveluvalikoimaan kuuluu myös TV-palveluita kuten C-More ja Liiga. Vuonna 2021 yrityksen liikevaihto oli 1,2 miljardia euroa ja yritys työllisti noin 2700 henkilöä. Telian toimipisteitä löytyy ympäri Suomen, suurin osa sijaitsee kuitenkin Etelä- ja Keski-Suomessa.

Opinnäytetyön tavoitteena on automatisoida uusien mobiililaitte tilauksien käsittelyprosessi ohjelmistorobotiikka työkalulla. Opinnäytetyön aihe syntyi Telian tarpeesta saada uusien mobiililaitte tilauksien käsittelyprosessia nopeammaksi ja edullisemmaksi. Tarkoituksena on automatisoida uusien laitteiden tilausprosessi kokonaisuudessaan. Prosessissa robotin on tarkoitus työskennellä kahdessa eri järjestelmässä sekä hakea laitteiden tietoja API rajapintaa apuna käyttäen. Työssä automatisoitavat tilaukset koskevat 4G mobiililaitteita ja siihen kuuluvia materiaaleja ja asennustöitä.

Työ toteutetaan Blue Prism nimisellä ohjelmistorobotiikka työkalulla. Blue Prism tarjoaa kehittäjälle helppokäyttöisen käyttöliittymän automaatio ratkaisujen rakentamiseen. Blue Prism:llä prosessit rakennetaan raahaamalla ja pudottamalla objekteja käyttöliittymän piirtoalueelle. Piirtoalueelle rakennetaan vuokaaviomaisesti työnkulku, jossa robotin logiikka on rakennettu objekteilla.

## 2 Ohjelmistorobotiikka

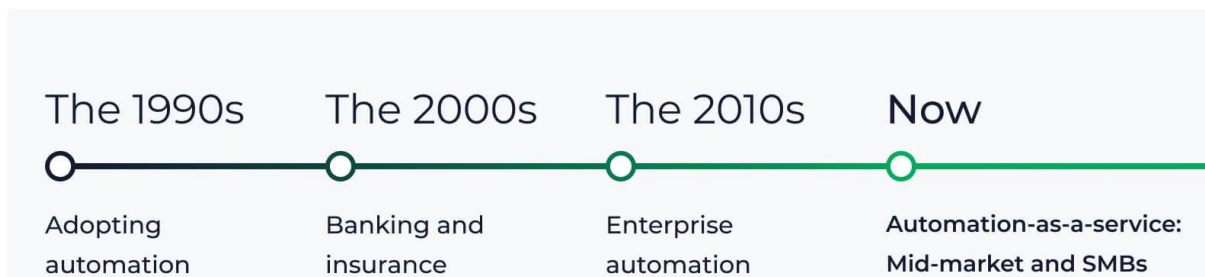
### 2.1 Ohjelmistorobotiikka eli RPA

Ohjelmistorobotiikka on uusi, jatkuvasti kehittyvä tekniikka, jonka avulla automatisoidaan toistuvia ja säännöllisiä tehtäviä, jotka suoritetaan tietokoneella. Toimintoja suorittavaa tietokoneohjelmistoa kutsutaan robotiksi, joka ohjailee hiirtä ja näppäimistöä opetetulla tavalla, sekä lukee näytöllä näkyviä tietoja. Ohjelmistorobotiikalla rakennetut robotit pystyvät työskentelemään vuorokauden ympäri ja ne pystyvät toimimaan nopeammin yksinkertaisissa työpöytätehtävissä kuin tietokoneella työtä tekevä ihminen. Nämä robotit voivat esimerkiksi avata sähköposteja, kirjautua järjestelmiin ja tehdä laskelmia. Robottien avulla pystytään tehostamaan ihmisen työpöytätehtäviä tai parhaimmassa tapauksessa korvata ihminen kokonaan.

Robotit voidaan mieltää digitaaliseksi työvoimaksi, joka kykenee olemaan vuorovaikutuksessa minkä tahansa järjestelmän tai sovelluksen kanssa. Ja koska robotit voivat mukautua lähes mihin tahansa käyttöliittymään tai työnkulkuun, sovelluksia tai olemassa olevia prosesseja ei tarvitse muuttaa automatisoidakseen. Ohjelmistorobotiikalla rakennettuja robotteja on helppo käyttää, niitä voidaan kloonata, muokata ja jakaa liiketoimintaprosessien suorittamiseksi koko organisaatiossa (Robotic Process Automation (RPA), 2022.)

### 2.2 Historia

Ohjelmistorobotiikan ensivaiheet alkoivat käyttöliittymätestauksen automatisoinnista screen scraping ohjelmistolla, jolla testattiin käyttöliittymän visuaaliset elementit, sekä varmistettiin että ne toimivat oikein. 90-luvulla markkinoilla oli niukasti erilaisia tietokonemalleja nykypäivään verrattuna. Windows 95 käyttöjärjestelmän ilmestyessä seurasi muutos, joka mahdollisti käyttöliittymätestauksen sujuvamman kehittämisen, kun vaatimukset ja näyttökoot monipuolistuivat. Windows 95 käyttöjärjestelmällä saatiin screen scraping ohjelmasta yhteensopivampi kohdesovellusten kanssa. Kuvassa 1 on esitetty ohjelmistorobotiikan historia.



Kuva 1. RPA:n historia

2000-luvun alussa yritykset päätyivät ketterään kehityskonseptiin, jonka keskeisin arvo on ihmisten priorisointi prosessien ja työkalujen edelle. Alettiin ymmärtämään tarve nopeuttaa yrityksen toimintaa pysyäkseen kilpailukykyisenä markkinoilla. Tästä syystä erilaiset käyttöliittymättestaukset ja automatisointiskriptit yleistyivät.

Ohjelmistorobotiikan avainelementit syntyivät 2000-luvun alussa. Suurimmaksi osaksi käytettiin vielä screen scraping -menetelmää, joka automatisoi tietojen poimimista sovelluksista muihin käyttötarkoituksiin. Tämä menetelmä on tehokas yrityksille, jotka joutuvat käsittelemään suuria määriä dataa.

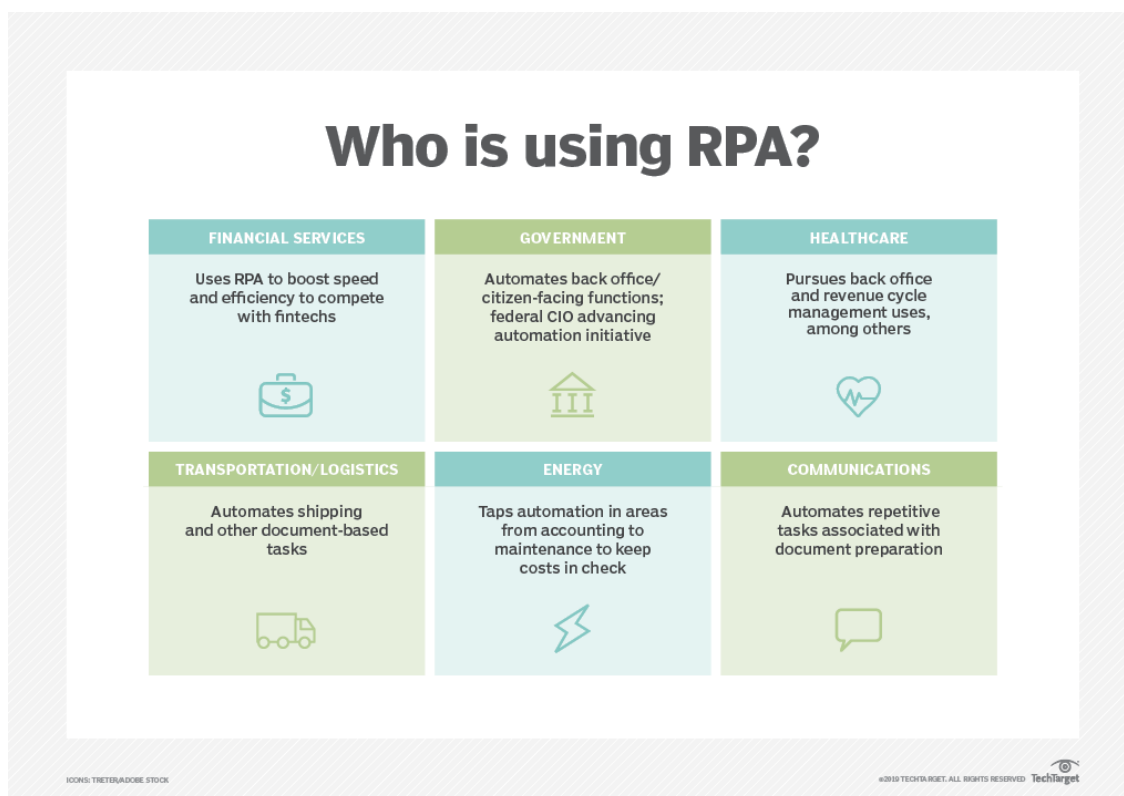
Käännekohta RPA-tekniikalle tapahtui 2012 vuoden tienoilla, kun se tunnustettiin virallisesti suuryritysten keskuudessa. Tähän vaikutti myös finanssikriisi, jonka seurauksena yritykset etsivät tapoja vähentääkseen kulujaan. Ohjelmistorobotiikkaa pidettiin yleisesti helpoana ja edullisena ratkaisuna digitalisoitumiseen.

2020-luku on ohjelmistorobotiikan demokratisoitumisen ja kasvavan vaikutuksen aikaa kaikilla talouden aloilla. Demokratisoituminen on mahdollista, kun siirrytään asteittain pois kalliista lisenssimaksuista ja luodaan kumppanuuksia palveluntarjoajien kanssa SaaS-mallin mukaisesti. Tämän avulla RPA tekniikka voi laajentua pieniinkin yrityksiin, tarjoten samalla suuria tuottavuustuloksia. (The Evolution of RPA: A 30-Year Journey, 2022, 4.)

### 2.3 Käyttökohteet

Ohjelmistorobotiikalla jäljitellään ihmisten tekemää tietotyötä eli käyttökohteita on rajattomasti. Robotiikan avulla yksinkertaistetaan prosesseja, vapautetaan resursseja muualta ja lisätään työn tehokkuutta (Skaler, 2022). Tämä mahdollistaa ihmisten vapauttamista toistuvasta ja väsyttävästä työstä usealla eri toimialoilla. Kuvassa 2 on esitetty esimerkki toimialat, joissa ohjelmistorobotiikkaa hyödynnetään. Finanssialalla ohjelmistorobotiikka auttaa rahoituslaitoksia tarjoamaan ympäri vuorokautisen tuen tärkeille toiminnoille ja prosesseille. Lisäksi sillä saadaan nopeutettua asiointipalveluja luomalla asiakaspalvelubotteja,

jotka kykenevät auttamaan asiakkaita arkisissa asioissa. Kunnat ja kaupungit saavat ohjelmistorobotiikan avulla mahdollisuuden virtaviivaistaa ja optimoida prosesseja. Terveystieteiden huollossa robotiikkaa voidaan käyttää esimerkiksi tapaamisaikojen varaamiseen ja käyntimaksujen lähettämiseen. Tällä tavalla saadaan helpotettua työntekijöiden työkuormaa. Logistiikka-alalla voidaan hyödyntää sähköisten lähetysten ja muiden dokumenttien lähettämistä. Ohjelmistorobotiikkaa voidaan myös hyödyntää energian säästämiseksi, sen avulla voidaan säätää lämmitystä, valoja ja vesihanan käyttöä sellaisissa kohteissa missä käydään harvoin. Asiakaspalvelualueilla ohjelmistorobotiikkaa voidaan hyödyntää usealla eri tavalla. On olemassa asiakaspalvelurobotteja, jotka kykenevät palvelemaan asiakkaita yksinkertaisissa asioissa. Asiakaspalvelurobottien ansiosta työntekijöiden työtaakka kevenee ja asiakas saa palvelua nopeasti. Lisäksi kaikki säännöllisesti toistuvat asiakaspalvelutehtävät, kuten esimerkiksi asiakirjojen täyttö pystytään automatisoimaan ohjelmistorobotiikalla.



Kuva 2. RPA:ta hyödyntävät toimialat (George Lawton, RPA, 2011)

## 2.4 Teknologiat

Ohjelmistorobotiikka on tekniikan kehityksen myötä noussut suuremmaksi kuin pelkäsi trendiksi. Se on uusi tapa, jolla liiketoimintaprosesseja voidaan ratkaista ja hallita. Tulevina

vuosikymmeninä robotit tai ihmiset eivät voi yksinään johtaa yrityksiä. Useat tulevaisuuden johtavat yritykset ovat niitä, jotka voivat yhdistää nämä kaksi tehokkaasti (Different Types of RPA, 2019).

Erilaisia RPA tuotteita on markkinoilla useita kymmeniä. Tällä hetkellä kolme suosituinta ovat UiPath, Automation Anywhere ja Blue Prism. Tämä käy ilmi Gartnerin tutkimuksessa, jossa vertaillaan eri RPA tuotteiden suosiota (Kuva 3).



Kuva 3. Gartnerin tutkimus ohjelmistorobotiikan markkinajohtajista ja -haastajista

Blue Prism perustettiin vuonna 2001, joka tekee siitä vanhimman teknologian RPA markkinoilla. Vuonna 2003 perustettu Automation Anywhere tarjoaa alustan yrityksen kokonaisvaltaiselle automatisoinnille. Näistä kolmesta suosituimman RPA tuotteen tarjoaa UiPath, joka oli Amerikan nopeimmin kasvavin yritys vuonna 2020. (RPA Market Size and Popular Vendors in 2022, 2022)

## 2.5 Screen scraping

Screen scraping, suomeksi näytön kaapiminen, on menetelmä millä otetaan talteen näytöllä olevia tietoja näytettäväksi tai käytettäväksi toisessa järjestelmässä. Screen scraping toimii

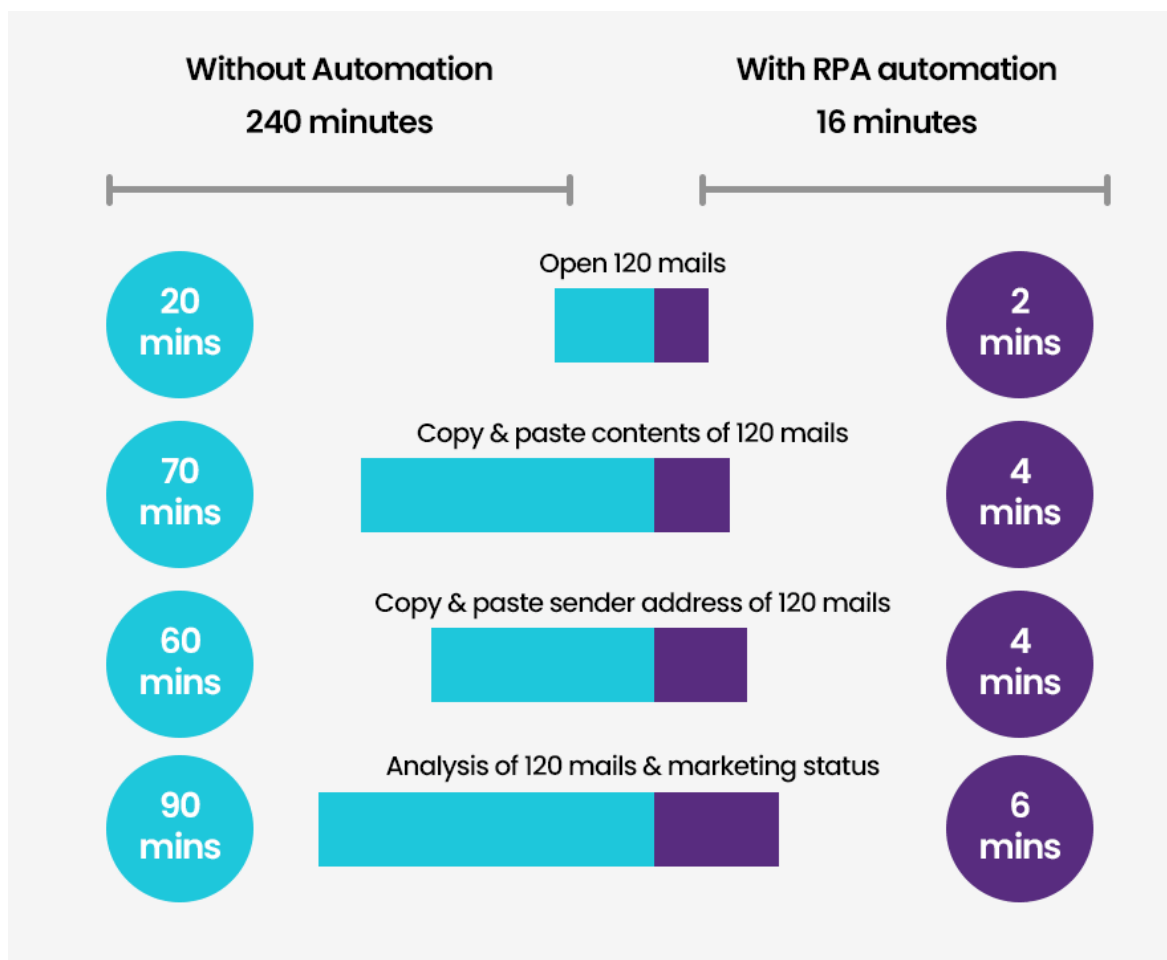
melkein missä tahansa käyttöliittymässä ja se antaa sovelluksille mahdollisuuden käyttää tietoja vanhoista järjestelmistä, jotka eivät tarjoa API:a tai muuta vastaavaa lähdetietojen käyttötapaa. Automaatioprosesseissa robotit kykenevät hakemaan nopeasti ja tarkasti tietoja screen scraping -menetelmällä (Everything You Need to Know About Screen Scraping Software, 2017.)

Screen scraping menetelmällä voidaan etsiä ja tunnistaa ennalta määritetyillä termeillä käyttöliittymäelementtejä. Esimerkiksi Excel taulukon sarakkeita tai verkkosivujen painikkeita ja tekstejä. Verkkosivuilta voidaan kaivaa tietoja screen scraping ohjelmalla monella eri tapaa. Yksi tapa on etsiä käyttöliittymästä kohdetta ennalta määrättyllä ID:llä. Toisena tapana Screen scraping menetelmä tarjoaa käyttöliittymässä olevien kohteiden ankkurointia, jolloin saadaan haluttu kohde etsittyä käyttämällä hyödyksi muita elementtejä. Ankkuroinnin avulla ohjelma osaa etsiä halutun kohteen hyödyntämällä muiden elementtien sijaintia.

## 2.6 Hyödyt

Ohjelmistorobotiikan keskeisin hyöty on yrityksen liiketoiminnan parantaminen. Robotit eivät tarvitse taukoja tehtävissään, vaan ne kykenevät työskentelemään taukoamatta vuoden jokaisena päivänä, jopa pyhäpäivinä. Yritykset säästävät resursseja, kun työtehtäviä siirretään robotin käsiteltäväksi. Tämän avulla kevennetään työntekijöiden työtaakkaa, sekä heitä voidaan sijoittaa kokonaan uusiin työtehtäviin. Robottien mahdollistamat työtehtävät voivat toimia ponnahduslautana korkeamman lisäarvon työpaikoille (20 Ways Bots Improve Businessess For The Better, 2017.)

Yleisesti ottaen yksi robotti voi suorittaa päivässä saman verran työtehtäviä kuin 2–5 kokopäiväistä työntekijää pystyy suorittamaan manuaalisesti. Kuvassa 4 vertaillaan robotin ja ihmisen käyttämää aikaa samojen työtehtävien tekemiseen (12 Popular Benefits of RPA in Business, 2022.)



Kuva 4. Ihmisen ja robotin käyttämä aika rutiininomaisessa työtehtävässä

Suuri hyöty kehittäjälle ohjelmistorobotiikassa on myös sen käyttäjäystävällisyys. Ohjelmistorobotiikkasovellukset tarjoavat visuaalisen käyttöliittymän prosessien rakentamiseen. Yleisesti robotille rakennetut prosessit ovat suoraviivaisia, vuokaavio mallin mukaisesti rakennettuja työnkuluja. Tämä mahdollistaa prosessien ymmärtämistä ilman aikaisempaa ohjelmointikokemusta.

Robotit eivät tee virheitä ihmisten tavoin, vaan ne tekevät juuri sitä mitä siihen on ohjelmoitu, ei enempää eikä vähempää. Virheet, joita robotin kanssa voi tulla, ovat yleensä prosessiin rakennettuja loogisia virheitä, jotka syntyvät robotin ohjelmointivaiheessa. Muita virheitä voi olla esimerkiksi tuntemattoman datatyypin käsittely, mallinnettu käyttöliittymäelementti ei täsmää oikean elementin kanssa tai tiedonsiirrossa tapahtuvat virheet.

## 2.7 RPA:n elinkaari

RPA:n onnistunut käyttöönotto tapahtuu ensimmäiseksi sillä, että yrityksen sisällä etsitään prosesseja, jotka voisivat jollain tavalla hyötyä siitä. Kuvassa 5 on esitetty RPA:n elinkaari. RPA:n elinkaaren alkuvaihe on löytää potentiaaliset prosessit mitkä voidaan automatisoida ohjelmistorobotiikka työkalulla. Prosessit dokumentoidaan ja arvioidaan, että voidaanko RPA:n avulla säästää prosessin kustannuksissa tai saadaanko prosessista tehokkaampi robotiikan avulla. Esimerkiksi Teliällä on oma sisäinen kanava, missä työntekijät voivat ehdottaa prosesseja mitä halutaan ohjelmistorobotiikka työkalulla automatisoida. Kaikista ehdotuksista tehdään arviot hyödyistä ja haitoista. Jokainen RPA-prosessi riippuu sen kannattavuudesta.



Kuva 5. RPA:n elinkaari (Life Cycle of RPA, 2022)

Prosessin soveltuvuusarkoituksen jälkeen määritetään RPA:n tavoitteet sekä suunnitellaan prosessin automatisointi vaiheet. Tämän jälkeen asiantuntijat luo PDD (Process Definition Document) -asiakirjan, johon sisältyy kaikki tiedot automatisoitavasta prosessista. Seuraavassa vaiheessa päätetään projektiin osallistuvat henkilöt, budjetti ja käytettävä aika. Analyttikko laatii vuokaavion prosessin kulun ymmärtämiseksi, mikä auttaa hahmottamaan prosessin automatisointia. Prosessista on myös hyvä luoda kuvallinen dokumentaatio, josta selviää robotin toiminta vaihe vaiheelta. Tämä auttaa kehittäjää prosessin kehityksessä, jos kehittäjällä ei ole automatisoitavasta prosessista aikaisempaa kokemusta.

RPA:n kehitysvaiheessa analyttikko luovuttaa prosessista luodun dokumentaation ja vuokaavion RPA-kehittäjälle, joka aloittaa kehittämään prosessia ohjelmistorobotiikka

työkalulla. Kehityksen jälkeen aloitetaan RPA prosessin testaaminen joko kehittäjän toimesta tai organisaation kehitystiimin voimin. Ensimmäisessä testausvaiheessa RPA prosessia testataan testiympäristössä, jotta voidaan ennakoida mahdolliset virheet ennen prosessin tuotantoon siirtämistä. Testausvaiheessa RPA prosessia testataan ja korjataan niin kauan, että vaadittavat testikerrat menevät onnistuneesti läpi. Onnistuneiden testikertojen jälkeen RPA prosessi viedään käyttöönottovaiheeseen (RPA Life Cycle, 2021.)

### 3 Blue prism

#### 3.1 Historia

Ohjelmistoautomaation asiantuntijatiimi perusti Blue Prismin vuonna 2001 kehittääkseen teknologiaa, jolla voidaan parantaa organisaatioiden tehokkuutta. Se on markkinoiden vaikiintunein RPA-yhtiö, jonka pääkonttori sijaitsee Isossa-Britanniassa (Blue Prism is raising \$130M to fuel new technologies, 2019.)

Blue Prism julkaisi ensimmäisen kaupallisen tuotteen vuonna 2003, joka oli nimeltään Automate. Vuonna 2005 julkaistiin Automaten päivitetty versio, joka sisältää ominaisuuksia suurien data määrien käsittelemiseen.

Blue Prism on tuonut markkinoille lukuisia teollisuuden ratkaisuja, joista jokaisella on aktiivisia asiakkaita. Käyttöliittymä on graafinen ja helppokäyttöinen, jotta yksinkertaisia automaatio ratkaisuja pystytään luomaan aloilla, jossa henkilökunnalla on vähemmän teknistä osaamista (Top 15 Robotic Process Automation (RPA) Companies, 2020.)

#### 3.2 Käyttöliittymä

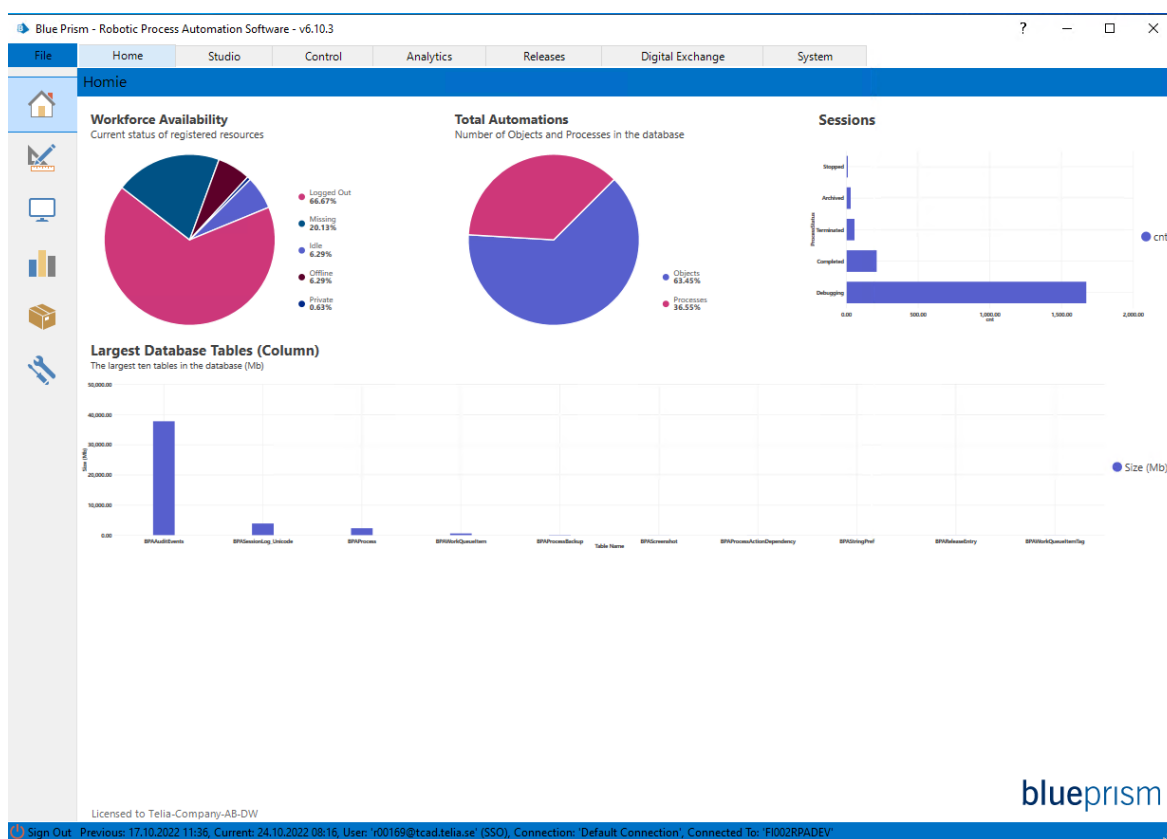
Blue Prism tarjoaa käyttäjilleen helppokäyttöisen automaatiopohjan ohjelmistorobottien rakentamiseen. Se toimii yhdellä käyttöliittymällä, joka on nimeltään Robotic Process Software. Yhdestä käyttöliittymästä löytyy kaikki mitä robotin rakentamiseen ja ylläpitämiseen tarvitaan. Kuvassa 5 on esitetty Blue Prism käyttöliittymä, josta löytyy seitsemän välilehteä. Robotin rakentamisessa oleelliset välilehdet ovat Studio ja Control -välilehdet.

Studio välilehdeltä löytyy hakemistoluettelo kaikista tehdyistä prosesseista. Prosessissa käytettävät objektit löytyvät myös Studio välilehdeltä, niitä pystytään muokkaamaan tai voidaan tehdä kokonaan uusia. Blue Prism tarjoaa runsaasti objekteja, jotka soveltuvat hyvin yksinkertaisiin automaatioihin. Esimerkiksi Microsoft 365 -sovelluksien automatisointiin tarvittavat objektit löytyvät Blue Prism:in omasta objektikirjastosta.

Control välilehdellä ylläpidetään robotin toimintaa ja siellä pystytään tarkastelemaan robotin työkulkua. Välilehdeltä voidaan tarkastella esimerkiksi robotin työjonoa, joka kertoo, miten robotti on suoriutunut työn eri vaiheissa. Analytics välilehti tarjoaa saman visuaalisen näytön kaavioista ja mittareista kuin kuvassa 6 on esitetty Blue Prism:in käyttöliittymän kotisivulla. Kaaviot ja mittarit kuvastavat kuinka paljon automaatioprosesseja on tällä hetkellä yrityksen verkossa.

Blue Prism paketteja voidaan siirtää tai tuoda eri Blue Prism ympäristöstä Release välilehden kautta. Paketti voi olla esimerkiksi valmis prosessi tai objekti, joka siirretään

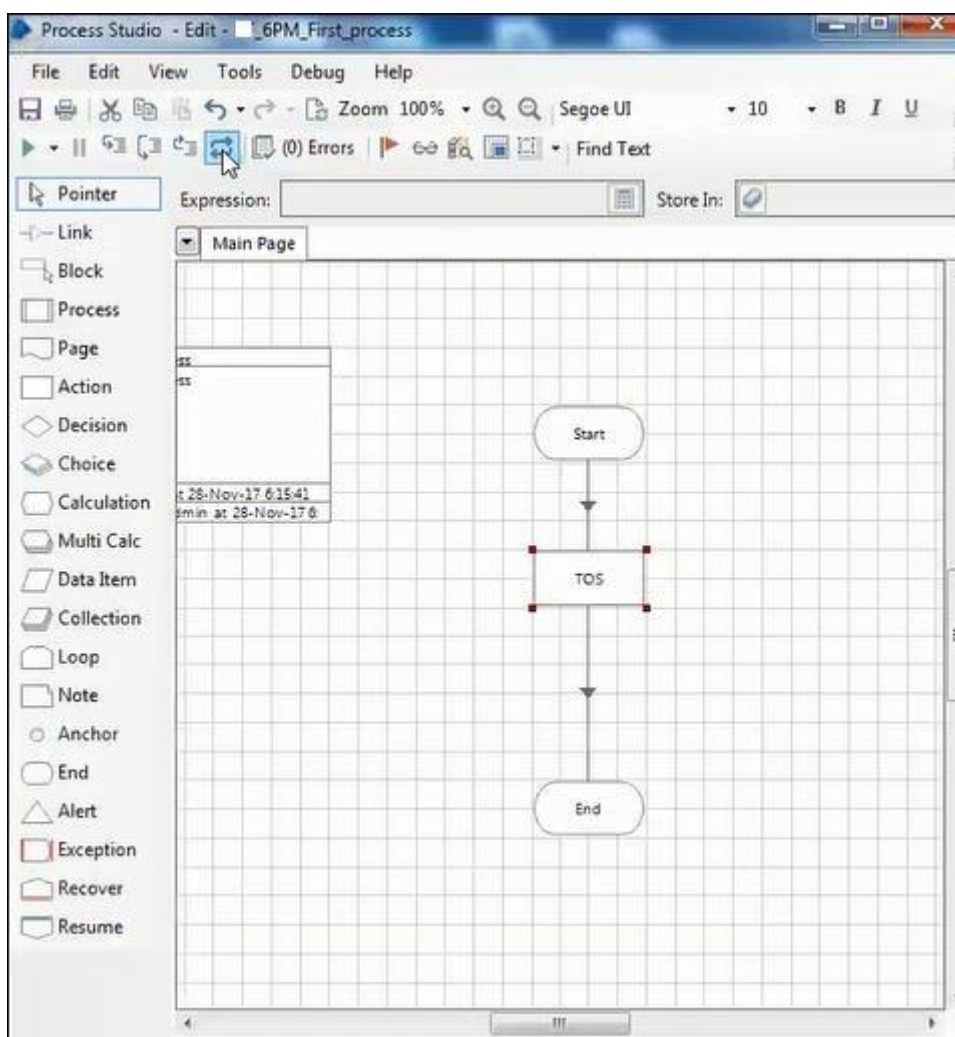
testiympäristöön testattavaksi ennen tuotantoon siirtoa. Jokaiseen prosessiin luodaan työjono, sekä muut ympäristömuuttujat System välilehdellä. System välilehdellä luodaan prosessikohtaiset käyttäjätunnukset, joilla kirjaudutaan kohde järjestelmiin. Lisäksi prosessissa käytettävien tiedostojen sijainnit lisätään omiksi muuttujiksi. Muuttujia on helpompi hallita ja ylläpitää ilman, että tarvitsee itse kohde prosessiin kajota.



Kuva 6. Blue Prism käyttöliittymä

### 3.3 Process Studio

Blue Prism:in Process Studio (kuva 7) on graafinen käyttöliittymä, jonka avulla luodaan, muokataan ja hallitaan prosessia. Prosessit suunnitellaan ja testataan prosessialueella vuokaaviokomponenttien avulla. Piirustusobjektilla voidaan piirtää kokonainen prosessi tai vaihtoehtoisesti prosessi voidaan rakentaa vetämällä ja pudottamalla objekteja käyttöliittymässä. Process Studion prosessialue on tehty ruudulliseksi. Ruudukon avulla helpotetaan objektien kohdistamista prosessin sisällä (Blue Prism docs, 2022.)



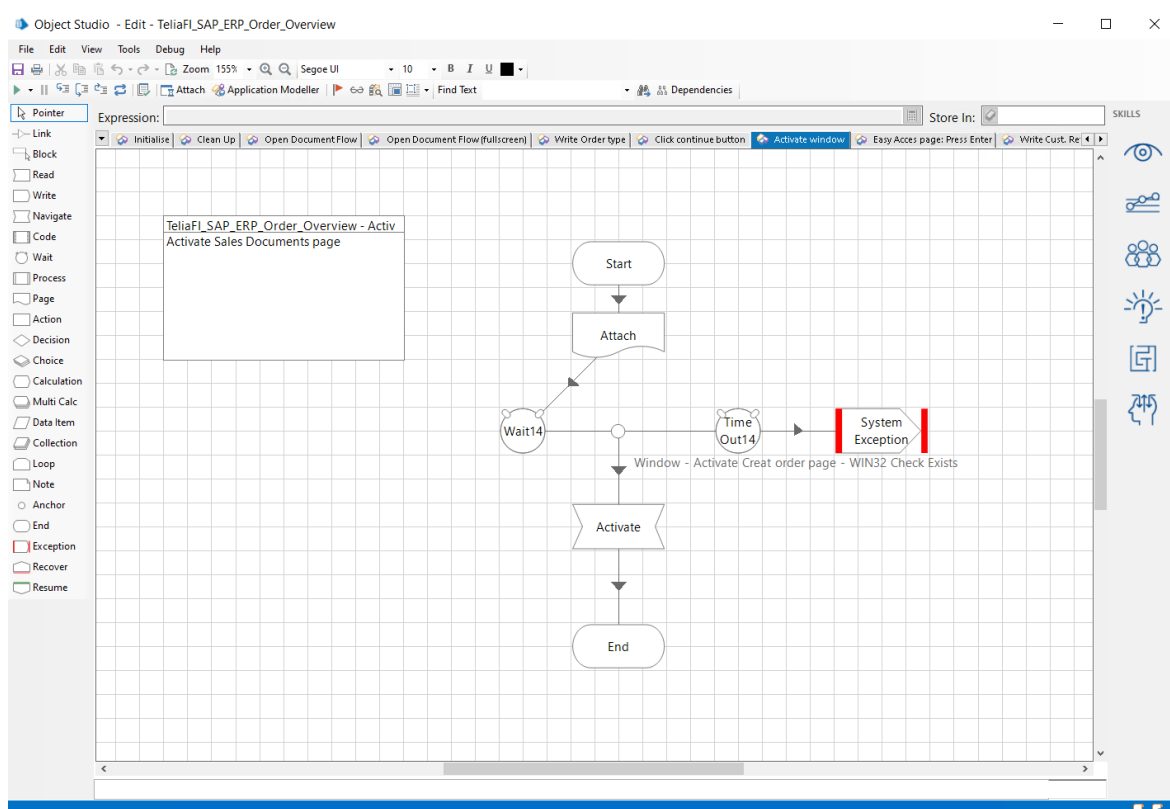
Kuva 7. Process Studio (Blue Prism - Process Studio, 2020)

### 3.4 Objektit

Objektit rakennetaan Blue Prism Object Studiolla, joka on ulkoasultaan ja toiminnoltaan samankaltainen kuin Process Studio. Toiminnot rakennetaan kaavioon kuten Process Studiassa. Näiden kahden Studion erona on se, että Object Studiassa eri sivut eivät ole vuorovaikutuksessa keskenään. Kun sivu ajetaan, se ajaa vain nykyistä sivua, kun taas Process Studiassa sivu ajetaan, se ajetaan pääsivun kautta riippumatta siitä, millä sivulla käyttäjä on.

Objektit ovat valmiita automatisointi kokonaisuuksia ja riippuen automaatiosta, niitä joudutaan muokkaamaan haluttuun muotoon tai tehdä kokonaan uusia. Objektien sisälle luodaan robotille olennaiset toiminnot suorittaakseen halutut toimenpiteet prosessin puolella. Toiminnot, joita objektien sisään rakennetaan koostuvat yleensä yhdestä robotin toiminnasta, esimerkiksi järjestelmään kirjautumisesta. Kuitenkin yhdessä objektissa voi olla useita eri toimintoja ja ne rakennetaan objektissa omille sivuille. Kuvassa 8 on esitetty Blue Prism

objektin näkymä ja siihen rakennettu toiminnallisuus. Objektiin on rakennettu järjestelmän aktivointi. Objektissa ensimmäiseksi kuljetaan Attach -sivun läpi, jolla varmistetaan se, että ollaan yhteydessä kohdesovelluksen kanssa. Tämän jälkeen odotetaan kohdesovelluksen tunnistamista. Tunnistamisen jälkeen edetään Activate kohtaan ja aktivoidaan sovellus. Aktivoinnin jälkeen objekti on ajettu loppuun, ja kohdesovellus on siirtynyt työpöydällä päällimmäiseksi.

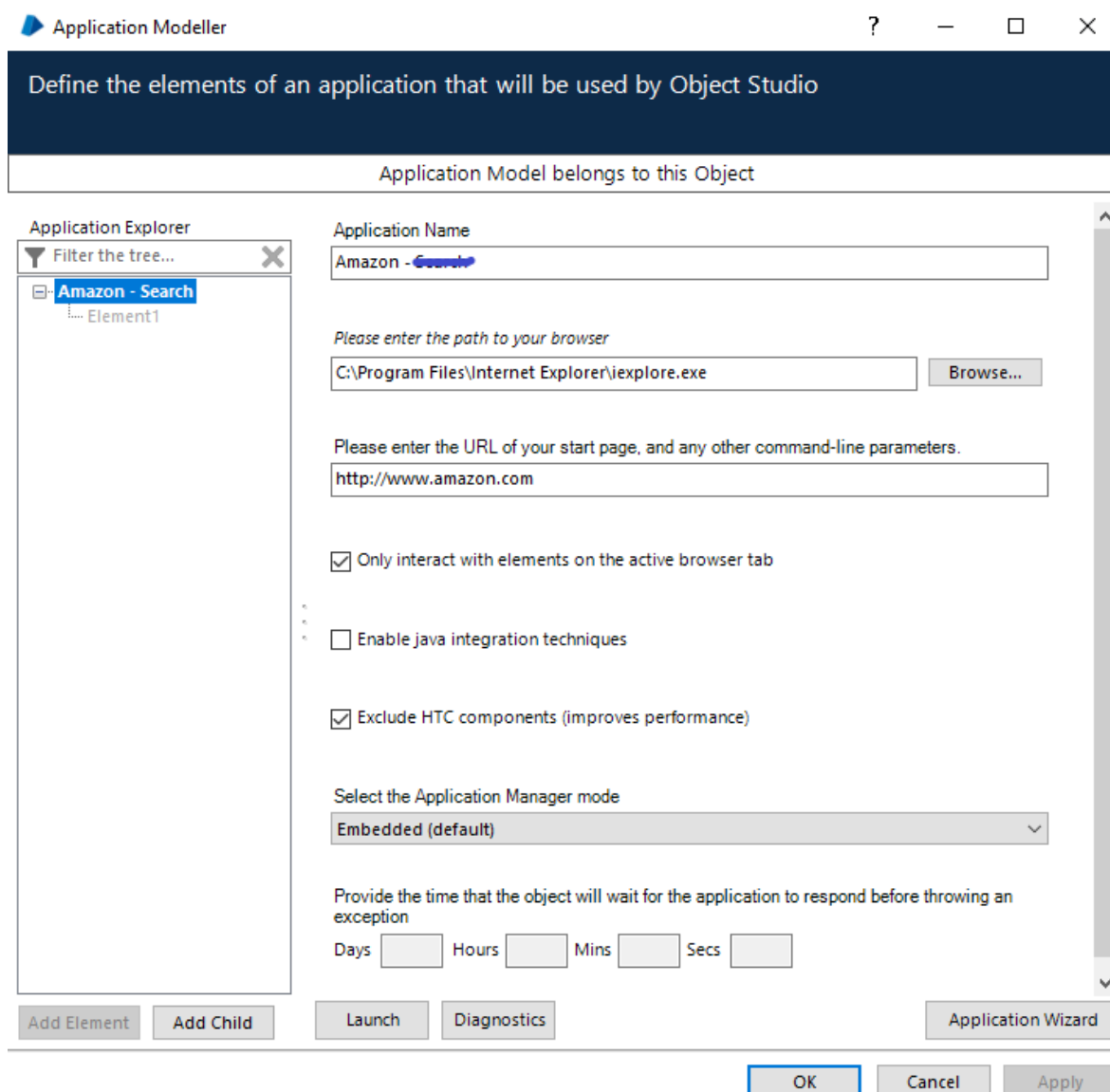


Kuva 8. Blue Prism objekti

### 3.5 Application modeller

Blue Prismin Object Studiosta löytyvä Application modeller tarjoaa käyttäjälleen mahdollisuuden ajaa Blue Prismiä ja kohdesovellusta rinnakkain. Application modellerin avulla kehitetään kohdesovelluksen ominaisuuksia. Sillä avataan kohdesovellus ja etsitään haluttuja tietoja tai painikkeita screen scraping menetelmällä. Application modeller:iin voidaan lisätä elementtejä, joita käytetään hyödyksi Objekteja rakentaessa.

Uuden objektin luonnissa tarvitaan application modellerille alustus, että se osaa löytää halutun kohde sovelluksen mitä ruvetaan kehittämään. Kuvassa 9 on esitetty Application modellerin aloitusnäkymä, johon syötetään kohdesovelluksen tarvittavat tiedot. Lisäksi tarvitaan halutun verkkoselaimen URL, jotta objekti osaa sen käynnistää. Kuvassa näkyvään Element1 kohdalle aloitetaan kehittämään ominaisuuksia logiikka kerrallaan.



Kuva 9. Application modeller

Application modellerin ominaisuuksia kehitetään Blue Prism:in omalla vakoiluohjelmalla, joka vastaa näytön kaapimista. Vakoiluohjelmalla voidaan lukea kohde sovellusta erilaisilla metodeilla. Metodit, joita vakoiluohjelma tarjoaa ovat:

- HTML Mode. Kun halutaan tunnistaa verkkoselainpohjaisen sovelluksen elementtejä.
- Win32 Mode. Kun vakoillaan Windows-sovellusta tai Web-pohjaisten sovellusten ponnahdusikkunoita.
- Accessibility Mode. Kun valitaan tietyn tyyppisiä elementtejä, kuten valintaruutuja tai vaihtopainikkeita.
- UIA Mode. Kun etsitään käyttöliittymästä monimutkaisia elementtejä, eikä mikään muu metodi toimi.

- Region Mode. Kun vakoillaan elementtejä Citrix-ympäristössä.

### 3.6 Työjono

Blue Prism tarjoaa mekanismin robotin työkohteiden jonotuksen, lukituksen ja raportoinnin mallintamiseen. Työjono sisältää useita kohteita ja niiden käsittely tapahtuu työjono -objektin avulla. Objekti tarjoaa lukuisia toimintoja kohteiden käsittelemiseen, kuten kohteen ympärillä olevien metatietojen muuttamiseen.

Työjono lukitsee kohteen, jota se käsittelee, jottei kohdetta pystytä käsittelemään samaan aikaan muualla. Kohde vapautetaan heti kun työ on tehty. Työjonot voivat olla minkä kokoisia tahansa ja ne luodaan Blue Prism käyttöliittymän System Manager välilehdellä. Jononhallinta tapahtuu Control Room välilehdellä. Control Roomissa pystytään keskeyttämään ja jatkamaan jonoja, valvoa jonossa olevia työkohteita, merkitä niitä poikkeuksilla ja poistaa ne kokonaan jonosta (Blue Prism docs, 2022.)

### 3.7 Poikkeuskäsittely

Virheitä ja poikkeuksia tapahtuu prosessin aikana useista syistä. Syitä voi olla esimerkiksi:

- Virhe prosessissa. (esimerkiksi virheellisesti luotu lauseke)
- Objektiin mallinnettu elementti ei täsmää. (esimerkiksi robotti ei löydä painiketta mitä painaa)
- Tuntematon datatyyppi mitä robotti ei osaa käsitellä.

Robotin kehitysvaiheessa prosessiin rakennetaan poikkeuskäsittely sen eri vaiheisiin. Poikkeuksen sattuessa prosessi pysähtyy ja se merkitään lopetetuksi, jos kyseistä poikkeusta ei ole otettu huomioon jo kehityksessä. Poikkeuskäsittelyllä varmistetaan, että mahdolliset ongelmat voidaan korjata automaattisesti tai, että robotin ylläpitäjät pystyvät helposti tunnistamaan ongelman ja siirtämään sen tarvittaessa ihmisen käsiteltäväksi. Jos poikkeukset jätetään käsittelemättä, ne heikentävät robotin toimintaa huomattavasti.

## 4 Prosessin käsittely ohjelmistorobotiikalla

### 4.1 Työn aloitus

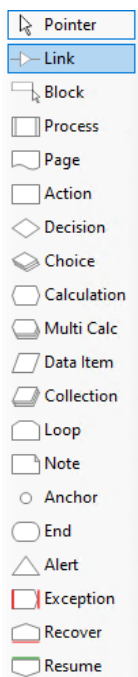
Tässä opinnäytetyössä hyödynnetään ohjelmistorobotiikkaa Telian 4G mobiililaitte tilauksien käsittelyssä. Robotille mallinnetaan toiminnot kahteen eri järjestelmään, luodaan automatisointi logiikka, sekä käytetään hyväksi API rajapintaa uusien tilausten hakemisessa.

Ennen työn aloitusta tutustutaan analyytikon luomaan dokumenttiin, jossa kerrotaan prosessi kokonaisuudessaan ja sen eri vaiheet. Analyytikko tekee dokumenttiin vuokaavion, josta selviää mitä roboti tekee missäkin vaiheessa. Dokumentista löytyy lisäksi kirjallinen ja kuvallinen kuvaus robotin toiminnasta.

Työn aloitus tapahtuu prosessissa käytettävien järjestelmien tutustumiseen ja ottamalla yhteyttä ihmiseen, joka tällä hetkellä suorittaa työn manuaalisesti. Otetaan myös selvää siitä, että onko Teliällä aiemmin tehtyä robotiikka työssä käytettäville järjestelmille. Tämä helpottaa kehitystyötä, jos objektit ovat jo valmiina.

### 4.2 Projektissa käytettävät menetelmät

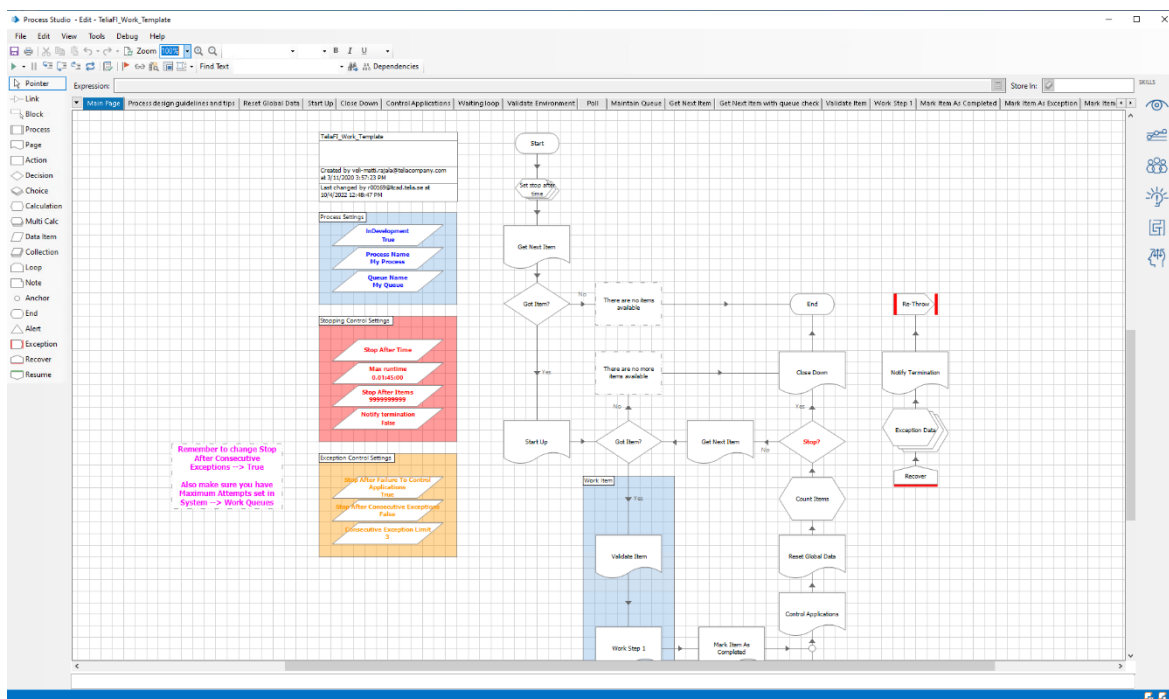
Tässä opinnäytetyössä käytettäviä menetelmiä on runsaasti. Työ vaatii objektien luontia ja kahden eri järjestelmän mallinnusta robotille. Kuvassa 10 on kuvattu Process Studion sivupaneeli, josta löytyy kaikki Blue Prismin logiikka komponentit. Logiikka komponenteilla rakennetaan robotille työnkulku. Komponentit löytyvät Blue Prism:in Studion sivun vasemmasta laidasta. Sivupaneelin Action -komponentti on objekti, joka on valmis toiminnallisuus. Action -komponentti voi sisältää esimerkiksi Excel tiedoston lukemista tai sähköpostin käsittelyä. Objektit luodaan prosessiin erikseen. Blue Prism tarjoaa omia objekteja runsaasti, jos työnkulun kannalta haluttua objektia ei löydy, niin niitä voi itse luoda haluamallaan tavalla.



Kuva 10. Blue Prism logiikka komponentit

### 4.3 Prosessin luonti

Robotin työnkulun rakentaminen aloitetaan luomalla uusi prosessi Blue Prism:in Studiassa. Prosessille luodaan uniikki, työhön soveltuva nimi ja käytetään hyväksi toimeksiantajan valmista työpohjaa. Työpohja sisältää robotille valmiin työnkulun, jota muokataan työhön sopivaksi. Kuvassa 11 on esitetty valmis työpohja, joka sisältää robotille välttämättömät työvaiheet.



Kuva 11. Blue Prism prosessin työohja

Työohja sisältää useita sivuja, joihin rakennetaan työvaiheiden logiikka. Työnkulun kannalta tärkein sivu on Main page eli pääsivu. Pääsivulle linkitetään kaikki tarpeelliset sivut toteuttamaan robotin toimintaa. Prosessit rakennetaan komponentteja vetämällä ja pudottamalla Studion prosessialueelle. Prosessialue löytyy Studion keskiosasta, joka on ruudullinen.

#### 4.4 Järjestelmien mallinnus

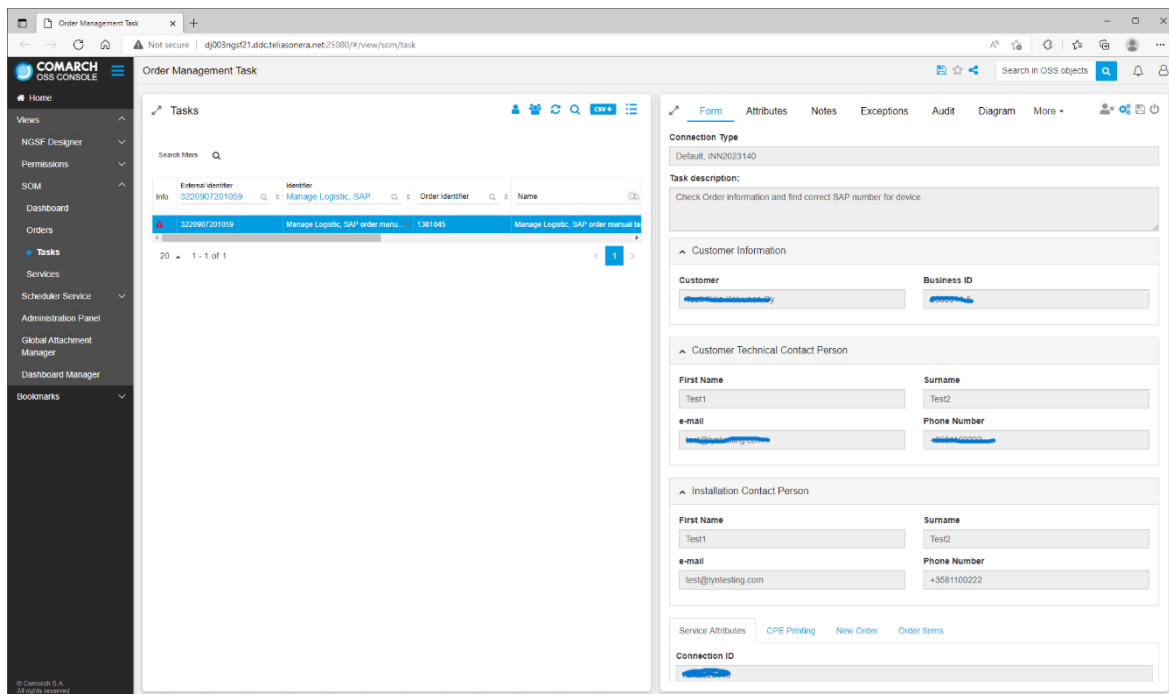
Tilauksen käsittely tapahtuu SAP ERP- ja NGSF-järjestelmässä. Robotin toiminta järjestelmissä perustuu mallinnukseen, jotta robotti osaa käsitellä ja lukea haluttuja tietoja. Järjestelmien mallintaminen tehdään objektin sisälle, yksi toiminto kerrallaan. Tällä tavoin saadaan prosessista selkeä ja helposti luettava. Jokaisessa objektissa käytetään Application modeller:ia, jossa hyödynnetään screen scraping -menetelmää eli kaapataan näytöllä näkyviä tietoja.

Application modeller:in avulla voidaan ajaa Blue Prism:ia ja kohdesovellusta rinnakkain, ja sen avulla opetetaan objektin sisällä kohdesovelluksen ominaisuuksia osoittamalla ja klikkaamalla (Blue Prism docs, 2022.)

##### 4.4.1 NGSF

Laitetilauksia pystytään hallinnoimaan NGSF-järjestelmässä. Kun uusi laite tilataan Telialta, syntyy Telian omaan tilausjärjestelmään tieto, jonka kautta siitä syntyy uusi tehtävä NGSF

järjestelmään. Robotin tarkoituksena on käsitellä näitä tehtäviä. Kuvassa 12 on näytetty NGSF-järjestelmän päänäkymä. Päänäkymän keskiosassa näkyy robotin käsittelyssä oleva tehtävä ja sivun oikealla puolella näkyy asiakkaan yhteystiedot, mitkä robotti poimii talteen jatkokäsittelyä varten.



Kuva 12. NGSF tehtävän näkymä

NGSF-järjestelmän kautta robotti merkitsee syntyneet keräilytehtävät itselleen, jotta ihminen ei kykene samaa tehtävää käsittelemään samanaikaisesti. Järjestelmään syntyy kaikista uusista laitetilauksista tehtävä, jonka robotti ottaa käsittelyyn. Jokaisesta tehtävästä löytyy asiakkaan yhteystiedot ja lähetettävän laitteen tiedot. Kun tehtävästä on luotu SAP ERP toiminnanohjausjärjestelmän kautta keräilytilaus, syötetään siitä syntynyt tilausnumero NGSF:n tehtävälle ja suljetaan tilaus.

#### 4.4.2 SAP ERP

SAP ERP on toiminnanohjausjärjestelmä, jossa on kaikki tarvittavat työkalut varastotoimintojen suorittamiseen. SAP ERP:in avulla voidaan seurata ja hallita varastosaldoja sekä luoda varastolta lähteviä tilauksia. Tässä työssä käsiteltävien mobiililaitetilauksien keräilynumero sekä SAP tilausnumero luodaan SAP ERP järjestelmässä. Keräilynumeroa ja SAP tilausnumeroa tarvitaan, kun laitetilauksen työtehtävä saatetaan NGSF:ssä loppuun. Kuvassa 13 on esitetty SAP ERP -järjestelmän käyttöliittymä ja tietokentät mitkä täytetään tilaustehtävästä saatujen tietojen perusteella.

Kuva 13. SAP käyttöliittymän tilauspohja

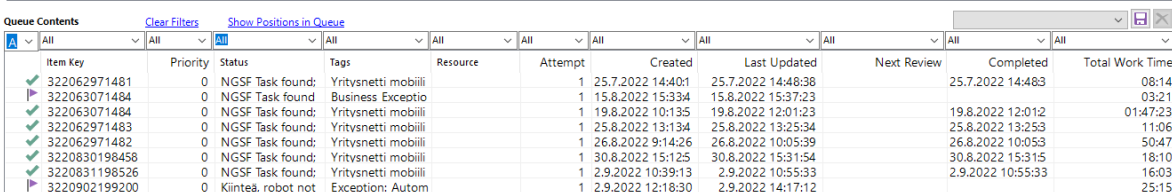
#### 4.5 Tilausten hakeminen ja työjonon täyttö

Robotin työprosessin alussa haetaan uudet laitetilaukset API rajapinnalla Telian tilausjärjestelmästä. Uusista tilauksista kerätään kaikki tarvittavat tiedot jatkokäsittelyä varten, kuten esimerkiksi laitteen nimi sekä laitteen tilausnumero. Kuvassa 14 on näytetty logiikka prosessista, jossa haetaan uusien laitetilauksien tiedot.



Uudet laitetilaukset siirretään robotin omaan työjonoon, josta robotti kerää ne yksi kerrallaan käsiteltäväksi. Työjonossa olevat kohteet merkitään aina uniikilla työhön liittyvällä tiedolla, tässä prosessi merkitään työkohteet työn alussa haetulla laitteen tilausnumerolla. Työkohteen tilaa voidaan seurata jonon vasemmasta laidasta löytyvän kuvakkeen avulla. Valmiit työt merkitään vihreällä ”oikein” merkillä, virheelliset työt ”lippu” -merkillä ja keskeneräiset harmailla palloilla. Kuten kuvasta 15 käy ilmi, työjonosta löytyy muitakin oleellisia tietoja työkohteen tilasta ja työn käsittelyyn käytetystä ajasta.

Yksi työkohte vastaa yhtä mobiililaitetilausta. Työkohteiden merkitsemisellä saadaan robotille selkeä työjono, jota voidaan ylläpidon toimestakin helposti hallita. Mahdollisten virhetilanteiden sattuessa pystytään paikallistamaan, minkä laitetilauksen kohdalla virhe on sattunut.



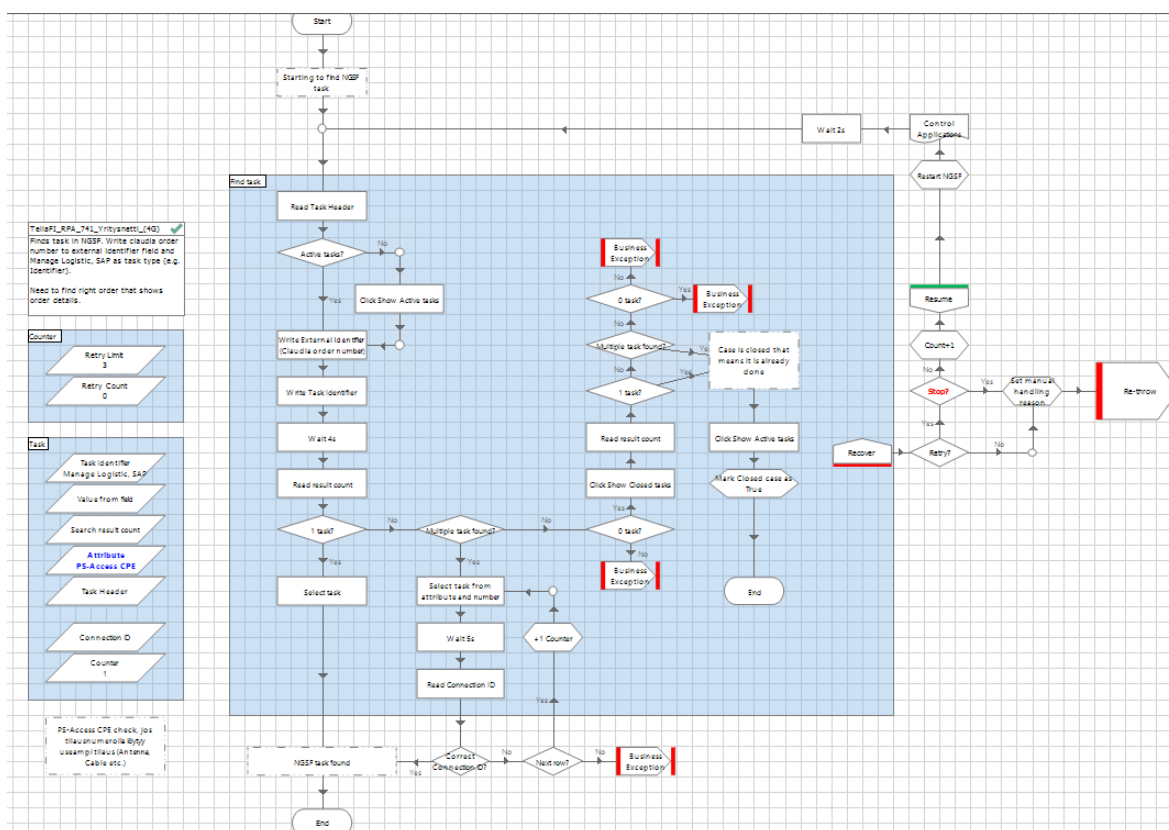
Item Key	Priority	Status	Tags	Resource	Attempt	Created	Last Updated	Next Review	Completed	Total Work Time
322062971481	0	NGSF Task found:	Yritysnetti mobiili		1	25.7.2022 14:40:1	25.7.2022 14:48:38		25.7.2022 14:48:3	08:14
322063071484	0	NGSF Task found:	Business Exceptio		1	15.8.2022 15:33:4	15.8.2022 15:37:23			03:21
322063071484	0	NGSF Task found:	Yritysnetti mobiili		1	19.8.2022 10:13:5	19.8.2022 12:01:23		19.8.2022 12:01:2	01:47:23
322062971483	0	NGSF Task found:	Yritysnetti mobiili		1	25.8.2022 13:13:4	25.8.2022 13:25:34		25.8.2022 13:25:3	11:06
322062971482	0	NGSF Task found:	Yritysnetti mobiili		1	26.8.2022 9:14:26	26.8.2022 10:05:39			50:47
3220830198458	0	NGSF Task found:	Yritysnetti mobiili		1	30.8.2022 15:12:5	30.8.2022 15:31:54		30.8.2022 15:31:5	18:10
3220831198526	0	NGSF Task found:	Yritysnetti mobiili		1	2.9.2022 10:39:13	2.9.2022 10:55:33		2.9.2022 10:55:33	16:03
3220902199200	0	Kiinteä, robot not	Exception: Autom		1	2.9.2022 12:18:30	2.9.2022 14:17:12			25:15

Kuva 15. Blue Prism prosessin työjono

#### 4.6 Tilautustietojen hakeminen

Tilauksesta syntyy NGSF-järjestelmään tehtävä, josta robotti hakee kaikki tarvittavat tiedot myöhempää käsittelyä varten. Tehtävästä kerätään tilauksen luonti- ja eräpäivä, laitemallin numero, asiakkaan yhteystiedot sekä asennusosoite. Luonti- ja eräpäivän mukaan robotti käsittelee tilauksen. Jos tilauksen luontipäivä on kello 14:00 jälkeen, robotti käsittelee tilauksen vasta seuraavana päivänä. Joissakin tilauksissa eräpäivä on vasta kuukausien päässä, nämä tilaukset robotti siirtää takaisin työjonoon ja käsittelee ne vasta silloin, kun tilauksen eräpäivään on 2 viikkoa jäljellä.

Prosessin logiikka tehtävien hakemisessa aloitetaan hakemalla oikea tehtävä oikealle tilaukselle. Työvaiheen ensimmäisessä osiossa robotti syöttää Telian tilausjärjestelmästä noudetun tilausnumeron NGSF:n tunniste kenttään. Tunniste kentän kautta löydetään oikea työ robotille käsiteltäväksi. Kun tehtävä löydetään, robotti poimii NGSF:stä kaikki tiedot, mitä tarvitaan jatkokäsittelyä varten. Yhdessä tehtävässä voi olla useampi laitetilaus ja jokaisesta laitteesta on tehtävä erikseen tilaukset SAP ERP- järjestelmään. Robotille rakennetaan poikkeuskäsittely, jos yhden tilausnumeron alla on useampi tehtävä, niin robotti osaa valita oikean tehtävän liittymätunnuksen mukaan. Tietojen poimimisen jälkeen robotti kirjaa työn aloitetuksi ja siitä syntyy merkintä tehtävälle, että sitä ei voi ihminen käsitellä samaan aikaan. Kuvassa 15 on kuvattu robotin logiikka tehtävän etsimiseen NGSF-järjestelmässä.



Kuva 16. Tehtävien haku NGSF-järjestelmästä

Tehtävästä saadulla laitemallin numerolla robotti tarkistaa sen, että käsiteltävä tilaus sisältää oikean laitteen. Robotti käsittelee vain 4G mobiililaitetilauksia, kaikki muut laitteet merkitään työjonossa "Out of Scope" -tagilla ja merkitään työkohde valmiiksi. Robotti siirtyy seuraavaan työkohteeseen, kun laitetilaus on jotain muuta kuin 4G laite. Ihminen käsittelee muut laitetilaukset, mutta tulevaisuudessa on tarkoitus kehittää robottia niin, että se kykenee käsittelemään kaikkia laitetilauksia laitteesta riippumatta.

Tehtävästä kerättävät asiakkaan yhteystiedot sekä asennustiedot tarvitaan SAP ERP -järjestelmässä tilauksen luomisessa. Yksi asiakas voi tilata useita laitteita eri osoitteisiin ja näistä kaikista robotti tekee omat tilauksensa SAP ERP:iin. Tehtävä pystytään sulkemaan vasta silloin, kun tehtävästä on saatu keräilytilaus SAP ERP:stä. Tilausnumero syötetään NGSF:n tehtävän tilausriville ja tämän jälkeen tehtävä pystytään sulkemaan.

#### 4.7 Tilauksen luonti

Tilauksen luonti tapahtuu SAP ERP:ssä. Mobiililaitetilauksen luomisessa käytetään SAP ERP- toiminnanohjausjärjestelmää. Järjestelmään syötetään kaikki tiedot mitä robotti on aiemmissa työvaiheissa kerännyt. Tilaukseen liittyvät tiedot syötetään järjestelmään, josta

syntyy tilaus- sekä keräilynumero jatkokäsittelyä varten. Syntynyt tilausnumero syötetään NGSF järjestelmään, jossa tilaustehtävä saatetaan loppuun. Keräilynumeron syntyminen luo varastoon uuden keräilytehtävän ihmiselle kerättäväksi.

Robotille rakennetaan logiikka siten, että prosessiin lisätään uusi sivu, johon rakennetaan kaikki toiminnallisuudet, mitä robotti tarvitsee osatakseen luoda uusia tilauksia SAP ERP-järjestelmään. Kun uusia sivuja luodaan ja myöhemmin lisätään pääsivulle, niin on tärkeää prosessin alussa tarkistaa, että onko sen hetkinen työkohde jo mennyt kyseisen logiikka sivun läpi. Näin vältetään tupla töiltä, jos robotille sattuu virhe työkohteessa, niin se osaa jatkaa työkohdetta virheen jälkeen prosessin oikeasta kohdasta eli siitä mihin se on viimeksi jäänyt. Robotille rakennettava logiikka on suoraviivaista, yksi tietokenttä kerrallaan täyttämistä aiemmin haetuilla tiedoilla.





varastosaldoista, jolloin keräilynumeroa ei synny ollenkaan, eikä tilausta tässä tapauksessa voida tehdä. Kun oikeat numerot saadaan syötettyä NGSF:ään, robotti tallentaa ja sulkee tehtävän. Tallentamisen jälkeen robotti siirtyy seuraavaan työkohteeseen, jos seuraavaa työkohdetta ei ole, robotti sulkee kaikki järjestelmät. Robotti käy aina puolen tunnin välein prosessin läpi etsien uusia mobiililaitte tilauksia.

#### 4.9 Manuaalikäsittely

Prosessin viimeisessä vaiheessa rakennetaan manuaalikäsittely logiikka, jossa robotti lähettää sähköpostin kaikista virheellisistä ajokerroista. Virheitä voi tulla prosessissa missä kohtaan tahansa. Prosessin yleisimmät virheet ovat kohdesovelluksien kaatumiset tai puutteelliset tilaustiedot kohdejärjestelmässä.

Robotin logiikka rakennetaan siten, että virheen sattuessa robotti sulkee ja käynnistää kaikki järjestelmät uudestaan ja yrittää jatkaa prosessia siitä mihin se on jäänyt. Prosessissa tätä yritetään tehdä kolme kertaa, jos kolmannella kerralla kohdataan sama virhe, niin robotti siirtää käsittelyssä olevan työn virheenkäsittelyyn. Virheenkäsittely vaiheessa robotti siirtää työn puolen tunnin päähän uudestaan käsiteltäväksi. Tämä tehdään siksi, että kohdesovellukset voivat olla sillä hetkellä ylikuormitettuna, eikä robotti kykene poimimaan tarvittavia tietoja ajallaan.

Virheenkäsittelyn kautta siirretty työ käsitellään vielä kerran uudelleen ja jos kohdataan taas sama virhe prosessissa, niin robotti siirtää työn lopetettavaksi ja lähettää virheellisestä työstä sähköpostin manuaalikäsittelijälle. Sähköposti sisältää tiedot virheellisestä käsittelystä tilauksesta sekä virheeseen johtaneet syyt. Sähköposti lähetetään Postin työntekijälle, joka ottaa kyseisen mobiililaitte tilauksen manuaalisesti käsittelyyn.

## 5 Yhteenveto ja pohdinta

Tämän opinnäytetyön tavoitteena oli rakentaa ohjelmistorobotiikalla prosessi, jolla käsitellään 4G mobiililaitetilauksia. Robotille rakennettu logiikka oli melko suoraviivaista, tietojen siirtämistä järjestelmästä toiseen. Iso haaste työssä oli rakentaa API rajapinnalla Query kutsuja, joiden mukaan laitteesta saadaan halutut tiedot robotin käsiteltäväksi. Telian tilausjärjestelmästä on saatavilla niin paljon tietoa laitteista, että niiden karsiminen ja oikeiden tietojen löytäminen oli työlästä.

Prosessia rakentaessa opin ohjelmistorobotiikasta paljon, koska tässä työssä käsitellään laajasti ohjelmistorobotiikan eri toimintoja. Laajuutta kuvastaa hyvin se, että työssä käsitellään Exceliä, eri järjestelmiä sekä käytetään hyväksi API rajapintaa.

Tällä hetkellä prosessi käsittelee vain tiettyjä laitetilauksia, tulevaisuudessa olisi tarkoitus saada Telian kaikki lähtevät laitteet kulkemaan saman prosessin kautta. Robotti on toiminut muutaman viikon tuotannossa ja pääosin se on toiminut juuri niin kuin pitää. Prosessin ensimmäisten ajokertojen jälkeen robotissa huomattiin virhe. Virhe koski tilauksia, jossa käsiteltiin useita laitteita. Virheenkorjauksen jälkeen logiikka toimii tällä hetkellä niin kuin pitääkin. Muutaman viikon toimintajakson aikana yksi ongelma robotin toiminnassa on ollut järjestelmien toiminnallisuus. Järjestelmät eivät aina toimi niin kuin pitää ja töitä valuu muutama kerran viikossa manuaalikäsittelyyn.

Tämän työn alkuperäisenä tarkoituksena oli saada suurimmat laitetilaukset ohjelmistorobotiikan käsiteltäväksi, koska ne ovat kaikista helpoiten automatisoitavissa. Muiden laitteiden kohdalla täytyy ottaa huomioon oheistuotteiden määrät, sekä täytyy tarkistaa, pystyykö asennusosoitteeseen uusimpia laitteita asentamaan ilman mittavia kustannuksia.

Työn suurimpana haasteena ja aikaa vievin osuus on ollut sidosryhmäläisten kanssa työskentely. Työn alkumetreillä huomattiin puutteita prosessin dokumentaatioissa, jonka mukaan prosessia aloitetaan kehittämään. Tämä johtui siitä, että prosessin dokumentaatio oli tehty kauan aikaa sitten ja järjestelmiin oli kerennyt tulemaan päivityksiä. Päivitysten takia SAP ERP-toiminnanohjausjärjestelmä oli hieman muuttunut alkuperäisestä ja se vaati uudelleen dokumentointia. SAP ERP-järjestelmän mallintaminen oli myös yksi työn haasteista. Järjestelmä oli vaikea mallinnettava sen käyttöliittymän takia. Kun käyttöliittymässä painetaan hiirellä väärästä kohdasta, se menee lukkoon, eikä työtä voida enää jatkaa. Kaikesta vastoin käymisen jälkeen robotti on kuitenkin saatu vietyä tuotantoon.

## Lähteet

20 Ways Bots Improve Businesses for The Better. 2017. Saatavissa

<https://research.aimultiple.com/top-robotic-process-automation-rpa-benefits/>

12 Popular Benefits of RPA in Business. 2022. Saatavissa

<https://marutitech.com/benefits-of-rpa-in-business/>

Blue Prism is raising \$130M to fuel new technologies. 2019. Saatavissa

<https://www.builtinaustin.com/2019/01/24/blue-prism-130-million-funding>

Blue Prism Docs. Application modeller. 2022. Saatavissa

<https://bpdocs.blueprism.com/bp-6-9/en-us/frmlIntegrationAssistant.htm>

Blue Prism Docs. Process Studio. 2022. Saatavissa

<https://bpdocs.blueprism.com/bp-6-9/en-us/frmProcess.htm>

Blue Prism Docs. Queue Management. 2022. Saatavissa

<https://bpdocs.blueprism.com/bp-6-7/en-us/control-queues.html>

Different Types of RPA. 2019. Saatavissa

<https://www.smartb.co/different-types-of-rpa/>

Everything You Need to Know About Screen Scraping Software. 2017. Saatavissa

<https://www.uipath.com/blog/rpa/screen-scraping-software-everything-you-need-to-know>

Gartnerin tutkimus ohjelmistorobotiikan markkinajohtajista ja -haastajista. 2022.

Saatavissa <https://www.gartner.com/doc/reprints?id=1-2AOPUPBE&ct=220727&st=sb>

George Lawton. 2021. Robotic Process Automation (RPA). Saatavissa

<https://www.techtarget.com/searchcio/definition/RPA>

Life Cycle of RPA. 2022. Saatavissa [https://intellipaat.com/blog/tutorial/rpa-tutorial/rpa-](https://intellipaat.com/blog/tutorial/rpa-tutorial/rpa-lifecycle/)

[lifecycle/](https://intellipaat.com/blog/tutorial/rpa-tutorial/rpa-lifecycle/)

Robotic Process Automation (RPA). 2022. Saatavissa

<https://www.automationanywhere.com/rpa/robotic-process-automation>

RPA Market Size and Popular Vendors in 2022. 2022. Saatavissa

<https://research.aimultiple.com/rpa-market/>

RPA Life Cycle. 2021. Saatavissa <https://www.javatpoint.com/rpa-life-cycle>

Skaler. 2022. Do the Human Stuff. Automate Everything Else. Saatavissa

[https://skaler.siili.com/learn/intelligent-process-automation?qclid=EAlaIQobChMIhrq92KDn-glVVAqiAx0DTwFLEAAYAiAAEgleD\\_D\\_BwE](https://skaler.siili.com/learn/intelligent-process-automation?qclid=EAlaIQobChMIhrq92KDn-glVVAqiAx0DTwFLEAAYAiAAEgleD_D_BwE)

The Evolution of RPA: A 30-Year Journey. 2022. Saatavissa

<https://electroneek.com/rpa/history-of-rpa/>

Top 15 Robotic Process Automation (RPA) Companies. 2020. Saatavissa

<https://www.datamation.com/artificial-intelligence/top-15-robotic-process-automation-rpa-companies>