

LAPPEENRANNAN-LAHDEN TEKNILLINEN YLIOPISTO LUT  
School of Engineering Science  
Tuotantotalous

*Joona Virtanen*

**OHJELMISTOROBOTIIKAN YLLÄPIDON, VALVONNAN JA SKAALAAMISEN  
JÄRJESTÄMINEN**

Diplomityö

Tarkastajat:

Dosentti Jouni Koivuniemi  
TkT Lasse Metso

# TIIVISTELMÄ

Lappeenrannan-Lahden teknillinen yliopisto LUT  
School of Engineering Science  
Tuotantotalouden koulutusohjelma

Joona Virtanen

## Ohjelmistorobotiikan ylläpidon, valvonnan ja skaalaamisen järjestäminen

Diplomityö

2021

76 sivua, 12 kuvaa, 1 taulukko ja 2 liitettä

Tarkastajat: Dosentti Jouni Koivuniemi ja TkT Lasse Metso

Hakusanat: ohjelmistorobotiikka, ohjelmistoylläpito, digitalisaatio, yritysviestintä

Keywords: RPA, software maintenance, digitalization, organizational communication

Työn tarkoituksena on tuottaa toimintamalli päivittäisen ja huomattavan kokoiseksi kasvaneen ohjelmistorobottilaivueen ylläpitoon, valvontaan ja skaalaamiseen. Se käy läpi liiketoiminnan prosessien hallinnan, mallintamisen perusteet ja esittelee muita menetelmiä, jolla näitä saadaan automatisoitua. Siten saada laajempi kuva digitalisaatiosta ja robotiikan roolista siinä.

Käydään läpi robotiikan peruseräpäätteet, miten se toimii, mitkä ovat sen edut, millaisilla työkaluilla robotiikkaa kehitetään ja miten ohjelmistojen ylläpidon kirjallisuus soveltuu robotiikan ylläpitoon. Erityisesti kiinnitetään huomiota ohjelmistorobotiikan skaalaamisen organisaatiossa, josta saadaan selvästi kuva siitä, että teknologian itsensä lisäksi viestinnällä, tietojärjestelmillä ja paljon laajemmalla liiketoiminnan kontekstilla on suuri vaikutus ohjelmistorobotiikan skaalaamiseen toteutumisessa. Yritysviestinnästä esitetään joitakin peruseräpäätteitä, eli mitä se on ja miten se tyypillisesti hierarkkisessa ja osastoihin jakautuneessa organisaatiossa rakentuu. Henkilöstön luoton rakentamista uutta teknologiaa, eli robotiikkaa kohtaan tuodaan vahvasti esille. Luottamuksessa korostuu viestinnän merkitys sidosryhmien välillä, etenkin ongelmatilanteissa.

Ratkaisuehdotus rakentuu niin viestinnän kuin teknologian komponenteista. Viestintää halutaan viedä rakenteelliseksi ja selkeäksi luomalla jaettu viestintäkanava robotiikalle ja ottamalla käyttöön tikettijärjestelmä, johon saadaan kerättyä selkeästi pyydyt muutokset toimenpiteet. Etenkin teknologisten ratkaisukomponenttien osalta hierarkkinen rakenne robottien käyttämille tiedoille sekä robottiin arkkitehtuurille ovat keskeisessä roolissa. Näiden lisäksi muut robotteja rakennetta yhtenäistävät toimenpiteet, kuten etäkonfiguraatiot ja argumenttien käyttö tehostavat ylläpitoa ja mahdollistavat tehokkaan skaalaamisen.

Työn lopputuloksena havaitaan, että etenkin skaalaamisen kannalta vahvempi yhteistyö niin johdon, RPA-tiimin, IT:n kuin loppukäyttäjien, eli liiketoimintayksiköiden, välillä on avainasemassa. Vain lisäämällä ymmärrystä ja kouluttamalla etenkin robotiikan käyttäjiä saadaan aikaan aidosti skaalattava malli, kun valvonta ja muutostarpeiden ennakointi siirtyy heille.

## **ABSTRACT**

Lappeenranta-Lahti University of Technology LUT  
School of Engineering Science  
Degree Programme in Industrial Engineering and Management

Joona Virtanen

### **Organizing the maintenance, supervising, and scaling of software robotics**

Master's thesis

2021

76 pages, 12 figures, 1 table and 2 appendices

Examiners: Docent Jouni Koivuniemi and D.Sc. (Tech.) Lasse Metso.

Keywords: RPA, software maintenance, digitalization, organizational communication

This work aims to produce an operating model for the daily maintenance, orchestration, and scaling of a software robot fleet of a considerable size. It goes through the basics of business process management and modelling and introduces other common methods of business process automation. This gives a wider picture over digitalization and to role of RPA in it.

The basics of robotics are introduced. How does it work, what are the benefits, what are the tools used for development and how does software maintenance literature apply on robotic maintenance? Especial interest is placed over scaling robotics within an organisation and how along with the technology also communication, information systems and far wider business context has a huge effect in its realisation. Some principles of organisational communication are also introduced i.e., what it is and how it typically builds in hierarchical organisations divided into different units. Building the personnel trust towards new technology -meaning software robotics- is placed here in a high role. For trust the role of communication between key shareholders especially in the exception situations is highlighted

The solution is built up from both communicational and technological components. The communication is wanted to be turned more structural and clearer by creating a shared communication channel for robotics and by introducing a ticket system, where clear change requests can be collected. Especially in the technological components of the solution hierarchical structure for the information that robot uses place a key role. Aside from these other unification methods for the robots, such as distant configuration and use of arguments make maintenance more effective and allow efficient scaling.

As the result of the work, it is perceived that especially for the scaling of robotics even stronger cooperation with leadership, RPA development team, and end users meaning business units, plays a key role. Only when understanding on the topic and education is increased especially for the robot users truly scalable model can be built as the orchestration work and predicting changes is moved to them.

## ALKUSANAT

Haluan kiittää perhettä ja ystäviäni. On heidän tuen ansiota, että minulla riitti energiaa tällaisen työn kyhäämiseen. Erityisesti sparrailu ystävän kanssa, joka myös toteutti opinnäytetyötä samaan aikaan, oli suureksi avuksi. Läheisten vuoksi sitä jaksaa yrittää ja tehdä elämässä, eikä vain jäädä sohvalle lojumaan kattoa tuijottamaan.

Haluan kiittää yliopistoani, jossa vietin semmoiset nelisen lukuvuotta. Ei se opiskelu niin ylivoimaiseksi osoittautunut, kun keskittyi enemmän määrään kuin laatuun. Sanovat, että paras on hyvän pahin vihollinen. Missään muualla en voisi kuvitella näiden opiskelun vuosien sujuneet paremmin kuin Skinnarilan vappaavaltiossa. Kiitos työn ohjaajille, jotka varmistivat, että myös jonkinlaiset laatukriteerit täyttyivät ja joilta sain arvokkaita ehdotuksia tutkimuksen toteuttamiseen.

Lisäksi kiitän työn kohdeyritystä, tiimiäni ja minut rekrynyttä esimiestä. En saanut pelkästään mahdollisuutta päästä tekemään diplomityötä, vaan lisäksi aloittamaan urani juuri opintojen mukaisella alalla yhdistäen IT:n mahdollistamaa teknologiaa ja liiketoimintaa hyvin käytännöllisellä tavalla huippujoukon kanssa. Koska työni on käytännönläheinen toteutus kohdeyrityksessä, on kollegoideni pohjatyöllä ja ymmärryksellä aiheesta ollut suuri vaikutus.

Kiitos ja Anteeksi

16.6.2021

Joona Virtanen

# SISÄLLYSLUETTELO

|     |  |    |
|-----|--|----|
| 1   | JOHDANTO .....                             | 3  |
| 1.1 | Työn tausta.....                           | 3  |
| 1.2 | Tavoitteet ja rajaus .....                 | 4  |
| 1.3 | Tutkimuksen toteutus.....                  | 5  |
| 1.4 | Raportin rakenne .....                     | 6  |
| 2   | LIIKETOIMINNAN PROSESSIT .....             | 9  |
| 2.1 | Prosessien mallintaminen ja hallinta.....  | 9  |
| 2.2 | Prosessien automaatio .....                | 11 |
| 3   | OHJELMISTOROBOTIIKKA .....                 | 14 |
| 3.1 | Ohjelmistorobotiikan perusteet .....       | 14 |
| 3.2 | Ohjelmistorobotiikan hyödyt .....          | 16 |
| 3.3 | Ohjelmistorobotiikan kehitys.....          | 18 |
| 3.4 | Ohjelmistorobotiikan ylläpito .....        | 20 |
| 3.5 | Ohjelmistorobotiikan skaalaus.....         | 24 |
| 4   | KOMMUNIKAATIO ROBOTIIKASTA.....            | 32 |
| 5   | ROBOTIIKAN ARVOT .....                     | 37 |
| 6   | VIESTINTÄMALLI OHJELMISTOROBOTIIKKAAN..... | 40 |
| 6.1 | Lähtötilanne .....                         | 40 |
| 6.2 | Ensimmäinen ratkaisuehdotus .....          | 40 |
| 6.3 | Palautteen kerääminen .....                | 44 |
| 6.4 | Parannettu ehdotus .....                   | 45 |
| 7   | TEKNINEN MALLI.....                        | 47 |
| 7.1 | Lähtötilanne .....                         | 47 |

|     |   |    |
|-----|---|----|
| 7.2 | Ensimmäinen ratkaisuehdotus .....                       | 48 |
| 7.3 | Palautteen kerääminen .....                             | 52 |
| 7.4 | Parannettu ehdotus .....                                | 55 |
| 8   | QFD ANALYYSI ROBOTIIKAN ARVOISTA JA TOTEUTUKSESTA ..... | 57 |
| 9   | TULOKSET JA NIIDEN ARVIOINTI .....                      | 60 |
| 9.1 | Työn keskeiset tulokset.....                            | 60 |
| 9.2 | Tulosten arviointi .....                                | 62 |
| 9.3 | Jatkotoimenpiteet ja suositukset .....                  | 63 |
| 10  | YHTEENVETO .....  | 64 |

Lähteet

Liite/Liitteet

## TYÖSSÄ KÄYTETYT TERMIT JA LYHENTEET

|              |   |
|--------------|---|
| RPA          | Robotic Process Automation, ohjelmistorobottiikka. Käyttää botteja automatisoimaan rutiinitehtäviä ohjelmistosovelluksissa (G2 2020).                                 |
| REF          | Robotic Enterprise Framework. Tiloihin perustuva projektipohja, joka mahdollistaa mm. parhaat logi, virnehallinta ja sovelluksen käynnistys käytännöt (UiPath 2021e). |
| Attended     | Robottiityppi. Auttaa tiettyä käyttäjää tehtävissä samalla koneella kuin käyttäjä (UiPath 2021c).   |
| Unattended   | Robottiityppi. Ajaa itsenäisesti automatisoituja prosesseja usein tarkoitukseen varatulla tietokoneella (UiPath 2021c).   |
| Orchestrator | Verkkoympäristö, jossa robotteja valvotaan ja hallitaan (UiPath 2021d).   |
| Robot Owner  | Robottien käyttäjä kohdeyrityksessä.  |

# 1 JOHDANTO

## 1.1 Työn tausta

Ohjelmistorobotiikka, eli RPA (Robotic Process Automation), on yksi digitalisaation alle mahtuvista tekniikoista, jotka muuttavat ja tehostavat työelämää sekä yhteiskuntaa. Siinä missä teollisessa tuotannossa käytetyt robotit toistavat säännönmukaisia tehtäviä tuotantolinjalla, esim. siirtävät ja kiinnittävät kappaleita, tekee ohjelmistorobotiikka tätä tietotyölle. Myös koulutusta vaativassa asiantuntijatyössä merkittävä osa työaika kuluu yksinkertaisiin, itseään toistaviin ja suorastaan tylsiin tehtäviin, kuten lomakkeiden täyttelyyn. Juuri näiden tehtävien toteuttamiseen voidaan robotiikalla pureutua. Kun sopivat prosessit tapahtuvat nopeammin ja vähemmällä virheillä, työntekijöiden aikaa säästyy mielekkäämpiin tehtäviin ja mikä tärkeintä: työn tuottavuus lisääntyy.

Kun tietoprosesseja aloitetaan toteuttamaan yrityksessä robotiikalla, on ensimmäisiä toteutuksia verrattain helppo löytää. Ne ovat prosesseja, joiden kulku on hyvin tiedossa ja säännönmukaista ja joiden automatisoinnilla on merkittävä liiketoiminnallinen hyöty. Tällaisia prosesseja on esimerkiksi taloushallinnossa huomattavasti. Talouden ja hallinnolliset tehtävät ovatkin tähän asti olleet RPA:n tyypillisin käyttökohde (Cabello et al. 2020, 188). Lukuisat kirjapidon perustoimenpiteistä ovat hyvin säännönmukaisia ja toistuvia, kuten päivittäisiin kirjauksiin ja kausittaiseen raportointiin liittyviä tehtäviä. Myös kaikenlaisten raporttien luominen tai sähköpostin läpikäyminen ovat tyypillisiä sovelluskohteita.

Robottien kasvava määrä tuotannossa asettaa haasteita näiden hallinnassa. Jokainen robotti tarvitsee omat resurssinsa, kuten käyttöympäristön, toiminnan seuranta ja pidemmällä aikavälillä varmasti ylläpitoa ja muutoksia. Tämän diplomityön kohdeyrityksen sisäisiä talousprosesseja on automatisoitu jo laajasti. Prosessien määrän jatkuvasti kasvaessa näiden hallinnasta ilman selkeää mallia on kuitenkin tullut työlästä. Robotiikan roolia on edelleen tarkoitus kasvattaa yrityksessä ja tämä vaati skaalattavan hallintamallin. Tällaista mallia voitaisiin myös tarjota ulkoisille robotiikka-asiakkaille.



## 1.2 Tavoitteet ja rajaus

Työn keskeinen tavoite on löytää ja toteuttaa yhtenäistetty malli robotiikan skaalaamiseen kohdeyrityksessä. Ensisijainen tutkimuskysymys on seuraava: Miten jatkuvaa ja kasvavaa robotiikkaprosessien laivuetta on tehokasta hallita? Tämä jakaantuu useampaan alakysymykseen:

- Miten seurannasta ja ylläpidosta tehdään mahdollisimman suoraviivaista
- Miten teknisiä toteutuksia on yhtenäistettävä
- Miten sidosryhmien kesken kommunikoidaan tuloksista, ongelmista ja kehityskohteista

Lopputulemana on siis tarkoitus löytää ja toteuttaa selkeä toimintamalli, joka on skaalattavissa. Tämä on tärkeää, jotta robotiikasta saatavaa hyötyä voidaan merkittävästi kasvattaa. Puhuttaessa kymmenistä prosesseista on ylläpidosta tullut jo melko työlästä. Ilman selkeää toimintamallia saattaisi satojen prosessien toteuttaminen muodostua työtaakaltaan täysin kestäättömäksi. Helpottamalla valmiiden prosessien ylläpitoa, uusien luomiseen saadaan lisää puhtia, ja siten skaalaus jopa näihin satoihin prosesseihin tulee mahdolliseksi. Kun yrityksen sisäinen malli on saatu testattua ja todettua hyväksi, voi tästä kokemuksesta jatkossa olla arvoa myös asiakkaille.

Ensiksi koostetaan tiedossa olevat haasteet yhteen. Tämän jälkeen luodaan yhtenäinen viestintäkanava eri robotiikkaa hyödyntävien yksiköiden kesken. Luodaan yhtenäistetty malli robottien tekniselle toteutukselle, eli miten lähtötiedot ja raportointi toteutetaan. Nämä yhteen nivoutuvat mallit otetaan käyttöön ja parannetaan iteratiivisesti. Ulkopuolelle jää kuitenkin yksittäisen projektin käytännöt, resurssien hallinta sekä yksittäiset prosessit.

Erilaisia RPA tekniikoita on markkinoilla jo useita, niin kaupallisia ohjelmistoja, kuin avoimeen lähdekoodiin perustuvia. Merkittävimpiä tekijöitä markkinoilla ovat UiPath, Automation Everywhere, ja Blue Prism (G2 2021). Markkinoille on myös odotettavissa uusia tulokkaita, kuten Power Automate, jotka voivat hyvinkin muuttaa kilpailutilannetta (Microsoft 2021b). Tässä työssä kohdeyrityksen oletetaan kuitenkin toistaiseksi keskittyvän UiPath ohjelmistoon, jonka termistöön opinnäyte vahvasti nojaa. Työn keskeiset ajatukset ja mallit eivät kuitenkaan ole käytettyyn ohjelmistoon sidottuja.

### 1.3 Tutkimuksen toteutus

Tutkimus toteutetaan työsuhteessa sen tarjonneeseen yritykseen keräämällä laadullista ja määrällistä tietoa robotiikkalaivueen nykytilasta, sekä etenkin ratkaisuehdotusten vaikutuksista. Vapaaseen ideointiin, kollegoiden haastatteluihin ja teoriaan pohjautuen hahmotetaan ja toteutetaan runko haetulle mallille. Perustuen jo tiedossa oleviin haasteisiin ja kehityskohteisiin mallia aletaan rakentamaan. Samalla kerätään palautetta sen toimivuudesta. Laajemmat kyselyt tehdään, kun ensimmäistä ratkaisuehdotusta on alettu pilotoimaan. Näin saadaan käytäntöön perustuvaa palautetta ja tämän pohjalta toteutetaan parannuksia malliin.

Tutkimuksen voi siis nähdä olevan case-tyylinen, sillä se keskittyy yhteen ilmentymään tietynlaisesta tilanteesta tosimaailmassa. Tarkoitus on tämän tapauksen avulla löytää laajemmin sovellettavaa tietoa tiedostaen sen rajallisuus. Kuitenkin näin vahvasti ihmisten toimintaan sosiaalisessa ympäristössä liittyvässä aiheessa ei voida luoda ehdottomia lakeja. Lähestymistapa auttaakin ymmärtämään syyseuraussuhteita sekä rajoituksia, joiden sisällä ne pätevät. (Byrne & Ragin 2009, 1–2)

Teknologian lisäksi ja ehkä sitä merkittävämminkin robotiikka nähdään tässä työssä organisaationaalisenä ilmiönä, joka muokkaa sen toimintatapoja ja jopa kulttuuria. Siten yleiset keinot, joita käytetään tällaisten ilmiöiden tutkimiseen, soveltuvat hyvin. Tällaiset tutkimukset vaativat organisaation ilmiöiden ymmärtämistä tietyssä paikan ja ajan kontekstissa, sekä lukuisten toisistaan riippuvien muuttujien verkosta, jotka yksi muodostavat kohdetapauksen raamit. Tyypillisimpiä käytettyjä kvalitatiivisia tutkimusmenetelmiä ovat esimerkiksi organisaation jäsenten haastattelut, toiminnan tarkkailu, kohderyhmien hyödyntäminen sekä sen keräämien toimintaan liittyvien tietojen tutkiminen. (Byrne & Ragin 2009, 432)

Myös kyselyllä saadaan hyödyllistä tietoa tutkimuksen aineistoksi. Kyselyn datan laatu riippuu vastaajien määrästä, datan keräämiseen käytetty tekniikoista, haastattelujen hyödyntämisestä sekä etenkin käytetyistä kysymyksistä. Keskittymällä näiden huolelliseen suunnitteluun saadaan tehokkaasti parannettua vastausten laatua. Kysymysten tulee olla selkeästi ymmärrettyjä, ja vastaajilla tulee olla käytettävissä tarvittava tieto. Tavan, jolla kysymyksiin

vastataan, tulee välittää tieto siitä, mitä vastaaja haluaa sanoa. Lisäksi vastaajien tulee olla halukkaita vastaamaan kysymyksiin (Bickman & Rog 2009, 375–376)

Käytännössä palautetta kerätään robotiikkatiimin haastattelujen kautta, sekä robotiikan sidosryhmille lähetettävällä online-kyselyllä, jolla samalla selvitetään mielipiteitä robotiikan hyödyntämisestä yrityksessä yleisesti. Vastaukset kyselyyn ovat anonyymejä eivätkä vastanneet tiedä toistensa vastauksia. Robotiikan käyttäjinä sekä hyötyjinä heillä on myös asiaan liittyvää tietoa. Mallissa pyritään myös huomioimaan alan kirjallisuuden esittelemiä toimintatapoja suoraan robotiikkaan, mutta myös ohjelmistojen ylläpidon ja ketterää kehityksen menetelmien soveltuvuutta arvioimalla. Myös yrityksen sisäistä dokumentaatiota ja keräämää tietoa hyödynnetään toteutuksen eri vaiheissa. Keskeisiä lähteitä ovatkin tiimin jo aiemmin koostamat robotiikan kehityskohteet ja toteutuksessa hyödynnetään olemassa olevia ominaisuuksia. Ehdotetulle mallille tehdään myös QFD analyysi verraten sitä robotiikasta kerättyihin arvoihin. Näin saadaan kuva siitä, miten se vastaa käyttäjien ohjelmistorobotiikassa arvostamiin ominaisuuksiin.

#### **1.4 Raportin rakenne**

Luvussa 2 laajennetaan tarkastelua liiketoiminnan prosesseihin yleisesti. Mitä ne ovat, miten niitä hallitaan ja mallinnetaan ja mitä muita automaation menetelmiä niissä käytetään. Luvussa 3 käsitellään ohjelmistorobotiikkaa yleisesti. Miten se määritellään, miten se muuttaa liiketoimintaa, miten sitä käytännössä kehitetään ja ylläpidetään ja kuinka sitä lähdetään skaalaamaan. Luvussa 4 keskitytään kommunikaation teoriaan ja sovelletaan sitä robotiikan kontekstiin.

Luvussa 5 siirrytään pois teoriasta, ja esitetään kohdeyrityksen sidosryhmiltä selvitetty robotiikan arvot. Luku 6 esittelee tavoitellun mallin viestinnän ns. pehmeitä osa-alueita. Millaista kommunikaation keskittymistä kohdeyrityksessä lähdettiin tekemään ja millaista sen yleensä kannattaa olla. Raportissa malli esitetään ensimmäisenä ratkaisuvaihtoehtona, palautteen keräämisenä ja toisena ratkaisuehdotuksena, vaikka kehitys ei todellisuudessa ole lineaarinen. Luku 7 rakentuu edeltävän tavoin keskittyen robottitoteutusten teknisempään

puoleen. Niin tekniset kuin kommunikaation aspektit ovat vahvasti kytköksissä keskenään ja tarjoavat siten ratkaisukokonaisuuden. Tutkimuksen tulokset esitetään ja käydään läpi. Tuloksia arvioidaan ja ehdotetaan jatkotoimenpiteitä. Luvussa 10 esitetään yhteenveto työn etenemisestä ja johtopäätöksistä. Kuvassa 1 on esitettyä input/output-kaavio työn vaiheista.



**Kuva 1** Työn input/output -kaavio

Kuva 1 listaa kunkin luvun vaatiman syötteen ja sen tulokset. Luvun 1 syötteenä on tutkimuksen esittely ja näistä johdetaan sen tavoitteet, rajaukset ja tutkimuskysymykset. Luvussa 2 tarvitaan liiketoimintaprosessien ja automaation teoriaa ja niillä saadaan esiteltyä prosessien hallinnan ja automaation periaatteet. Luvussa 3 tarvitaan ohjelmistorobotiikan teoriaa, ja sen pohjalta esitetään yleiskuva ohjelmistorobotiikasta ja sen skaalaamisesta. Luku 4 puolestaan käyttää viestinnän ja luottamuksen teoriaa ja näiden avulla saadaan käsitys niiden merkityksestä ohjelmistorobotiikalle. Luku 5 käyttää kohdeyrityksen sidosryhmille lähetettyä kyselyä, jonka vastauksilla saadaan tieto robotiikan arvoista. Luvussa 6 tarvitaan viestinnän teoriaa, tuntemusta käytetyistä järjestelmistä ja kyselyn palautetta. Näillä luodaan malli viestinnälle. Luvussa 7 tarvitaan ohjelmistorobotiikan tuntemusta ja sen asiantuntijoiden haastatteluja. Näillä luodaan tekninen malli helpottamaan robotiikan ylläpitoa ja skaalaamista. Luvussa 8 yhdistetään ehdotettu malli ja robotiikan arvot ja näistä luodaan QFD analyysi. Luvun 9 syötteenä toimii tutkimuksen siihenastiset tulokset, ja lopputulemana nämä arvioidaan ja ehdotetaan jatkotoimenpiteitä. Luku 10 tarvitsee yhteenvedon aikaisemmista vaiheista ja niistä koostetaan tutkimustulokset.

## 2 LIKETOIMINNAN PROSESSIT

### 2.1 Prosessien mallintaminen ja hallinta

Jotta ohjelmistorobotiikka saadaan toimimaan samalla tavalla kuin ihminen, on ensisijaisen tärkeää ymmärtää mitä nämä tekevät. Liiketoiminnan prosessit ovat keskeisimpiä yrityksen omistuksia. Niiden suora vaikutus tuotteiden ja palveluiden kysyntään asiakaskokemuksen kautta vaikuttavat lopulta liikevaihtoon. Ne määrittävät hinnoittelua ja toiminnan tuottavuutta. Erityisesti ne määrittävät tehtäviä, töitä ja vastuuta niille toimijoille, jotka prosessiin kuuluvat. Toisin kuin monien muiden yrityksen omistusten, vaikkapa selkeästi raha-arvoisten tuotteiden ja varastojen tai brändin kohdalla, niiden koettu merkitys pysyi pitkään aliarvostettuna. Globalisaation, innovoinnin, standardin ja tuottavuuden kasvun myötä niihin olette kiinnittää enemmän huomiota. Erilaisia työkaluja, tekniikoita ja menetelmiä on luotu, ja useat näistä hyötyvät digitaalisista tekniikoista. (Dumas et al. 2018, vii)

Kaikkien organisaatioiden, niin julkisten, hyväntekeväisyys, kuin yritysten, on hallittava lukuisia prosesseja. Tyypillinen esimerkki, joka löytyy useasta organisaatiosta, olisi esimerkiksi ”Order-to-cash”, eli tilauksesta maksuun. Myyntitilanteessa asiakas toimittaa tilauksen, myyjä tilatun tuotteen tai palvelun ja tilaaja maksaa sen. Toinen esimerkki olisi ”Application-to-approval”, eli hakemuksesta hyväksytään. Siinä joku hakee hyötyä tai oikeutta ja se päättyy tämän myöntämiseen tai epäämiseen. Ympäristöt ja tavat eri toimintoihin vaihtelevat suuresti, mutta periaate on sama. Liiketoiminnan prosesseja tapahtuu siis aina kun yritys toimittaa tuotteen tai palvelun asiakkaalle. Tapa, jolla tämä prosessi on suunniteltu vaikuttaa kilpailuasemaan, kun palvelun laatu ja tehokkuus vaihtelevat. (Dumas et al. 2018, 1–2)

Useimmat organisaatiot ovat hierarkkisia rakenteeltaan. Ylin johto ohjaa yrityksen suuntaa visioin ja strategisten tavoittein, keskijohto tuovat nämä tavoitteet käytäntöihin ja operatiivisella tasolla keskitytään jokapäiväisiin liiketoiminnan tilanteisiin. Liiketoiminnan prosessit heijastelevat tätä järjestystä, joka siis tyypillisesti jakautuu strategiseen, taktiseen ja operatiiviseen tasoon, ja kullakin tasolla eri toimijat toteuttavat erilaisia arvoa luovia aktiviteetteja, prosesseja. Ylemmillä tasoilla ohjataan alempaa ja alhaalta taas saadaan

palautetta takaisin. Voidaan siis puhua monitasoisesti liiketoimintaprosessien pyramidista. (G. Schuetz 2015, 1–2)

Prosessien hallinta on lähellä ja tukee monia muita liiketoiminnan käytäntöjä ja suuntauksia. Total Quality Management (TQM), eli kokonais-laadunhallinta tähtää prosessin hallinnan tavoin jatkuvasti parantamaan ja ylläpitämään tuotteiden ja palveluiden laatua. Prosessin sijaan keskitytään kuitenkin lopputulokseen, eli tuotteisiin ja palveluihin itsessään. Operations Management, eli toiminnan hallinta keskittyy fyysisten ja teknisten toimintojen hallintaan etenkin tuotannon ja varmistamisen yhteydessä. Siinä todennäköisyydet, päätös ja analyysit, ja matemaattiset mallit ovat tärkeitä optimoinnin menetelmiä. Lean on valmistavassa teollisuudessa syntynyt johtamismenetelmä, jolla pyritään ennen muuta välttämään hukkaa, eli arvoa lisäämättömiä toimenpiteitä. Six Sigma, joka myöskin on syntynyt valmistavassa teollisuudessa, keskittyy mittaamaan prosessien ja toimenpiteiden tuloksia, etenkin laadun suhteen. käytännössä nämä kaikki siis tukevat toisiaan eivätkä ne ole poissulkevia. Yhdistettynä IT:n kanssa ne auttavat parhaiten ohjaamaan toiminnan organisaation tavoitteiden saavuttamiseksi. (Dumas et al. 2018, 7–8)

Liiketoimintaprosessit koostuvat erilaisista tekijöistä. Tapahtumilla tarkoitetaan tiettyä tarkkaa tapahtumaa, joka ei prosessin kannalta vaadi aikaa. Oikean työkalun saapuminen työmaalle toimii esimerkkinä tästä. Työkalu ei ole saatavilla, kunnes se on, ja tämä muutos on tapahtuma. Tämä tapahtuma saattaa laukaista sarjan erilaisia toimenpiteitä, joku saattaa esimerkiksi hakea työkalun. Eron tapahtumasta toimenpiteet vaativat aikaa. Toimenpiteiden ja tapahtumien lisäksi liiketoimintaprosesseihin kuuluu usein päätöskohtia, eli vaiheita, jossa päätetään, miten prosessi jatkuu. Esimerkiksi muista tapahtumista riippuen työkalua saatetaan tarvita tai ei. Lisäksi prosesseissa on toimijoita kuten ihmisiä tai ohjelmistoja, fyysisiä esineitä, kuten tarvittuja materiaaleja sekä tietoesineitä, kuten levyille tallennettu tiedosto tietokoneella. (Dumas et al. 2018, 3–4)

Liiketoimintaprosessien mallinnus, engl. Business Process Modelling (BPM), on menetelmä, joka muuttaa IT:n ja liiketoiminnan yhteensovittamista interaktiiviseksi tavaksi rakentaa ratkaisuja. Sillä saadaan aikaan nopeampia, parempia ja halvempia ratkaisuja ja se yhdistää

ylhäältä alaspäin jatkuvan kehittämisen ylhäältä katsottuna strategiseen kuvaan. Sillä saadaan yrityksen satunnaiset toimintatavat yhtenäistettyä ja uudelleenkäyttöön. (IBM 2020, 2)

Mallia määrittää kolme ominaisuutta; se on kuva oikean maailman ilmiöstä eli mallin kohteesta, se mallintaa kohteesta olennaisia asioita riittävällä tarkkuudella ja sillä on tarkoitus, eli mitä varten malli tehdään (Dumas et al. 2018, 78). Liiketoimintaprosessien mallintaminen on tärkeää niiden hallinnan kannalta. Prosesseja mallinnetaan, jotta niitä voidaan ymmärtää ja jakaa tämä ymmärrys muiden kanssa, jotka prosessiin liittyvät. Kun kokonaiskuva saadaan luotua, voidaan myös löytää mahdollisia ongelmia. Tämän pohjalta prosesseja voidaan analysoida, suunnitella uudestaan ja jopa automatisoida. (Dumas et al. 2018, 75)

Mallinnetut prosessit ovat usein liiketoiminta alueiden ja IT-järjestelmien yhteensovittamista. Kunnia esimerkiksi onnistuneesta muutosprojektista IT painotteisemmaksi tulee kuitenkin jakaa myös mallintamiselle. Ilman sitä prosessien muutokset olisivat paljon vaivalloisempia. Prosessien mallintamiseen tulee olla strategisesti keskittyntä eikä jäädä vain taktiselle tasolle. Yksittäisten sekalaisesti toimivien teho yksikköjen sijaan on pyrittävä kokonaisuuteen, jossa oppiminen lahjakkuus, uudelleenkäyttö ja viestintä toimivat hyvin. Heikkotasoiset tai jopa huonot prosessit tulee huomata ja korjata nopeasti. Vaikka kaikissa prosesseissa olisi parantamisen varaa on syytä aloittaa suurimmista ongelmista. Prosessin mallinnusmenetelmän käyttöönotolla on vaikutus IT:n toimintaan, kun se muuttaa lähestymistapaa projekteihin. On tärkeää huomata, että prosessit ovat usein käytettävissä uudelleen, mitä lisää niiden arvoa. (IBM 2020, 4)

## **2.2 Prosessien automaatio**

Digitaalisia liiketoiminnan prosesseja voidaan tehostaa ja automatisoida myös monilla muilla tavoilla kuin robotiikalla. Jotta ymmärretään robotiikan roolia muiden digitalisaation ilmiöiden joukossa, on hyvä ymmärtää myös näitä muita menetelmiä ja verrata robotiikkaa tähän.

IT-ala on vuosikymmenet keskittynyt prosessien automatisointiin kasvattaakseen tuottavuutta ja tehokkuutta sekä vähentääkseen käyttäjien virheitä. Markkinat ovat muuttaneet pääpainon tarpeesta tuottaa informaatiota nopeammin liiketoiminnan tuottavuuden kasvattamiseen.



Tämän johdosta haetaan kustannustehokkuutta sekä tehokkuuden kasvatusta, jotta pääoma voidaan ohjata strategisiin investointeihin. Tähän asti prosessien automaatio on toteutettu skripteillä, jotka yhdistävät tietokoneiden komentoja ja jotka ovat sidottuja tiettyyn alustaan tai ohjelmistoon. (Antonoaie 2017, 1)

Liiketoimintaprosessien hallintajärjestelmä, engl. Business Process Management System (BPMS), mahdollistaa prosessien suunnittelun analysoinnin, toteutuksen ja seurannan prosessimalleihin perustuen. Ne ovat lähtöisin vanhemmista järjestämistä, kuten työjonojen hallinta järjestelmistä, engl. Workflow Management System (WfMS). Niiden tarkoitus on ohjata automatisoituja liiketoiminnan prosesseja niin, että kaikki työ tehdään oikeaan aikaan ja oikeilla resursseilla. Näillä järjestelmillä on paljon yhtäläisyyksiä tietokantojen hallintajärjestelmien kanssa. (Dumas et al. 2018, 344–345)

Prosessienhallintajärjestelmillä varmistetaan, että prosessiin kuuluvat toimenpiteet toteutuvat oikeassa järjestyksessä ja tarjotaan tarvittavia resursseja. Järjestelmä saattaa pystyä jopa suorittamaan osan näistä toimenpiteistä itsenäisesti, eli automaattisesti. Liiketoimintaprosessien automaatioon on luotu jopa lukuisia ohjelmointikieliä. (G. Schuetz 2015, 15)

Järjestelmät jakautuvat erilaisiin alaluokkiin. Ryhmä järjestelmiin kuuluvat sellaiset järjestelmät, joissa mahdollistetaan helppo dokumenttien ja informaation jako sekä suoraa viestintää eri käyttäjien välillä. Nämä järjestelyt ovat hyvin suosittuja joustavine käyttötapojen johdosta. Esimerkiksi Microsoft Teams voitaisiin nähdä tällaisena (Microsoft 2021a).

Ad hoc työkulkujärjestelmät mahdollistavat hetkessä luotavat ja muokattavat prosessimääritelmät. Se edellyttää, että loppukäyttäjät tuntevat prosessin, jota suorittavat ja että heillä on käytettävissään edistyneitä työkaluja ja osaamista liiketoiminnan prosessin mallintamiseen. Esimerkiksi TIBCO ActiveMatrix BusinessWorks järjestelmän ideana on integroida sovelluksia helppokäyttöisellä visuaalisella käyttöliittymällä (TIBCO Software 2021).

Tyypillisimpiä ovat tuotannon työnkulku järjestelmät. Missä työ on järjestetty tarkasti erikseen määriteltyjen ja mallinnettujen prosessien mukaan, ja ne ovat usein kytköksissä tietokantojen hallintajärjestelmään. Ne tukevat prosesseja, jotka ovat lähes kokonaan automatisoitu. Tapaushallintajärjestelmissä puolestaan tuetaan prosesseja, jotka eivät ole tarkasti tai kokonaan määriteltyjä. On myös lukuisia muita järjestelmätyyppejä, jotka tyypillisesti yhdistävät edellä kuvattujen ominaisuuksia. (Dumas et al. 2018, 345–347)

Yksi konkreettinen teknologia, joka mahdollistaa tehokkaan tiedonsiirron ja kommunikaation eri järjestelmien välillä ja siten lisää automaatiota ovat ohjelmistorajapinnat, engl. application programming interface (API). Nimensä mukaisesti se on rajapinta eri ohjelmistojen välillä, mikä määrittelee näiden välisen kommunikaatiotavan ilman käyttäjän vaikutusta. Kun varaa hotellihuoneen matkustussivustolta, tarvitaan rajapintoja esimerkiksi hotellin varausjärjestelmän, maksuohjelman ja pankkisovelluksen välillä. Tiedot siirtyvät näennäisen saumattomasti järjestelmästä toiseen ja kaikki varauksen liittyvät kirjaukset tapahtuvat automaattisesti. kommunikaation järjestelmien välillä perustuu siihen, että rajapinnan avulla osa eri järjestelmien funktioista näkyvät niiden ulkopuolelle. niin voidaan jakaa palveluita, objekteja, ja dataa ilman koko lähdekoodin jakamista. (De & Doda 2017, 1)

Makrot, engl. macro instructions, ovat ohjelmoitavia kaavoja, jotka vähentävät toiston tarvetta esittämällä sarjan näppäimen lyönnejä, hiiren liikkeitä, komentoja tai muunlaisia syötteitä (Computer Hope 2021). Erilaisia makroja voidaan tehdä lukusissa ohjelmistoissa, kuten Microsoft Excelissä. Käyttämällä joitakin VBA-ohjelmointikielen komentoja voidaan automatisoida joitakin tyypillisiä rutiinitehtäviä. Makroja voidaan tallentaa tai muokata kooditasolla editorissa. Niiden avulla voidaan esimerkiksi mallintaa datan syöttöä tai tarkastaa taulukoita tuplakirjausten varalta. (Korol 2017, 3–4)

## 3 OHJELMISTOROBOTIIKKA

### 3.1 Ohjelmistorobotiikan perusteet

RPA, eli ohjelmistorobotiikka on historian nopeimmin kasvanut yritysohjelmistoliiketoiminnan osa-alue. Sen tapa luoda uutta arvoa on pelottavan yksinkertainen, eli laittaa robotit tekemään työlääät digitaaliset tiedonkäsittelyprosessit puolestamme. Ohjelmistorobottien toteuttamisen on tarkoitus olla nopeaa, sillä kuukausien kalliin rajapintakehittämisen sijaan ne saadaan käyttämään tietokoneen näyttöpäätettä samalla tavoin kuin ihmiskäyttäjätkin. Siten robotti voi esimerkiksi kopioida ja liittää tietoa sovelluksesta toiseen, avata nettisivustoja ja kirjautua sisään, lukea ja kirjoittaa tietokantoja, kerätä tietoa dokumenteista ja tehdä laskutoimituksia ja datan muokkaamista. Nämä voivat kuulostaa yksinkertaisilta ja tylsiltä tehtävältä, mutta juuri se onkin tarkoitus. RPA on keskittynyt tehtäviin, jotka muuten tuhlasivat ihmiset työntekijöiden aikaa. (Taulli 2020, XV & 3)

RPA ohjelmistot automatisoivat olemassa olevia toistuvia tehtäviä. Niillä rakennetaan robotteja, jotka matkivat ihmistä käyttämällä eri sovelluksia. Nykyisin on käynnissä laaja ohjelmistorobotiikan käyttöönottoa esimerkiksi taloushallinnon, terveystieteiden ja vakuutusaloilla. Robotiikka muistuttaa prosessien hallintaa (engl. Business Process Management, BPM) ja prosessiautomaatiota (engl. Business Process Automation, BPA), mutta eroaa siinä, että laajojen prosessien ja ohjelmistojen yhteensovittamisen ja muokkaamisen sijaan RPA toteuttaa jo olemassa olevia prosesseja samalla tavalla. (Trustadius 2021)

RPA eroaa muista automaation menetelmistä sillä, että toisin kuin esimerkiksi makrot se ei ole rajattu yhteen ohjelmistosovellukseen. RPA voi tallentaa ihmiskäyttäjän toimintoja luodakseen automaation ja se voi jopa hyödyntää tekoälyä. Lisäksi RPA on tehokas tapa taltioita tietoa siitä, miten asioita tehdään yrityksessä, mikä voi puolestaan epäsuorasti parantaa prosesseja.

On erilaisia lähestymistapoja ohjelmistorobotiikkaan. Jotkin robotit avustavat ihmistä näyttöpäätetyössä eli ovat ns. attended-robotteja. Tämä on varhaisin robotiikan muoto ja sitä esiintyy jo 2000-luvun alussa. Robotti voi esimerkiksi avustaa puhelinkeskuksesta etsimällä tietoa asiakkaasta, jonka kanssa työntekijä puhuu. Myöhemmin kehittyivät täysin itsenäiset

robotit eli unattended-robotit. Itsenäinen robotti voi toimia ilman ihmisen puuttumista asiaan. Se voi esimerkiksi käynnistyä tiettyjen määriteltyjen asioiden tapahtuessa, esimerkiksi sähköpostin saapuessa. Viimeisimpinä on kehittynyt älykäs automaatio tai ns. kognitiivinen robotiikka. Tällöin tekoäly mahdollistaa ajan kanssa paranevan suorituksen. (Taulli 2020, 5–6)

Automaatiolla on jo pitkä historia, joka jakaantuu useisiin vaiheisiin ja jotka johtavat lopulta moderneihin RPA alustoihin. Tietokoneiden alkutaipaleella niin sanotulla mainframe-kaudella oli vain valtavia tietokoneita, joita kehitti muun muassa IBM. Niitä käytettiin vain suurissa yrityksissä kuten pankeissa ja ne pystyivät auttamaan monissa ydintoiminnoissa kuten asiakastileissä ja palkanmaksussa. PC:n valtakausi alkoi Intelin kehittämän mikroprosessorin ja Microsoftin kehittämän käyttöjärjestelmän myötä. Sen ansiosta paljon suurempi kirjo yrityksiä pystyi automaatiolla avittamaan prosessejaan, kuten käyttämään taulukko- ja kirjoitusohjelmia, modernien toimisto-ohjelmistojen esiasteita. Näissä varhaisissa automaation menetelmissä oli omat haittapuolensa. Ne johtivat monimutkaisiin IT-ympäristöihin, jotka vaativat kalliita ja aika aikka vieviä integraatioita ja koodaamista. Tämän johdosta työntekijä saattoi joutua käyttämään useita eri ohjelmia tehottomiin tehtäviin kuten datansiirtoon. Näistä aineksista syntyi ohjelmistorobotiikka jo varhaisella 2000 luvulla. Pitkään se ei kuitenkaan saanut yhtä paljoa huomiota kuin muut teknologiat, kuten pilvipalvelut, jotka kumosivat vanhoja IT-järjestelmiä, kiinnittivät sijoittajien ja yrittäjien huomion. (Taulli 2020, 7–8)

Vasta 2010-luvulla RPA alkoi kasvattaa suosiotaan. Finanssikriisin jäljiltä yritykset etsivät tapoja pienentää kustannuksiaan. Yksinkertaistettuna teknologiat, kuten ERP (Enterprise Resource Planning) saavuttivat täyden potentiaalinsa, ja siten tarvittiin uusia ajureita. Ymmärrettiin myös, että perinteisen liiketoiminnan oli puolustauduttava eri aloja mullistavilta teknologiayrityksiltä. RPA:ta pidettiin yksinkertaisempuna ja kustannustehokkaana tapana edistää digitalisaatiota. Joistakin aloilla kuten pankeilla liiketoiminnasta tuli myös säädellympää, jolloin syntyi paine vähentää paperityötä kuitenkin parantaen seurattavuutta ja turvallisuutta. Samalla RPA kehittyi helpommaksi käyttää tarjoten parempaa sijoituksen palautusta yhä nopeammin toteutettavien kehitysprojektien myötä. Vaikutuksensa oli myös työelämään siirtyville nuorille, jotka kaipaivat entistä mielekkäämpää työtä. (Taulli 2020, 8–9)

Näiden tekijöiden voi edelleen nähdä olevan keskeisessä asemassa ohjelmistorobotiikassa. Esimerkiksi UiPathin viestintä kiteytyy ajatukseen: ”We make software robots, so people don’t need to be robots”, eli ”teemme ohjelmistorobotteja, jotta ihmisten ei tarvitse olla robotteja” (UiPath2021a).

RPA on siis ohjelmisto, joka auttaa automatisoimaan tiettyjä liiketoiminnan prosesseja, kuten CRM:n tai ERP:n, myynnin ja varastohallinnan järjestelmien kanssa työskentelyä. Ohjelmistorobotin voisi nähdä siis digitaalisena työntekijänä. RPA osittaisista yhtäläisyyksistään huolimatta ei ole makro, eikä siten ole rajattu tiettyyn sovellukseen ja ympäristöön. RPA-toteutusten tekeminen on suunniteltu verrattain helpoksi. Attended RPA tarkoittaa toteutusta, joka auttaa ihmistä näyttöpäätte työssä, kun taas unattend RPA on itsenäinen toteutus, joka toimii ilman ihmisen puuttumista sen voi käynnistää esimerkiksi sähköpostin saapuminen. (Taulli 2020, 19)

### **3.2 Ohjelmistorobotiikan hyödyt**

Ohjelmistorobotiikan hyötyjä mitataan usein säästöissä ajassa, sekä erinäisillä mittareilla, kuten virhemäärin pienentymisellä, asiakastyytyväisyyden kasvamisella tai jollakin muulla määritellyillä hyödyillä (UiPath 2020a). Robotiikan hyötyjä ja käyttökohteita on viime vuosina tutkittu laajasti myös akateemisesti, ja aiheesta onkin tehty mm. diplomitöitä; Mäkelä 2019, Tuominen 2020.

Ohjelmistorobotiikasta saadaan erilaisia hyötyjä, kuten perinteistä henkilötövuosien kustannussäästöä, prosessin suorittamisteen nopeutta ja laatua sekä tämän tasaisuutta, sekä ihmistyön voiman kohdistamista hyödyllisempään tehtäviin (Mäkelä 2019, 20). ”Ohjelmistorobotiikan hyötyjä ovat toiminnan tehokkuuden lisääntyminen ja ajan säästäminen, kun ohjelmistorobotti tehostaa taloushallinnon toimintoja säästää aikaa, kun huolimattomuusvirheitä ei synny samalla tavalla kuin ihmisten suorittamana (Tuominen 2020).”

### -Rahallinen hyöty

Ehkä useimmiten puhuttu ohjelmistorobotiikan hyöty on sijoituksen palautus, eli rahallinen säästö. Verrattuna muihin ohjelmistotekniikoihin tilastot puhuvat puolestaan. Ohjelmistorobotiikan hyödyt eivät kuitenkaan jää vain suoriin rahallisiin hyötyihin. Otetaan esimerkiksi pienten parannusten vaikutus: yksittäisen työntekijän toimenpiteen nopeuttaminen 10 sekunnilla ei vaikuta sinänsä paljon, mutta jos sama vaikutus skaalataan tuhansiin työntekijöihin, vaikutus muuttuu suureksi. Joissain yrityksissä esimerkiksi seurataan, kuinka paljon työaikaa on saatu säästettyä ohjelmistorobotiikalla. Tämä vaikuttaa osaltaan ROI:hin (Return of Investment), eli sijoituksenpalautukseen. (Taulli 2020, 10–11)

Aquirre & Rodriguez (2017, 69) case-prosessissa esimerkiksi saatiin ryhmä RPA:ta käyttämällä hoitamaan 21 % enemmän tapauksia. Juuri tällaista tuottavuuden paranemista ohjelmistorobotiikalta odotetaan. Toisaalta keskiarvoon nähden robotti tarvitsi vain 2 % vähemmän aikaa yhteen suorituskertaan. Selitys löytyy siitä, että yksittäiset taitavat työntekijät pystyvät suorittamaan tehtäviä jopa robottia nopeammin, mutta robotti pystyi käsittelemään monta tapausta samalla ja tasaisesti. (Aquirre & Rodriguez 2017, 69)

### -Verrattain nopea implementointi

Yksi keskeisistä robotiikan hyödyistä on verrattain helppo implementointi verrattuna moneen perinteiseen ja laajempaan ohjelmistoprojektiin, kuten uuden järjestelmän käyttöönottoon. Sen sijaan robotiikkaa rakennetaan olemassa olevan IT- arkkitehtuurin päälle, mikä tekee siitä hyvinkin ketterää. Voisi jopa sanoa, ettei ohjelmistorobotiikka ole osa tätä arkkitehtuuria, sillä se toimii itsenäisesti käyttäjätunnuksillaan tämän arkkitehtuurin päällä (Syed & Wynn 2020, 148). RPA:n käyttö on helpompaa, kuin monimutkaisen perinteisen ohjelmoinnin ymmärtäminen ja siten IT-osaston tukea ei tarvita huomattavasti. Yksinkertaisesti sanottuna hyödyntämällä RPA:ta tavoitteet saadaan saavutettua nopeammin ja IT-osaajilla on aikaa keskittyä korkeamman prioriteetin tehtäviin. (Taulli 2020, 12)

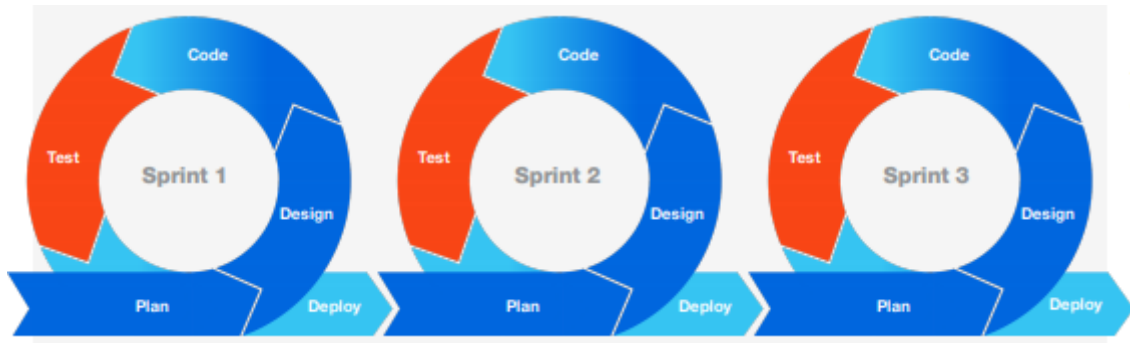
-Tarkkuus

Sääntöjen tai jopa lakien rikkomisesta voi olla valtava vaikutus yritykseen ja yksikin rike voi uhata sen koko olemassaoloa. Vaikka työntekijät useimmiten ovat täsmällisiä ja luotettavia, he voivat helposti tehdä virheitä, jos he eivät ymmärrä kaikkia tilanteeseen vaikuttavia sääntöjä. RPA:n kanssa tämä ei ole ongelma, sillä robotti on helppo konfiguroida noudattamaan sääntöjä tarkasti yhä uudestaan. Toisaalta ohjelmistorobotiikka vähentää myös riskiä tarkoituksenmukaiseen väärinkäyttöön, sillä ihmisen pääsyä dataan voidaan rajoittaa. RPA tarjoaakin tehokkaat keinot auditoitiin ja seurantaan. (Taulli 2020, 12)

### 3.3 Ohjelmistorobotiikan kehitys

UiPathin mukaan (2021b, 2) mukaan tyypillisiä ongelmia automaatio-ohjelman edetessä ovat ylläpidon kuormittuminen, ”kitka”, projektien rajausten jatkuva muutos sekä laatu. RPA-tiimi joutuu siis käyttämään liikaa aikaa olemassa olevien robottien ylläpitoon, eikä enää ole tarpeeksi aikaa ja resursseja uusien toteutusten tekemiseen. Robotteja ei saada tehtyä niin paljon kuin halutaan ja niin nopeasti kuin tarvitsee, projektin laajuus ja vaatimukset muuttuvat yhä uudestaan, eikä aika määreissä pysytä. Toteutuksen testaamisesta tulee rasittavaa. Ongelmien syynä saattaa olla käytetty toimintamalli -tai sen puute. Ohjelmistorobotiikan alkutaipaleella vesiputous malli voi toimia hyvin, mutta sen edetessä puuttuvat tai kankeat käytännöt voivat aiheuttaa esteitä; huonot tavat kerätä vaatimuksia ja näiden muuttuminen hidastavat projekteja ja vajaa testaaminen lisää virheitä ja siten ylläpitotyötä. (UiPath 2021b, 2–3)

Modernimmat ohjelmistokehityksen mallit sopivat siis paremmin robotiikalle. Nämä kehittyivät noin 20 vuotta sitten, kun ohjelmistotuotannossa painittiin samanlaisten ongelmien kanssa. Tekniikoita on useita erilaisia, mutta ne pohjautuvat samaan perusajatuksen. Jatkuva ohjelmistokehitys, Agile, perustuu interaktiiviseen ja asteittaiseen lähestymistapaan, mikä keskittyy yhteistyöhön, nopeisiin julkaisuihin, dynaamisuuteen ja jatkuvaan parantamiseen. Kuvassa 2 on kuvattuna ketterän kehityksen perusidea. (UiPath 2021b, 4)



**Kuva 2** Jatkuvan kehityksen malli (UiPath 2021b, 4)

Kuvassa 2 on kolme peräkkäistä sprinttiä. todellisuudessa ajatuksena on, että kehitys siis voi jatkaa kuinka monta kierrosta tahansa, kunnes on saavutettu tuotantokelpoinen toteutus. Ensimmäisellä kertaa kerätään vaatimukset vahvassa yhteistyössä sidosryhmien kanssa. Sitten aletaan toistamaan iteratiivista mallia suunnittelusta, yksityiskohdista, koodaamisesta testaamisesta ja julkaisusta. Tämä malli mahdollistaa suuren volyymin kehittämistä. Se nopeuttaa julkaisemista parantaa suunnittelun joustavuutta tiimin tuottavuutta ja kokonaislaatua. Siksi jatkuvan kehityksen malli onkin niin laajassa käytössä nykyisin (UiPath 2021b, 4)

Ketterällä ohjelmistokehityksellä on ollut suuri vaikutus ohjelmistotuotantoon. Se on yläkäsite erilaisille muutoksille ennen vallassa olleille tavoille siinä, miten kehittäjät suunnittelevat ja koordinoi työtänsä, kuinka he kommunikoivat asiakkaan ja muiden ulkopuolisten sidosryhmien kanssa ja miten kehitystä järjestetään kaikenkokoisissa yrityksissä. Sen erottaa perinteisestä suunnitelmapohjaisesta kehityksestä metodit etenkin muutoksiin sopeutumiseen ja sillä päästään korkealaatuiseen lopputulokseen yksinkertaisella prosessilla. (Moe et al. 2010)

Jatkuva kehitys vaatii vastaavasti jatkuvaa testaamista, mutta tämän hoitaminen manuaalisesti muuntuisi ajan kanssa yhä raskaammaksi. Siksi myös testaamisen automaatio on tärkeää. Ohjelmistorobotiikassa automaattisen testaamisen tarve on jopa suurempi kuin perinteisessä ohjelmistotuotannossa, sillä robotit toimivat usein sovellusten sisällä. Ne käyttävät juuri kyseisen ohjelmiston käyttöliittymää, tarkkaan määritettyä ympäristöä ja niin edelleen. Kaikki



nämä muuttuvat ajan kanssa, ajoittain jopa ilman varoitusta. Monissa yrityksissä on lukuisia sovelluksia, joita päivitetään viikoittain, päivittäin tai jopa tunneittain. Siksi mahdolliset ongelmat olisikin tärkeää löytää ennen kuin ne toteutuvat tuotannossa. (UiPath 2021b, 5–6)

Jatkuva testaaminen voidaankin sisältää keskeiseksi osaksi robottien rakentamista. Tämä onnistuu neljällä vaiheella. Suunnitellaan robotin vaatimukset ja testattavat asiat, rakennetaan prosessin kulut ja varmistetaan että kukin on testattavissa, valvotaan robottien tilaa testaamalla tuotannossa toimivia robotteja jatkuvasti ja toimitaan ennen kuin robotit epäonnistuvat tuotannossa. (UiPath 2021b, 7)

### **3.4 Ohjelmistorobotiikan ylläpito**

Ohjelmistorobotiikkaan liittyy paljon ylläpitoa, sillä:

1. Ohjelmistorobotiikkaa tehdään tietyllä ohjelmistolla, ja kuten muutkin ohjelmistot sitä parannetaan ja muutetaan ajan kanssa.
2. Ohjelmistot, joita robotit käyttävät päivittyvät ja muuttuvat ajan kuluessa, kun niitä ylläpidetään.
3. Liiketoiminnanprosessit, joita robotit toteuttavat, muuttuvat lukuisista syistä.

Siten on jopa perusteita väittää, että robotiikka vaatii erityisen paljon ylläpitoa ja muutoksia verrattuna muuhun ohjelmistotuotantoon. Ohjelmistorobotiikan ylläpitoa ei ole vielä laajasti erikseen tutkittu, joten on syytä tutustua yleiseen teoriaan ohjelmistojen ylläpidosta ja verrata tätä RPA:n erottaviin ominaisuuksiin. Toisaalta ohjelmistorobottien ylläpidolla voidaan nähdä yhtäläisyyksiä fyysisten koneiden, kuten robottien, ylläpitoon sillä ne voidaan nähdä ”tuotantolinjana” tietotyölle.

Ohjelmistot (engl. software) eivät koostu vain ohjelmasta, vaan niiden kokonaisuuteen kuuluu myös monia muita osia, kuten dokumentaatio ja käyttöproseduuri. Dokumentaatio jakautuu edelleen projektin vaiheisiin. Määrittelyvaiheessa luodaan, esimerkiksi vaatimukset. Suunnitteluvaiheessa luodaan esim. vuokaavio tai ER-kaavio. Toteutusvaiheessa dokumentaatiota ovat lähdekoodilistaus ja viittaukset. Testauksessa näitä ovat taas testien data

ja tulokset. Käyttöön kuuluvat ohjeet ympäristön valmistamisesta, ohjelman käyttämisestä ja virhetilanteiden käsittelystä (Grupp ja Takang 2003, 8)

Ohjelmiston ylläpidettävyyttä on vaikeaa mitata kattavasti, mutta monia sen tekijöitä, kuten monimutkaisuutta voidaan, ja tiedetäänkin, että standardoinnin taso on tärkeää. Ylläpito kokonaisuudessaan koostuu hieman eroavista toimista, kuten virheiden korjaamisesta, muutostarpeista ja käyttäjän avustamisesta. Siihen liittyy siis virheiden havaitsemista ja korjaamista, muutosten tekemistä, kun vaatimukset tai ympäristö muuttuvat sekä käyttäjän tukemisesta epäselvissä ja virhetilanteissa. (Grupp ja Takang 2003, 8)

Siinä missä uudessa toteutuksessa ratkaisut tehdään puhtaalta pöydältä, ylläpidossa on myöhemmin toimittava näiden mukaisesti. Kun siis olemassa olevaan toteutukseen halutaan tehdä muutos, esimerkiksi lisäys tai parannus, tulee tutkia olemassa olevaa systeemiä ja sisäistää sen suunnittelua eri tasoilla. Sitten pitää keksiä, miten lisäys saadaan toteutumaan ja mitkä ovat mahdollisia heijastevaikutuksia (engl. ripple effects). (Grupp ja Takang 2003, 9–10)

Uusien ominaisuuksien lisäämistä ohjelmistoon saatetaan pitää samanlaisena kuin näiden ominaisuuksien lisäämistä alusta alkaen, mutta todellisuudessa ylläpito eroaa kehitystyöstä joiltain osilta paljonkin. Ylläpidon tarvetta on saatettu jo alun alkaen pohtia, mutta tulevaisuuden tarpeita voidaan kehityksen alkuvaiheessa ennakoida vain tiettyyn pisteeseen. Ainakin näiden kaikkien huomioon ottaminen tulisi hyvin kalliiksi. Käytetyissä malleissa ylläpito voidaan kuitenkin ottaa huomioon, esimerkiksi dokumentointi voi auttaa tässä paljon. Kuitenkin siinä missä muilla tekniikan aloilla ei tule kuuloonkaan laiminlyödä tai toimia ilman dokumentaatiota, on tämä ohjelmisto tuotannossaan tyypillistä. Tällöin ylläpitäjän on tutkittava toteutusta, jotta sen osasten väliset riippuvuudet ja mahdolliset heijastevaikutukset voidaan ottaa huomioon. (Grupp ja Takang 2003, 73–74)

Yksinkertaisen ylläpidon menetelmä on ns. välittömän korjaamisen malli. Tällöin odotetaan, että ongelmia ilmenee ja korjataan ne ad hoc menetelmillä niin nopeasti kuin mahdollista. Tällaisessa mallissa korjauksia tehdään ilman syvällistä analyysia pitkän ajan vaikutuksista, kuten heijastevaikutuksista läpi ohjelman tai koodin rakenteen vaikutuksia. Dokumentaatiota on niukasti, jos ollenkaan. Malli ei kuitenkaan välttämättä ole kaikissa tilanteissa huono. Jos

kehittämisestä ja ylläpidosta esimerkiksi vastaa vain yksi henkilö, tuntee hän järjestelmän niin hyvin, että pärjää ilman yksityiskohtaista dokumentaatiota ja kykenee tekemään vaistonvaraisia päätöksiä muutoksen implementoinnista. Malli on myös halpa. (Grupp ja Takang 2003, 76–77)

Tavanomaisemmassa, huomattavan kokoisessa kaupallisessa tilanteessa malli ei kuitenkaan ole ihanteellinen ja tämä myös tiedostetaan. Kuitenkin tiukkojen aikataulujen ja rajallisten resurssien takia se saattaa jatkua. Kun asiakas vaatii nopeaa korjausta, voi yritykselle olla pienempi riski toteuttaa nopea korjaus, kuin viivästyttää asiakkaan toivetta yksityiskohtaisilla aikaa vievillä analyyseillä. Pitkällä aikavälillä tällainen toiminta johtaa kuitenkin vaikeisiin ja kalliisiin ongelmiin, jotka saatetaan lopulta ratkaista yhä uusilla ”helpoilla korjauksilla”. Onkin syytä erottaa lyhyen ja pitkän aikavälin korjaukset. Siinä missä yksittäisen bugi voidaan korjata nopeasti, suuremmat rakenteelliset muutokset voidaan tehdä harvemmillä suurilla päivityksillä. (Grupp ja Takang 2003, 77)

Boehmin malli esittää ylläpidon prosessin suljettuna syklinä. Siinä keskiössä ovat taloudelliset mallit ja periaatteet, joiden avulla voidaan mahdollisesti parantaa ylläpidon tuottavuutta ja ymmärtää prosessia. Tämän mallin mukaan johdon päätökset ajavat prosessia. Strategisten menetelmien ja kustannushyöty laskelmien pohjalta tuotetaan ehdotettu korjaus. Tälle lasketaan oma budjetti, mikä pitkälti määrittää käytetyt resurssit muutokselle. Sitten muutokset implementoidaan, mitä seuraa ohjelmiston uusi versio. Ohjelmiston uutta versiota käytetään ja sen tuloksia kerätään. Tuloksia arvioidaan ja niiden pohjalta tehdään muutosehdotuksia mikä taas johtaa johdon päätöksen tekoon. (Grupp ja Takang 2003, 80–81)

Ohjelmiston ylläpidon kannattavuus muuttuu ajan kanssa. Se jakautuu investointivaiheeseen, korkean maksun palautukseen ja vähenevien tuottojen vaiheeseen. Ensimmäisessä vaiheessa resursseja tarvitaan paljon ja saatu hyöty on matala. Tässä vaiheessa on esimerkiksi juuri julkaistu ohjelmisto, mikä vaatii paljon hätäratkaisuja ja parannuksia. Toisessa vaiheessa hyödyt ohjelmistosta kasvavat ja alkuperäiset ongelmat ja puutteet ratkaistaan. Käyttäjien pyytämiä parannuksia tehdään, dokumentaatio paranee ja tehokkuus kasvaa. Viimeisessä vaiheessa tuotot alkavat pienenemään, kun tuote on saavuttanut hyötynsä huipun, suurten muutosten kustannustehokkuus alkaa hiipua. (Grupp ja Takang 2003, 80)

Ohjelmistorobotiikalla toteutetaan prosesseja, joiden tarve on siten jatkuva. Silloin muutoksentarpeiden kustannustehokkuuden voidaan olettaa toimivan ohjelmistoista eroavalla tavalla. Rinnastus voidaan ehkä nähdä siinä, että jos prosessi esim. käytetyiltä ohjelmistoiltaan muuttuu huomattavasti, ei ainakaan olemassa olevan robotin valtava muuttaminen ole enää järkevää, vaan joko tarvitaan uusi toteutus tai lakataan käyttämästä robotiikka kyseiseen prosessiin.

Osbornen mallin ero muihin nähden on se, että se keskittyy suoraan realistiseen ylläpidon ympäristöön. Siinä missä muissa malleissa oletetaan ideaalinen ympäristö, tässä mallissa oletetaan ympäristössä sellaiseksi kuin se realistisemmin on. Ylläpidon mallina pidetään jatkuvia iteratiivisina ohjelmiston elinkaarina, joissa jokaisessa pyritään rakentamaan ylläpidettävyyttä. Jos hyvät ominaisuudet ovat jo olemassa, kuten tarkat määrittelyt ja täysi dokumentaatio, on tekemistä vähän, mutta muulloin annetaan lupa niiden rakentamisen. Mallissa on useita vaiheita ja tyypillisiä iteratiivisia askeleita. Hypoteesi on, että monet ylläpidon ongelmista johtuvat vajaasta johdon kommunikaatiosta ja hallinnasta. Parannusehdotuksia ovat ylläpidon vaatimusten sisällyttäminen muutosmäärittelyyn, laadun hallinta ohjelma laatuvaatimuksen, keinot varmistua siitä, että ylläpidon tavoitteet on saavutettu ja palaute suorituksesta johdolle. (Grupp ja Takang 2003, 83–84)

Iteratiivisen parantamisen mallissa oletetaan kattava dokumentaatio, sillä se on riippuvainen tästä joka iteraation alussa. käytännössä malli koostuu peräkkäisistä vaiheista, analyysistä, hahmotuksesta ja ehdotetuista muokkauksista sekä uudelleen suunnittelusta ja toteutuksesta. Joka kierroksen alussa olemassa olevaa dokumentointia, kuten vaatimuksia, suunnittelua, koodia testausta ja analyysijä, muokataan korkeimmasta tasosta alkaen sisältämään halutut muutokset. Nämä muutokset viedään läpi koko dokumentaatiosta ja systeemiä suunnitellaan uudelleen. Tämä malli tukee uudelleenkäyttöä ja mahdollistaa muut mallit kuten nopean korjauksen mallin. Ympäristön paineet usein pakottavat nopeisiin ratkaisuihin, mutta nämä saattavat johtaa jopa uusiin ongelmiin kuin ne ratkaiseva. Tässä mallissa nopeita muutoksia voidaan siis tehdä, mutta seuraavassa iteraatiossa ne otetaan laajemmin huomioon. Mallin heikkouksia ovat oletus kattavasta dokumentaatiosta, ja tiimin kyvystä ymmärtää olemassa olevaa kokonaisuutta. (Grupp ja Takang 2003, 80)

Uudelleenkäyttämisen mallin mukaan ylläpito voitaisiin nähdä toimintana, johon kuuluu olemassa olevien ohjelman osien uudelleen käyttö. Osien luokitteluun ja muokkauksiin tarvitaan yksityiskohtainen runko. Täydessä uudelleenkäytön mallissa lähtöpiste voi olla mikä tahansa vaihe ohjelmiston elinkaarta. Mallin keskiössä on siis uudelleen käytettävien komponenttien kirjasto. Vanhoista toteutuksista tallennetaan vaatimusten määrittelyä, suunnittelua lähdekoodia ja testidataa, ja näitä voidaan sitten uudelleenkäyttää uuden järjestelmän luomiseen. (Grupp ja Takang 2003, 85–86)

Uudelleen käyttämällä ohjelmiston osia vaaditaan vähemmän aikaa ja vaivaa määrittelyyn, suunniteltuun ja toteuttamiseen. Uudelleenkäytön mallissa uusien toteutuksien laatu tapaa parantua, sillä sen osia on jo laajasti testattu. Uudelleenkäyttö johtaa luotettavampia, eheämpiin ja laadukkaampiin tuotteisiin. (Grupp ja Takang 2003, 154)

Uudelleenkäytön hyödyt robotiikassa ovat ilmeiset. Tietyillä organisaatioilla on käytössä tietyt järjestelmät ja prosessit, ja kun eri prosesseja toteutetaan samalla järjestelmällä, on näissä varmasti yhteneväisyyksiä.

### **3.5 Ohjelmistorobotiikan skaalaus**

Tämän työn keskeinen tarkoitus on siis löytää toimintamalli, jolla robotiikan hyötyjä saadaan kasvatettua organisaatiossa. UiPath on kertonut ja markkinoinut skaalaamisesta laajasti (UiPath 2020A). Skaalaaminen vaatii teknisten määritysten lisäksi mallin myös ympäröivään liiketoimintaan, kun kommunikaation määrä robottien kehityksestä, seurannasta ja ylläpidosta kasvaa. Kaikkeen tähän tarvitaan jopa organisaation laajuisia muutoksia. viestintään perehdytäänkin luvussa 3.

Moe et al. (2010, 165–180) kuvaa suunniteltua yrityksen muuntumista ketterään kehitykseen ohjelmistokeskeisissä systeemeissä, mutta sen löydökset vaikuttavat yhtä lailla sopivan robotiikan skaalaamisen tarpeisiin. Avaintekijöitä onnistumisen ovat omistautuneet ja tuloksista vastaavat onnistumiselle kriittiset sidosryhmät, onnistumisperusteinen neuvottelu ja valinta, asteittainen kehitys ja kasvu systeemin määrittelylle ja sitouttamiselle, iteratiivinen kehitys ja määrittely, ajankohtainen tilannekuva ja sen kehitys sekä riskienhallinta (Moe et al.

2010, 168–169). Jotta pystytään jatkuvasti tarjoamaan uusia laadukkaita toteutuksia, tarvitaan vakaa, mutta joustava ohjelmistoarkkitehtuuri, jatkuva sidosryhmien sisällyttäminen, muutosten sisällön hallinta, tiheät tilannekatsaukset ja omistautuneet vastuuhenkilöt tekemään oikein ajoitettuja päätöksiä, dokumentointia ja jatkuvaa varmistamista (Moe et al. 2010, 179). Täydellistä mallia ei kuitenkaan voida luoda, joka pystyisi vastaavaan lukuisiin eri järjestelmiin nyt ja tulevaisuudessa (Moe et al. 2010, 180).

UiPathin (2020a, 3) mukaan ohjelmistorobotiikan kehityskulku organisaatiossa jakautuu vaiheisiin. Matkan alussa puhutaan todennusvaiheesta. Tässä vaiheessa teknologiaa lähinnä testataan, jotta nähdään, toimiiko se organisaation ympäristössä. Usein perustetaan ryhmä, joka tutustuu tekniikkaan ja joitakin varhaisia tuloksia nähdään. Tämä ensimmäinen vaihe kestää tyypillisesti noin puoli vuotta ja on usein vain yksittäisen osaston taikka sijainnin aloittama. Suurimmassa osassa organisaatioita tätä alkuvaihetta seuraa vaihe, jossa tehdään useita toteutuksia joko organisaation omin voimin tai ulkopuolisella avulla. Ne kohdistuvat usein prosessiryppäeseen eli kytköksissä oleviin prosesseihin tietyssä osastossa tai sijainnissa. Ideaalisesti tämä vaihe kestää 9 kuukaudesta vuoteen. Tästä vaiheesta eteneminen muodostuu monelle organisaatiolle ongelmalliseksi, sillä ei tunneta, miten suuren mittakaavan skaalausta saadaan aikaiseksi ja robotiikan kehittäminen alkaakin hidastumaan estäen siten hyödyt, joita laajempi automaatio voisi tuoda.

Ohjelmistorobotiikan toteuttaminen perustuu usein tietyn tiimin intensiiviseen projektityöhön, mitkä saattavat vaatia enemmän aikaa ja suurempia kustannuksia. Ideaalisesti robotteja voidaan implementoida niin, että niiden mahdollista myöhempää laajentamista otetaan huomioon. Tämänlainen skaalaaminen mahdollistaa tehokkaammat toteutukset, kun olemassa olevaa robottia voidaan käyttää uudessa yhteydessä ilman suuria lisäkustannuksia. (Rutschi & Dibbern 2020, 454)

Toisaalta, vaikka RPA:n avulla prosessien automatisointi on nopeampaa ja helpompaa, on siihen tarjolla muitakin tekniikoita, joista organisaatioiden tulee hyödyntää useampaa. Voi myös olla, että ohjelmistorobotiikka nähdään vain väliaikaisena ratkaisuna, mikä suuresti vaikuttaa sen asemaan. Joka tapauksessa robotiikkatoteutusten hallinnasta ja valvonnasta voi

tulla haastavaa, kun ketteriä RPA toteutuksia luodaan. Niin pitkän kuin lyhyen aikavälin vaikutuksia on siis ajateltava. (Hofmann et al. 2020, 103)

Jotta skaalaaminen onnistuu, on organisaation hallittava kuusi osa alueella (UiPath 2020a, 3):

- ylemmän johdon kiinnostus ja tuki
- osaamisen määrä ja laatu
- metodit automatisaation kohteiden löytämiseen ja priorisointiin
- rahoituksen tasot ja lähteet
- automaatio ohjelman laajuus ja lähestymistapa
- automaation operointimalli

Ylemmän johdon tukea tarvitaan toimivaan RPA ohjelmaan. Ylempään johtoon tässä kuuluu muun muassa liiketoiminta-alueiden johtajat, henkilöstöjohtaja ja talousjohtaja. Tärkeitä sidosryhmiä ovat tukipalvelut, IT ja sisäinen auditointi. Näihin kaikkiin on tärkeää olla yhteydessä jo varhaisessa vaiheessa, kun ensimmäiset hyödyt robotiikasta on saatu todettua. Mahdollisesti tärkeintä on liiketoiminta-alueiden johdon tuki, sillä he ovat ensisijaisesti automaatiosta hyötyvä taho. Usein tukipalvelut ovat helposti robotiikan kannalla, sillä nämä toteuttavat tyypillisesti paljon hallinnollista työtä. Kuitenkin kaikkien tahojen on ymmärrettävä mistä RPA:ssa on kyse. Jotta robotiikkaa aletaan todella hyödyntää, tarvitaan liiketoimintaan yksikkökohtaisia automaatio päämääriä. Tämä taas edellyttää, että talousjohtaja määrittää osaltaan näitä tavoitteita. IT-tukea tarvitaan, jotta robotiikka voidaan integroida organisaation laajempaan systeemiarkkitehtuuriin ja määränpäihin. (UiPath 2020a, 5)

Pääasiassa kolmenlaista osaamista tarvitaan robotiikan skaalaamisessa. Näitä ovat liiketoiminta, RPA-teknologia ja yleinen IT osaaminen. Liiketoiminnan osaaminen jakaantuu substanssiosaajaan, esimerkiksi prosessien asiantuntijaan, ja RPA business analysoijaan, joka toimii siltana subjektiosaajan ja teknisen toteuttajan välillä. Hän osaa muotoilla prosessin tarkasti teknisin termein, kerätä määrittelyitä ja ymmärtää mitä ohjelmistorobotiikalla voidaan tehdä ja mitä ei. Lisäksi tarvitaan automaatioprojektien johtaja, joka yhdistää vahvasti teknistä liiketoiminta ja projektinjohto-osaamista. Teknisiä rooleja ovat ratkaisuarkkitehti ja RPA-kehittäjä. Ensimmäinen omaa syvän automaatio ymmärryksen ja toimii vahvasti business

analysoijan kanssa, sekä varmistaa RPA-koodin teknisen laadun. RPA kehittäjä luo robotin, osallistuu testaamiseen ja vastaa ylläpidosta. IT-osaamista vaaditaan infrastruktuurin ja tietoturvan ymmärtämiseksi. Käytännössä nämä eri roolit voivat varsinkin robotiikan kehityksen alkuvaiheessa kuulua samoille henkilöille. Robotiikan skaalautuessa tarvitaan osaamista lisää. RPA-kehittäjiä löytyy usein niin teknologia osaavista business käyttäjistä, kuin IT-asiantuntijoista. Lean osaaminen voidaan helposti muuntaa RPA business analysoijan osaamiseksi. (UiPath 2020a, 6–7)

Automaatiokohteiden löytäminen ja priorisointi voi osoittautua vaikeaksi organisaatiolle. Alkuvaiheessa lupaavien prosessien löytäminen on intuitiivista, sillä säännönmukaiset työläät ja turhauttavatkin prosessit ovat usein tiedossa. Kun nämä automaatiot ovat toteutettu, tilanne muuttuu kuitenkin vaikeammaksi. Tässä vaiheessa saattaa esiintyä myös muutosvastarintaa, jos työntekijät alkavat kokea automaation uhaksi. Tilanne voidaan ylittää asteittain. Ensimmäisten toteutusten jäljiltä on toivon mukaan syntynyt hyviä tuloksia kasvaneella tuottavuudella tai asiakastytyväisyydellä. Seuraavaksi tulee järjestelmällisesti analysoida ja priorisoida automaatiokohteita. Menetelmiä on erilaisia, mutta ne pääasiassa ottavat huomioon potentiaalisen hyödyn ja vastapainona prosessin kompleksisuuden. Hyödyt -pääasiassa ajan säästö- riippuvat volyyymista, käytön tiheydestä ja varsinaisesta prosessin läpivientiajasta. Toteutuksen kompleksisuuteen vaikuttaa muuttujat, kuten vaiheiden määrä, poikkeustilanteiden määrä ja käytetyn datan rakenne. Nämä yhdistämällä löydetään parhaat automaatiokohteet. Tätä kutsutaan ylhäältä alaspäin lähestymistavaksi, joka onkin tärkeä ensiaskel. Tämä ei kuitenkaan riitä, vaan seuraavassa vaiheessa automaatioideoita halutaan kerätä työntekijöiltä. Tätä kutsutaan alhaalta ylöspäin lähestymistavaksi. Yhdistämällä nämä lähestymistavat saavutetaan tehokkuutta skaalaamiseen. (UiPath 2020a, 6–7)

Syed et al. (2020, 5) listaa kirjallisuudesta ominaisuuksia, jotka lisäävät prosessin soveltuvuutta robotiikalle. Näitä ovat mm. sääntöpohjaisuus, volyyymi eli toistuvuus, tulosten havaitsemisen helppous. Kaiken käytetyn datan tulee olla digitaalista ja rakenteellista. Lisäksi standardointi, dokumentointi ja vähäiset poikkeustapaukset edesauttavat robotiikka toteutuksia. (Syed et al. 2020, 5)



RPA-hankkeiden rahoittaminen on tärkeässä roolissa automaatiokehityksen jatkuessa. Alkuvaiheessa organisaatiot useimmiten käyttävät prosessikohtaista investointilaskelmaa ja priorisoivat projekteja tämän mukaisesti. Skaalaamisen mahdollistamiseksi on kuitenkin siirryttävä laajempaan hyötylaskemaan esimerkiksi liiketoiminta-aluekohtaisesti, sillä nämä itsessään kustannustehokkaat projektit voivat käydä vähiin. Portfoliotasolla tarkasteltuna tuottavuutta voidaan kuitenkin lisätä. Kuvassa 3 on hahmotettuna tämä portfolioefekti. (UiPath 2020a, 10)

| TIIMI A    |  |  |
|------------|--|--|
| Tehtävät   | Automaatiolla saavutettava ajan säästö | Prosessin viemä osuus tiimin työajasta |
| Prosessi 1 | 100 %                                  | 10 %                                   |
| Prosessi 2 | 50 %                                   | 20 %                                   |
| Prosessi 3 | 25 %                                   | 20 %                                   |
| Prosessi 4 | 25 %                                   | 20 %                                   |
| Prosessi 5 | 25 %                                   | 30 %                                   |

| TIIMI B    |  |  |
|------------|--|--|
| Tehtävät   | Automaatiolla saavutettava ajan säästö | Prosessin viemä osuus tiimin työajasta |
| Prosessi 1 | 100 %                                  | 10 %                                   |
| Prosessi 2 | 50 %                                   | 20 %                                   |
| Prosessi 3 | 25 %                                   | 30 %                                   |
| Prosessi 4 | 25 %                                   | 20 %                                   |
| Prosessi 5 | 25 %                                   | 20 %                                   |

**Kuva 3** Portfolioefekti mukailten UiPath 2020A, 10

Kuvassa 3 ylempänä on kuvitteellinen tiimi A, jonka tehtävänä on hoitaa 10 prosessia. Näitä prosesseja on tarkasteltu tapauskohtaisesti ja vain kaksi, joissa automaatiolla voidaan säästää 100 ja 50 % ajasta on päätetty automatisoida. Tällöin tiimin kokonaista työmäärästä on kuitenkin saatu automatisoitua 20 %. Sen sijaan alempana on kuvattu tilannetta tiimillä B, jossa mahdollisimman moni prosessi olisi automaatioavustettu huolimatta matalasta

automaatiotasosta. Tällöin kuitenkin kokonaisajasta on saatu säästettyä 37,5 %:ia. Kummallekin tiimille voitaisiin kuitenkin antaa uusia, ihmiselle sopivampia tehtäviä (UiPAth 2020a, 10)

Kärjistäen automatisaatio voidaan siis lähestyä joko ylhäältä alaspäin tai alhaalta ylöspäin taktiikoilla. Usein organisaatiot aloittavat hyödyntämään robotiikkaa ensimmäisellä tavalla, jolloin osoitettu tiimi toteuttaa selkeämpiä tuottoisimpia ja helpompia automaation kohteita. Pidemmän päälle, jotta kehitys voi kuitenkin jatkua, tarvitaan myös toista lähestymistapaa. Tällöin teknologiaorientoituneet työntekijät voivat opetella käyttämään robotiikkaa, kehittämään omia ratkaisujaan ja esittämään ideoita. Automaation kehittäminen pääasiallisen työn ohella on kustannustehokasta erityisosajien palkkaamiseen verrattuna ja niin voidaan päästä yli rajallisista resursseista. Lisäksi alhaalta ylöspäin lähestymistapa auttaa saamaan koko organisaation mukaan muutokseen, kun jokainen pystyy osallistumaan siihen. (UiPAth 2020a, 11–12)

Valtaosa yrityksistä aloittivat ohjelmistorobotiikkatoimintaansa ylhäältä alaspäin menetelmällä mikä merkitsi aluksi suurempia kustannuksia työntekijöihin lisenseihin koulutuksen ja kehityksen. Näillä yrityksillä oli selkeä strategia RPA:n edistämiseen. Toisella yrityksellä lähestymistapa oli maltillisempi, jolloin työntekijät kokeilivat teknologiaa ja eri käyttökohteita sille yrityksillä vaikuttaa olevan suuri kiinnostus kehittää sen sisäisiä robotiikkaa kyvykkyksiä kehitykseen kustannuksien vähentämiseksi. (Poussa 2020, 56)

Ohjelmistorobotiikalla voidaan saavuttaa eritasoista automaatiota. Yksinkertaisella tasolla automatisoidaan yksittäisiä rutiinitehtäviä, joita työntekijää voivat käyttää työasemallaan. Seuraavalla tasolla automatisoidaan kokonaisia prosesseja, joissa on monia vaiheita ja joihin osallistuu useampia henkilöitä, mutta nämä ovat yhä usein vain tietyn osaston sisällä. Korkeammalla tasolla tehdään yrityksen laajuista automaatiota, kuten monimutkaisia osastojen välisiä prosessien automaatiota ja voidaan hyödyntää erilaisia tekniikoita kuten tekoälyä. Alhaalta ylöspäin lähestymistapa usein keskittyykin tämän yksinkertaisemman tason automaatiotehtäviin, kun taas ylhäältä alaspäin suunnittelee korkeamman automaatiotason ratkaisuja. Kun nämä lähestymistavat yhdistetään, saadaan koko organisaation kattavaa automaatiota (UiPAth 2020a, 4 & 9)

Ohjelmistorobotiikkaan tarvitaan selvä toimintamalli. Ohjelmistorobotiikan käyttöönotossa korostuu vastuunjako kohteiden määrittely oikein valittu teknologia henkilöstön osaaminen ja sen kehitys (Kauppila 2020, 74). Usein automaation alkuvaiheessa on järkevintäkin, että robotiikasta vastaa yksi erikoistunut tiimi, jolla on kokonaisvaltainen rooli automaation edistämisessä, esimerkiksi toteutuksista kaikille liiketoiminnan osa-alueille. Pidemmän päälle ohjelman kasvaessa on tämän tiimin roolin kuitenkin muututtava. Tyypillisiä malleja on kaksi. Joko alkuperäinen RPA tiimi siirtyy puhtaasti neuvonantamisen rooliin tai keskittyy olemassa olevien automaatioiden operointiin. RPA tiimi saattaa myös jakaantua useampiin sijainteihin tai pysyä yhtenäisenä keskuksena organisaatiolle. Ohjelmistorobotiikan on myös sovittava kokonaisvaltaiseen yrityskulttuurin rakenteeseen. Keskeistä on määrittää miten tehtävät jakautuvat RPA-tiimeille, liiketoiminta-alueille sekä IT:lle. (UiPAth 2020a, 13)

Kun kuusi osa-aluetta on tarpeeksi edistyneitä, voidaan varsinainen skaalaaminen aloittaa. Tämä jakaantuu kolmeen vaiheeseen. Ensimmäisenä kannattaa valita tietty liiketoiminnan alue, jolla automaatio viedään kokonaisvaltaiselle portfoliotasolle. Tietyn osaston valinnalla on monia etuja. Tulokset konkretisoituvat nopeammin ja kun kokonaisvaltainen automatisaatio on saatu järjestettyä tässä osastossa, voi se karttuneen tiedon ja oppimisen myötä jatkaa kehitystä melko itsenäisesti. Automaation potentiaalin osoittamiseksi tulee myös pilotoida laajoja ja monimutkaisia osastojen välisiä prosesseja. Etuja on mm. osastojen välisen yhteistyön ja automaation näkyvyyden lisääntyminen organisaatiossa. Viimeiseksi on koulutettava lukuisasti RPA kehittäjiä ja business analysteja. Vain tällä päästään yli keskeisestä pullonkaulasta erikoistuneen osaamisen rajallisuudesta. Onnistuneessa skaalauksen jäljiltä organisaatiossa on voimakas RPA keskittymä, automaatiosta innostuneet osastojen johtajat, jotka ovat saavuttaneet suuria etuja automaatiosta. On opittu täysivaltaisesta automaatiosta ja saatu huomattavasti lisää ohjelmistorobotiikkaosaamista tulevaisuuta varten. (UiPAth 2020a, 15–16)

RPA:n skaalaaminen on hyvin tavoitteen mukaista ja siitä saataisiin suuria hyötyjä, mutta sille on myös lukuisia haasteita. Tyypillisiä haasteita ja toisaalta vaatimuksia onnistumiseen ovat esimerkiksi suunnittelu, kustannuslaskelmat, RPA:n osaamiskeskittymä ja IT:n tuki. Automaatiolla tulee olla selkeät tavoitteet ja dokumentaatiota. Tämä on tärkeää prosessin jokaisessa vaiheessa, ja mikäli näin ei ole, seuraa helposti kaaosta ja sekaannusta, kun robottien

määrä kasvaa. Jos laskemissa keskitytään liiaksi sijoituksen palautukseen voi RPA:n skaalaaminen tulla hankalaksi, sillä vain prosessit, joissa nämä kriteerit täyttyvät automatisoimalla ei saavuteta yhtä suuria etuja, kuin portfolioefektillä. Prosessi, joka maksaa itsensä takaisin lyhyessä ajassa, kuulostaa hyvältä, mutta näitä on rajallisesti. Kun asiaa tarkastellaan laajemmin, myös prosessit, joissa yksittäisvaikutus ei ole suuri, mutta käyttö laajaa, voi ajan säästö ja siten tuottavuuden kasvu olla merkittäviä. (Taulli 2020, 190–191)

RPA-tiimi ja -osaaminen ovat keskeisiä skaalaamisen onnistumiseksi. Robotteja tulee tehdä hallitusti ja yhtenäisesti, ja tilannetta pitää aktiivisesti tarkkailla. Ohjelmistorobotiikka tarvitsee toimintamallin ja robotiikkatiimin onkin hyvä koostaa oppimistaan ja parhaita toimintatapoja. Koulutuksen ja harjoituksen tulee olla jatkuvaa. Usein IT ei ole vahvasti mukana ohjelmistorobotiikan alkutaipaleella organisaatiossa. Tämä selittyy osittain RPA:n ”helppoudella”. Sitä ei nähdä osana varsinaista IT:tä. Kun IT-osasto ei kuitenkaan ole riittävästi tukena, voi se aiheuttaa esteitä kuin automaatio ohjelmaa kasvatetaan. IT-arkkitehtuuri pitää siis saada mukaan ja pelaamaan yhteen robotiikan kanssa. (Taulli 2020, 190–191)

Näiden lisäksi tarvitaan johdon tuki. Vain siten saadaan riittävät resurssit ja tahtotila tavoitteisiin pääsemiseksi. Johdon pitää olla myös samaa mieltä RPA:n tavoitteista. Robotiikkatiimi ei myöskään voi tehdä kaikkea itse, vaan työyhteisö ja etenkin robotiikan sidosryhmät on saatava vahvasti mukaan. Tämä vaatii opastusta, koulutusta, teknistä ja muuta tukea. Lopuksi myös paikallisuus pitää ottaa huomioon. Globaaleilla järjestöillä on paikallisia uniikkeja kulttuureja, normeja kuin myös lakeja infrastruktuuria ja muita sääntöjä ja ohjeita. Nämä kaikki tulee ottaa huomioon myös robotiikkatoteutuksissa. (Taulli 2020, 190–191)

Skaalaamiseen keskiössä on siis luoda malli, joka RPA-tiimiä laajemmassa ympäristössä kertoo, miten voidaan onnistuneesti tehdä yhteistyötä tiimien kesken automaation edistämiseksi organisaatiossa. Se koostuu kuudesta tekijästä: tavoitteista ja suoritettavasta suunnitelmasta, arvon mittaamisesta, liiketoiminnan yksiköiden laajentamisesta, organisaation rakenteesta, toteutusten laadusta sekä valvonnasta ja ohjauksesta. (UiPath 2020b)

## 4 KOMMUNIKAATIO ROBOTIIKASTA

Käyttäessä ohjelmistorobotiikkaa ja suunniteltaessa sen skaalaamista sitoutuu se vahvasti laajempiin teemoihin ja teorioihin kommunikaatiosta, kulttuurista ja muutosviestinnästä. Robotiikan ja yleisestikin digitalisaation aikaansaamat muutokset organisaatiossa voivat olla hyvinkin laajoja, ja verrattuna näihin teknologiset vaatimukset ja haasteet voivat olla hyvinkin pieniä. Siksi on syytä perehtyä näihin inhimillisiin vaikuttajiin, jotka todella mahdollistavat muutoksen tai voivat asettua hanakasti sitä vastaan.

Organisaation kommunikaatiota voidaan katsella kolmelta kantilta. Yksi on keskittyä kommunikaation kehittymiseen kommunikaation alueina ja kommunikaation assosiaatioita. Toinen näkökulma on määrittää organisaation kommunikaatio kommunikaatioksi, joka tapahtuu organisaatioissa. Tämä on hyvin tyypillinen ja yleinen lähestymistapa. Kolmas tapa on ajatella kommunikaatio keinona kuvata ja selittää organisaatioita. Kommunikaatioteoriaa voidaan käyttää selittämään sosiaalisia rakenteita, psykologisia tiloja, jäsenkategorioita, tietoa jne. sen sijaan, että se olisi tietty rajattu ilmiö organisaatiossa. Organisaation kommunikaation voidaan nähdä jakautuvan toimintaan, suhteisiin, sekä päätöksiin, sillä kommunikaatio johtaa toimintaan, vaikkakaan ei välttämättä suoraan tiettyyn toimenpiteeseen (Dov Te'eni 2001, 254). (Jablin et al. 2000, 4–5)

Organisaatiot hankkivat, tulkitsevat ja kontrolloivat tietovirtoja ympäristöstään selviytyäkseen; tunnistaakseen uhat, tarttuakseen mahdollisuuksiin ja hallitakseen riippuvuuksia resurssien ja sidosryhmien kesken (Jablin et al. 2000, 197). Organisaatiolla on monia ristiriidassa olevia päämääriä: lukuisia sidosryhmiä, eri intresseistä ja jatkuvasti muuttuva joukko ongelmia ja haasteita. Jotta näiden päämäärien, sidosryhmien ja ongelmien kanssa pärjätään, tulee tasapainotella siis sidosryhmien vakuuttamisen sekä näiden kanssa neuvottelun välillä. Organisaatiot eivät vain sopeudu ympäristönsä, vaan pyrkivät myös proaktiivisesti muokkaamaan sitä kommunikaatiolla. Pyritään siis jakamaan organisaation tapaa määrittää tilannetta ja ongelmia ympäristössä. (Jablin et al. 2000, 215–216)

Rakenteellinen kommunikaatio on tekijä suurimmassa osassa organisaation kommunikaatiota. Se perustaa, selittää ja vahvistaa organisaation rakennetta. Prototyypinen esimerkki on

organisaatiokaavio. Rakenteellisella kommunikaatiolla on kolme tärkeää ominaisuutta. Ensinnäkin se on suoran kommunikaation korvaaja. Toiseksi rakenne on metakommunikaation muoto jäsenen suhteesta organisaatioon. Rakenteet symboloivat sitä, mitä organisaatiot arvostavat. Kolmanneksi rakenteet eivät toimi samalla tavalla kaikkialla organisaatiossa. Onnistuneen rakenteen tarkoituksena on vähentää tällaista kommunikaatiota siten parantaa tehokkuutta, kuitenkin samalla heikentämättä päätöksenteon laatua (Jablin et al. 2000, 529–530)

Erikoistumisesta on hyviä ja huonoja puolia. Organisaation kannalta tietyn alan asiantuntijat on luonnollista järjestää yksiköihin yhteistyön helpottamiseksi. Jos kaksi organisaatiota esimerkiksi tekevät yhteistyötä, on vastaavan alan asiantuntijat suhteellisen helppo saada tekemään yhteistyötä. Toisaalta asiantuntijoiden erikoisosaamisella on huomattavaa valtaa. Sille voidaan jopa hiljentää niitä, joilla ole elä tätä tietämystä. Ei-asiantuntijalla saattaa olla hyvä ymmärrys ongelmasta ja mahdollisesta ratkaisusta siihen, mutta jos termit eivät sovi tietynasiantuntijaryhmän termistökehikkoon, saattaa nämä näkemyksensä jäädä huomiotta. Kuten kaikkea tietoa asiantuntemusta ja osaamista syntyy lisää, kun asiantuntija tekee yhteistyötä muiden kanssa; he oppivat tätä erityistermistöä. (Jablin et al. 2000, 531)

Monet tekijät vaikuttavat organisaation rakenteeseen: teknologia, ammatit, kilpailutilanne ja paikallinen kulttuuri esimerkiksi. Rakenteen toimivuus riippuu organisaation toimijoiden teoista. Erityisesti informaatioteknologia on muuttanut ja muuttaa organisaation rakenteita (Jablin et al. 2000, 532)

Ohjelmistorobotiikan yleistymisestä organisaatiossa luottamuksella on suuri merkitys. Käyttäjien ja informaatioteknologia välillä luottamusta on tutkittu jo laajasti ja informaatiojärjestelmäteorioissa se onkin keskeisessä asemassa. Kuten muut yleistyvät digitalisaation teknologiat, ohjelmistorobotiikka kärsii rajatusta käyttäjien luottamuksesta luvattujen ominaisuuksien kuin myös laajempien sosiaalikultuuristen suuntausten suhteen. (Syed & Wynn 2020, 149)

Uuden teknologian käyttöönotto organisaatiossa aiheuttaa usein uusia haasteita. Jotta organisaatiot voivat onnistuneesti ottaa käyttöön RPA-teknologiaa on työntekijöiden tärkeää

hyväksyä robotit ikään kuin digitaalisina kollegoina. Vain näin toiminta voi olla saumatonta. Kuitenkin tämän luottamuksen saavuttaminen on yksi keskeisistä haasteista ohjelmistorobotiikan yleistymiselle. Suoraan teknologian lisäksi luottamusta tarvitsee rakentaa operoivien työntekijöiden ja ohjelmistorobotiikan suunnittelijoiden välille. Keskeisten osapuolten kanssa on tärkeää kommunikoida aktiivisesti rakentaa luottamusta. Viestintä on kriittinen tekijä ohjelmistorobotiikan käyttöönotossa. hyvällä viestinnällä vähennetään muutosvastarintaa ja se vaikuttaa työntekijöiden asenteisiin organisaatiossa. (Mattila 2019, 56) Itse robotin käyttöönotossa tarvitaan paljon teknistä tukea ja ennen varsinaista julkaisua tarvitaan valtavia määriä laadun varmistusta ja toiminnan seuranta. (Syed & Wynn 2020, 147–150)

Astani et al. (2020, 151) mukaan luotto robotiikkaan, kuten uusiin teknologioihin yleensä, rakentuu neljästä tekijästä: kyvykkyyksiin, hyväntahtoisuuteen, toimintavarmuuteen ja kontekstiin, engl. Abilities, Benevolence, Integrity, Context. Ensimmäinen jakaantuu vastuullisuuteen, informaation tarkkuuteen ja luotettavuuteen. Hyväntahtoisuudella tarkoitetaan sallittua datan käyttöä, suunnittelijan hyväntahtoisuutta, käyttäjän ymmärrystä ja uskoa. Toimintavarmuus viittaa ennustettavuuteen, luotettavuuteen, datan eheyteen ja käyttäjän varmistamiseen. Viimeisellä viitataan strategisiin tarpeisiin käytäntöihin ja riskeihin. (Syed & Wynn 2020, 151)

Ylipäättään henkilöön kohdistumaton luotto voi lisätä työntekijät sitoutuneisuutta organisaationsa. Jos työntekijät luottavat organisaatioon, jossa työskentelevät he voivat luottaa tulevaisuutensa siinä jopa silloin kun suoraan kollegoilta ja esimiehiltä ei saada riittävästi tukea. Eriyisen tärkeää tällaisen luottamuksen muoto on silloin kun mahdolliset kasvokkain puheluun ja henkilöiden välisen luottamuksen rakentamiseen ovat rajallisia. etenkin tieto työskentely on tyypillisesti monimutkaista epävarmaa ja riskialtista, ja sitä suoritetaan kasvavissa määrin teknologia avusteisesti, projekteina ja virtuaalisesti. Johtajilla saattaa olla myös asiantuntijatehtäviä ja sitten vain osa ajastaan käytettävissä esimiehen rooliin. (Vanhanen et al. 2016, 12–13)

RPA toimittajien markkinoinnin, liiketoimintayksiköiden tavoitteiden ja johdon ansiosta robotin käyttäjän odotukset ovat aluksi korkealla. Jos robotti kuitenkin lakkaa toimimasta,

kuten se voi tehdä useasta syystä, turhauttaa tämä käyttäjää huomattavasti ja vähentää tämän luottoa robotiikkaan. Robotiikka edellyttää laadukasta ja hyvin määriteltyä dataa toimiakseen kunnolla ja mikäli tämä ei täyty on käyttäjän usein käytettävä ylimääräistä aikaa datan korjailuun. Esim. ihmiskäyttäjälle irrelevantit poikkeavuudet kirjoitusasussa muodostuvat robotille ongelmaksi ja tällaiset ongelmat voivat vähentää käyttäjien luottoa robotiikkaan. Tekemisen näkyvyys viittaa siihen, miten robotti käsittelee prosessiin liittyvää dataa. Ihmisten toteuttaessa tehtäviä on helppo kommunikoida niiden edistymisestä, mutta robotilla vastaavan tehtävän edistyminen voi olla vaikeampaa seurata, sillä ennen raportointia ei nähdä millaisia hakuja yms. toimenpiteitä tehdään. (Syed & Wynn 2020, 152–153)

Suunnittelijan osaamisesta riippuu, miten hyvin robotti saadaan toteuttamaan käyttäjän vaatimusten mukainen tehtävä. Robottia saatetaan syyttää epäonnistuneesta tehtävästä, vaikka todellisuudessa syyt ovat puutteellisissa määrittelyssä ja robotin suunnittelussa, molemmat ihmisten aikaansaannoksia. Vahva yhteistyö loppukäyttäjän kanssa ja sidosryhmien osallistuminen ottaminen robottien suunnittelu ja toteutusvaiheessa ovat siksi tärkeitä. Puuttuva suunnittelu ja määrittely ovat omiaan lisäämään käyttäjien turhautumista. Suunnittelussa analyysissä on otettava huomioon prosessin kompleksisuus -sen eri vaiheet ja vaikutukset eri ryhmien ja järjestelmien välillä. Prosessin konteksti, kuten liiketoiminta, on tärkeää ymmärtää hyvin prosessia suunnitellessa. Teknistä tukea on oltava saatavilla, kun robotissa ilmenee ongelma. (Syed & Wynn 2020, 153)

Robotin vaatiman organisaation datan käytön on oltava luotettavaa ja turvallista, vaikka tämä ei ehkä olekaan käyttäjien keskeinen huoli. Robotin pääsyoikeudet järjestelmissä on kuitenkin rajattu sen käyttäjätilillä. Jaettua työtä robotin kanssa tehtävien suorittamiseksi pidetään helposti lähestyttävänä, mutta tällöinkin teknistä tukea tarvitaan ongelmien ratkomiseksi. (Syed & Wynn 2020, 154)

Yksi robotiikkaan liittyvistä huolista on työn menettämisen pelko, ja tämä voikin olla yksi keskeinen syy muutosvastarinnalle. Johdon päämääränä automaation ja sitten myös robotiikan kasvattamiselle voikin olla kustannusten vähentäminen. Kilpailu alalla painostaa laajaan ohjelmistorobotiikan käyttöönottoon kilpailuedun saavuttamiseksi. Painotus tuottavuuden kasvattamisen merkitykseen on kuitenkin luottamusta herättävämpää, kuin kustannusten



minimointi. RPA:n strateginen tavoite voikin olla kasvattaa työntekijöiden kyvykkyyksiä ja suoritustehokkuutta. Epäselvä strategiset tavoitteet robotiikan suhteen ovatkin omiaan vähentämään luottamusta, kun taas selkeät toimivat päinvastoin. Ylimmän johdon tuki on siis ajavat tekijä myös luottamuksen kasvattamiseen. (Syed & Wynn 2020, 154–156)

Johdon tuki on siis tärkeää muutostilanteissa ja etenkin se on keskeinen tekijä edistämään työntekijöiden valmiutta muutokselle. Erityisesti työntekijöille, jolloin korkea tai puussa muutos vastarintaan on tärkeää puhua muutoksesta esimiestensä kanssa. sitten johtamistyyli, mikä mahdollistaa kommunikaation ja luottamuksen alaisten ja johdon välillä tarvitaan. Muutosvastarintaa ei siis tulisi nähdä esteenä muutokselle vaan mahdollisuutena kommunikaation johdon kanssa, joka mahdollistaa onnistuneen muutoksen. (Tanner & Otto 2016, 2198)

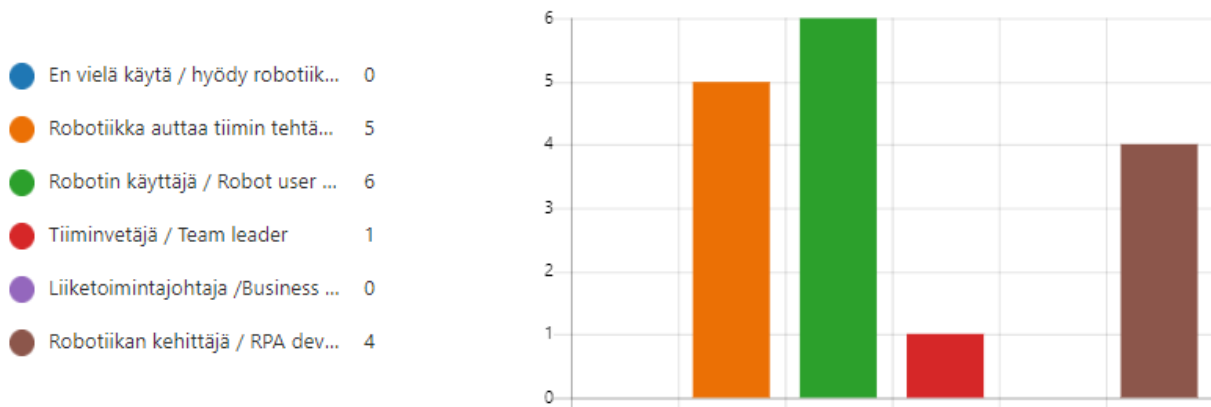
Myös hyvän IT-infrastruktuurin ja teknisen tuenvaraan voidaan rakentaa tuottavampaa ja luotettavanpa ohjelmistorobotiikkaa. Vaihtelu datakanavissa ja käytännöissä, kuten dokumentaation muodossa aiheuttaa helposti ongelmia robotiikkaan. Siten tämän yhtenäistäminen vaikuttaa positiivisesti robotiikan toimivuuteen ja luottoon. Kaiken kaikkiaan laaja ymmärrys siitä, mihin robotiikka pystyy ja mihin ei edesauttaa luottamuksen ja toimivan järjestelmän rakentamista. (Syed & Wynn 2020, 154–156)

RPA voi automatisoida prosesseja, jotka ylipäättään mahdollistavat liiketoiminnan ja siten suuresti vaikuttaa tähän. Organisaation strategian tulee siis ottaa huomioon niin suorat kuin epäsuorat robotiikan vaikutukset. Vaikutukset esimerkiksi ihmisten työskentelyyn, prosessien kokonaisuuteen ja tietojärjestelmien ympäristöön ovat merkittäviä. Yrityksen tulee toteuttaa jalkautussuunnitelma, jossa kaikki asianomaiset, kuten eri liiketoimintayksiköt, huomioidaan, sillä monet käytännön ongelmat johtuvat suuriltaosin peruskäsityksen puuttumiseen (Heijari 2020, 80). (Hofmann et al. 2020, 103)

## 5 ROBOTIIKAN ARVOT

Työssä on siis lähdetty hakemaan toimintamallia ohjelmistorobotiikan ylläpidolle ja skaalaamiselle organisaatiossa, jossa robottitoteutuksia on jo paljon ja niitä on tarkoitus tehdä lisää. Seuraavissa luvuissa tämän mallin teknisiä seikkoja sekä kommunikaatiopuolta on pohdittu erikseen, mutta käytännössä ne ovat sidottu vahvasti yhteen. Tämän kokonaisuuden kannalta hyvä lähtökohta oli selvittää, mikä ohjelmistorobotiikassa on ylipäättään tärkeää sen sidosryhmille.

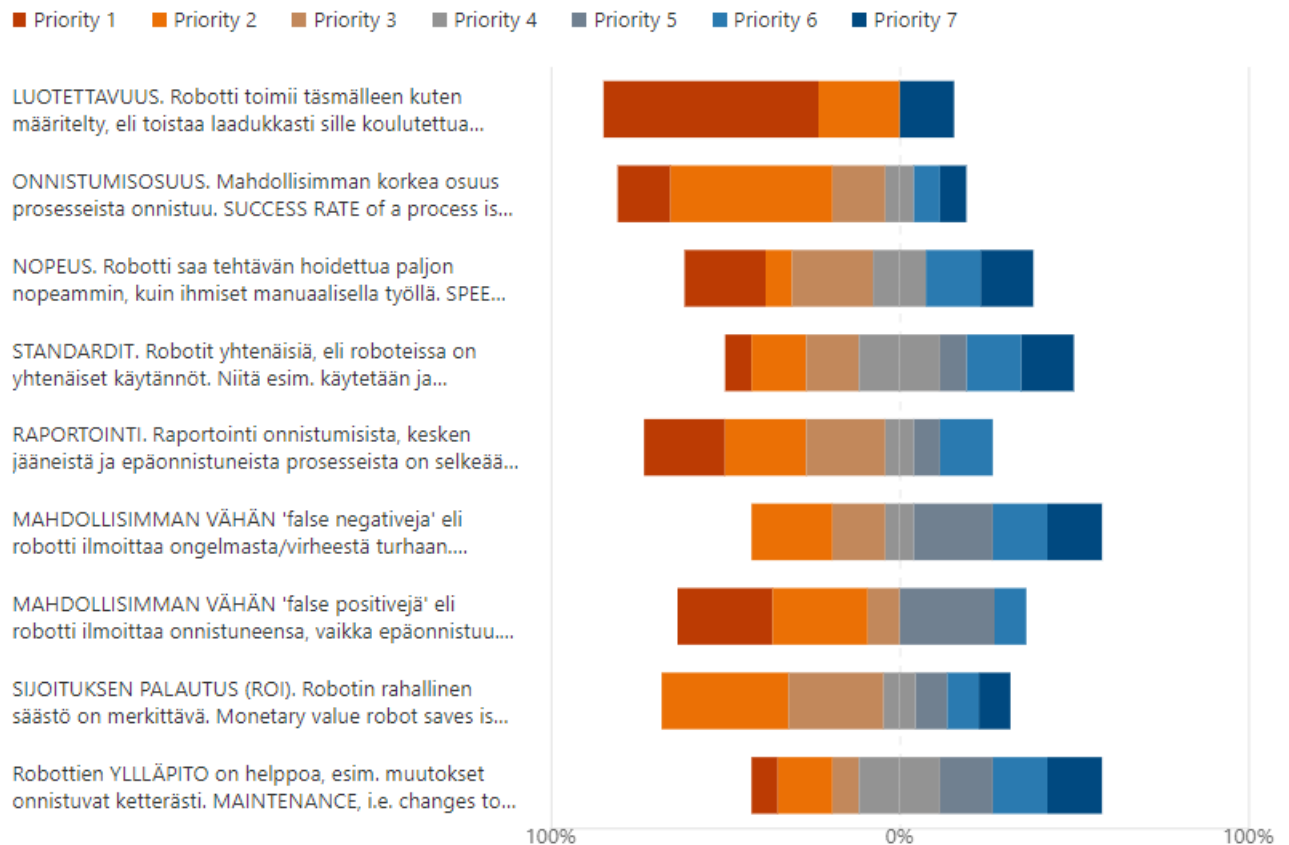
Aiemmin kerätyille robotiikan sidosryhmiin kuuluville yhteyshenkilöille lähetettiin kysely, jota tässä diplomityössä käytetään keskeisenä aineistona (Liite 2). Kyselyyn vastasi yhteensä 13 henkilöä. Muut vastaajat ovat hieman eri rooleissa, mutta eniten oli robotin käyttäjiä. Kyselyyn vastaajien roolit ovat esitetty kuvassa 4.



**Kuva 4** Vastaajien roolit

Vastausvaihtoehdot ovat listattuna vasemmalla kuvassa 4. Yleisimmät vastaajien roolit olivat joko robotinkäyttäjä tai tiimiläinen, jonka tiimissä robotti on käytössä. Roolia kysyttäessä sallittiin useampi vastaus ja siten roolien kokonaismäärä on 15, vaikka vastanneita oli vain 13. Lopulta kyselyä ei lähetettykään tiimeille, joille ei vielä ole kosketuspintaa robotiikkaan, eikä myöskään liiketoiminnanjohtajia saatu vastaamaan. Käytännössä vastaajat ovat kuitenkin juuri heitä, jotka robotiikan kanssa eniten ovat tekemisissä. Myös RPA tiimistä saatiin 4 vastausta.

Robottiikan arvot olivat yksi keskeisistä ja ensimmäisistä kysymyksistä, joka yhteyshenkilöille lähetetyssä kyselyssä oli mukana. Tällä haettiin hyvinkin selkeää kuvaa siitä, mikä ainakin kohdeyrityksessä koetaan robotiikassa tärkeäksi, mutta samojen arvojen yleismaalisuuden vuoksi ne voisivat päteä jopa eri toimialalla. Arvokyselyn tulokset ovat kuvattu kuvassa 5.



**Kuva 5** Robotiikan arvot

Kuvassa 5 vasemmalla on listattuna robotiikkaan liittyviä arvoja, mitkä ovat niitä tekijöitä, jotka johtavat hyvään robotiikkaan. Vastaajia pyydettiin valitsemaan eri tekijöille eri prioriteetit. Prioriteetti 1 tärkein ja Prioriteetti 7 vähiten tärkeä, mutta silti prioriteetti. Kaikkiin kohtiin ei tarvinnut laittaa vastausta, eikä samankaan prioriteetin käyttöä estetty. Kuitenkin tuloksista nähdään selvästi, että eri tekijöitä pidetään eri arvossa. Myös muuttujia, jota ei valmiiksi ollut listattuna kehoitettiin kirjoittamaan ylös. Niihin vastattiin vapaalla tekstillä kuitenkin ohjeistaen sisältäen vastaavalla tavoin prioriteettinumeron.

Valmiiksi valitut arvot, jotka ovat osin kytköksissä toisiinsa, valittiin seuraavasti:

- Luotettavuus. Robotti toimii täsmälleen kuten määritelty, eli toistaa laadukkaasti sille koulutettua prosessia, ja jos jokin tapahtuma ei täsmälleen seuraa tätä mallia, keskeyttää ja kertoo sattuneesta poikkeamasta.
- Onnistumisosuus. Mahdollisimman korkea osuus prosesseista onnistuu.
- Nopeus. Robotti saa tehtävän hoidettua paljon nopeammin, kuin ihmiset manuaalisella työllä.
- Standardit. Robotit yhtenäisiä, eli roboteissa on yhtenäiset käytännöt. Niitä esim. käytetään ja raportointi toimii samankaltaisesti.
- Raportointi. Raportointi onnistumisista, kesken jääneistä ja epäonnistuneista prosesseista on selkeää.
- Mahdollisimman vähän 'false negativeja' eli robotti ilmoittaa ongelmasta/virheestä turhaan.
- Mahdollisimman vähän 'false positivejä' eli robotti ilmoittaa onnistuneensa, vaikka epäonnistuu.
- Sijoituksen palautus (ROI). Robotin rahallinen säästö on merkittävä.
- Robottien ylläpito on helppoa, esim. muutokset onnistuvat ketterästi.

Kuten kuvasta nähdään, luotettavuus on selvästi arvostetuin tekijä robotin arvoista. Myös onnistumisen osuutta, raportointia, false positivien minimointia ja nopeutta pidettiin erityisen tärkeänä. Vastaajien lisäämiä tekijöitä olivat: Robottien työjonoon asettamisen ja tulosten hallinnoinnin yhtenäistäminen eri prosessien välillä, Robottikalenteri ja vastuuhenkilöt sekä päivituskalenteri eri sidosryhmien kanssa, sekä poikkeustilannefoorumi, läpinäkyvyys ja Robottien vaikutus prosessin virheettömyyteen sekä laadun variaatioon. Lisätyille ehdotuksille annettiin jokaiselle prioriteetti yhden ja kolmen välillä, eli niiden esittäjät pitivät näitä merkittävin. Kaiken kaikkiaan kattavan teeman voisi nähdä olevan robottien onnistuminen ja läpinäkyvyys.

## 6 VIESTINTÄMALLI OHJELMISTOROBOTIIKKAAN

### 6.1 Lähtötilanne

Robottiikan käyttöön ja ylläpitoon liittyy siis paljon viestintää. Tyypillisiä kommunikaatiotilanteita ovat epäselvien tai vikatilanteiden ratkaisut. Lisäksi erikokoisia muutoksia tarvitsee toteuttaa ajoittain. Näitä voi seurata, esim. yrityksen sisäisestä tai asiakkaan toiveesta, ja etenkin järjestelmien päivittämisestä.

Lähtötilanteessa kohdeyrityksellä ei ollut järjestelmällistä ja yleisesti käytettyä tapaa viestiä robotiikkaan liittyvistä asioista. Tiedossa ja pitkälti erinäiseen dokumentaation ylös kirjattuna oli kunkin robotin avainrooleissa olevia henkilöitä, kuten robotin tekijä, prosessiasiantuntija ja robot owner, eli käytöstä ja valvonnasta ensisijaisesti vastaava henkilö.

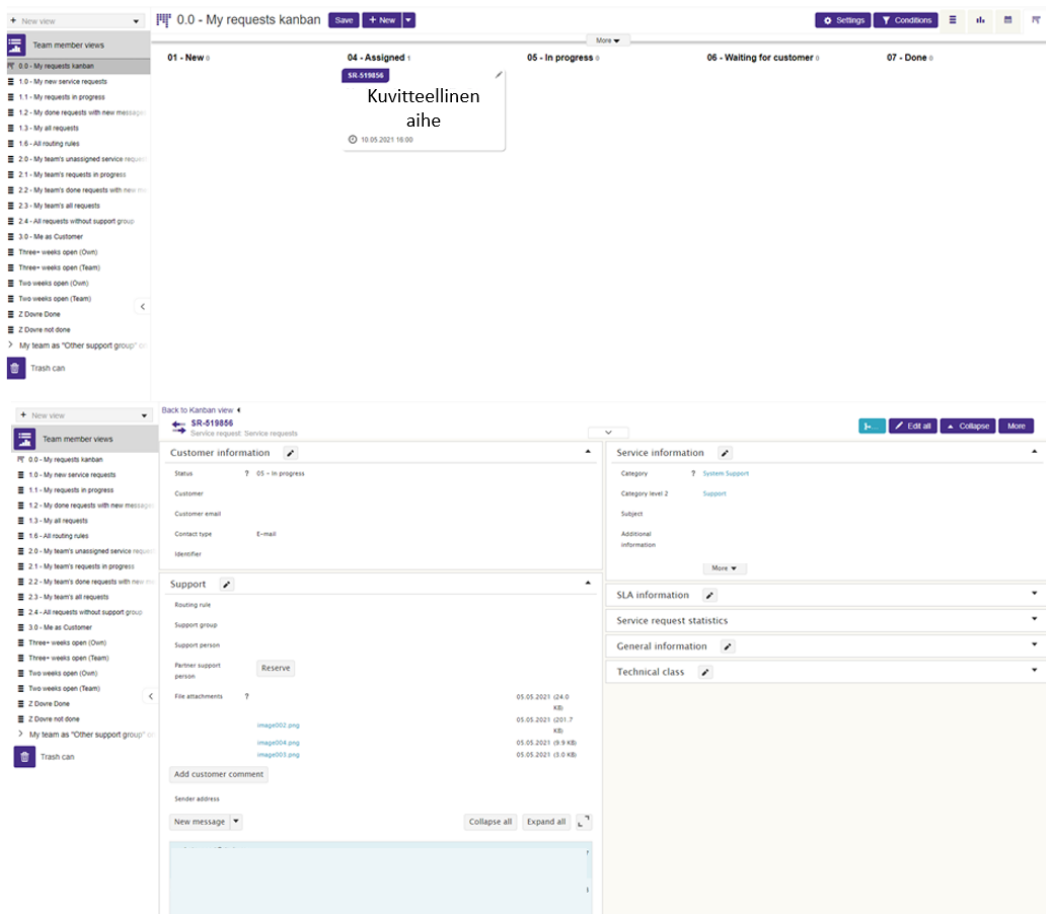
Yrityksen sisäisesti yleisesti käytettyjä viestinnänkeinoja ovat sähköposti, sekä virtuaaliyhteistyön ohjelmisto, joka mahdollistaa suoran viestinnän ja puhelut. Siten robotiikan sidosryhmät ovat myös käyttäneet näitä, usein suoraan henkilöltä toiselle. Myös sähköpostiketjut ja ryhmäpuhelut ovat yleisiä. Yhtä kaikki keskeinen ongelma on siinä, etteivät oleelliset tiedot saavuta kaikkia sidosryhmiä, mikä aiheuttaa toiston tarvetta ja väärinkäsityksiä. Useimmiten keskeiset henkilöt ovat kuitenkin liiketoiminta-alueittain, taikka tiimeittäin yhteneviä, ja siten näiden vastuualueiden sisällä saadaan pidettyä melko kattava yleiskuva sen roboteista.

### 6.2 Ensimmäinen ratkaisuehdotus

Merkittävin etu nähtiin siinä, että jonkinlainen yhtenäistetty kanava saatiin käyttöön. Ratkaisu koostui kahdesta eri kokonaisuudesta, sidosryhmien jakamasta toteutusportaaleista ja chattikanavasta, sekä tikketti-järjestelmän käyttöönotosta. Näillä pyritään siihen, että tarvittavat tehtävät ovat paremmin selvillä ja että olennaiset viestit menevät perille kaikille sidosryhmille samalla kertaa. Tikettiportaalin tarkoituksena on, että selkeät määritellyt ylläpitotoimenpiteet ja muutospyynnöt saataisiin kerättyä sinne olennaisine tietoineen. Toteutusportaaliin ja

viestikanaville on tarkoitus kerätä suunnittelua mahdollisista muutoksista tai uusista projekteista, ajankohtaisia asioita ja muuta hyödyllistä tietoa.

Samaa tikettijärjestelmää on käytetty yrityksessä muissakin liiketoimintayksiköissä ja siten toimittajien valinta ei tullut kysymykseen, vaan todettiin järkeväksi käyttää ehdottomasti samaa ohjelmaa kuin muukin organisaatio. Toisaalta järjestelmällä itsessään ei ole suurta merkitystä, kun se toimii tietyllä tavalla. Tikettijärjestelmän mahdollistaa johdonmukaisen ja yksinkertaisen tehtävien seurannan ja sen ominaisuuksiin kuuluu esimerkiksi Tehtävien tilanäkymä. Järjestelmän käyttöönotolla pyrittiin ensisijaisesti siihen, että satunnaisesti paikasta toiseen lähetettävien sähköpostiviestien määrä vähenee ja pienennetään riskiä tärkeiden sähköpostien ja tehtävien unohtumiselle tai huomiotta jättämiselle. Tikettijärjestelmässä niin työntekijän kuin esimiehen on helppo seurata, miten tietty tehtävä edistyy, ja edistymisestä voidaan myös helposti raportoida tiketin lähettäneelle taholle. Alla kuvassa 6 on kuvattuna tikettijärjestelmän tilataulukko ja kuvitteellinen tiketti.



**Kuva 6** Tikettijärjestelmä ja monia sen ominaisuuksista

Kuvassa 6 yläosassa näkyy tikettijärjestelmän tehtävien tilanäkymä ja monia sen ominaisuuksista. Tilanäkymässä tehtävien edistymistä voidaan helposti seurata. Nyt assigned tilassa, eli vastaavalle henkilölle ohjattuna on tiketti kuvitteellisella aiheella. Kuvan alaosassa on avattuna tiketti. Systemissä pystytään asettamaan tarkkoja SLA vaatimuksia ja muita yksityiskohtia, mutta nykyisellään vastaaminen ja tilan muuttaminen etenemisen mukaan ovat riittäviä toimintoja.

Itse tikettijärjestelmää ei siis tarvinnut suunnitella, mutta sen käyttöönotto oli toteutettava järkevällä tavalla. Pahimmillaan vain osa tahoista pääsisi perille uuden viestintäkanavan käyttöönotosta, joka pahentaisi sekavaa viestittelyä mahdollisesti entisestään. Vaihtoehtoina punnittiin koko yritykselle lähetettävän uutiskirjeen ja kohdennetun sähköpostilistan välillä. Suuri osa yrityksen tiimeistä ei vielä kovin aktiivisesti ollut tekemisissä robotiikan kanssa ja sitten todettiin kohdennettu julkaisu järkevämmäksi. Kerättiin jo pitkälle tiedossa oleva lista

robotiikan sidosryhmistä ja lähetettiin näille ohjeet uuden portaalin käyttöönottoon. Olennaisia yhteyshenkilöitä tälle tarkoitukselle olivat robottien käyttäjät ja tiiminvetäjät tiimeissä, joissa robotiikkaa oli käytössä.

Asiakkuusportaaliin on tarkoitus koostaa yhteen näkymään olennaiset tiedot tietyn yksikön robotiikasta. Tällainen malli oli saatavilla toisesta yhteydestä ja se oli suoraviivaista muokata sisäisen robotiikan tarpeisiin, vaikka tarkalla muodolla tuskin on merkittävästi väliä, on valittu malli kuvassa 7.

|  |                           |                   |
|--|---------------------------|-------------------|
| Kirjanpito-osasto  | RPA-toteutuksia           | 3                 |
|  | Tunnistettuja kohteita    | 1                 |
| RPA-Yhteyshenkilöt<br>Yhteys Henkilö<br>Vastaava Konsultti | TUOTANTOPROSESSIT         |                   |
|  | Prosessin nimi            | Organisaatio      |
|  | Prosessi 1                | tiimi 1           |
|  | Prosessi 2                | tiimi 1           |
|  | Prosessi 3                | tiimi 2           |
|  | TUNNISTETUT KEHITYSKOHEET |                   |
|  | Prosessin nimi            | Organisaatio      |
|  | Prosessi 4                | tiimi 2           |
|  | ROBOTTI                   |                   |
|  | robotin nimi              | Sisältö           |
| Robotti 1  | unattended                |                   |
| YMPÄRISTÖT   |                           |                   |
| Ympäristön nimi  | Tunnus                    | Lisätietoja       |
| Palvelin 1   | käyttäjä 1                | tuotantoympäristö |
| Palvelin1  | käyttäjä 2                | testiympäristö    |

**Kuva 7** Asiakasportaali (Yritys 2021a)

Asiakasportaali, kuva 7, sisältää mahdollisimman laajasti olennaisia tietoja asiakkuuden kokonaisuudesta. Mitä prosesseja on toteutettu, onko jotakin suunnitteilla, ketkä ovat ensisijaisia yhteyshenkilöitä niin asiakasyksikön kuin robotiikan kehittäjien osalta, käytetyt ympäristöt ja lisäksi sekä mahdolliset tiedot automaation vaikutuksista, jos nämä ovat tiedossa.

Tällaiset portaalit toteutettiin järkevän kokoisiin yksiköihin jakaen, joko liiketoiminta-alueittain tai tiimeittäin riippuen sisäisen asiakkuuden koosta. Liiketoimintayksikkö on hyvä

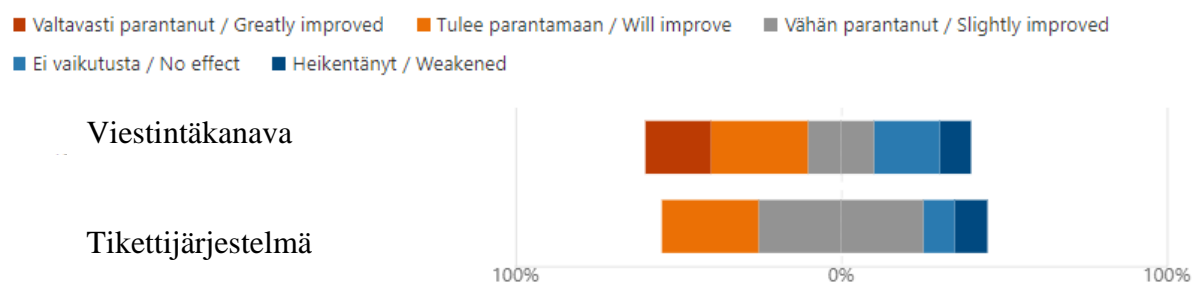


jakoperuste, kun robotiikkaa ei vielä hirveästi ole käytössä, mutta toteutusten määrän kasvaessa voidaan jako tehdä ennemmin tiimien mukaan.

Vanhoja viestintäkanavia ei suoraan lakattu käyttämästä vaan tarkoituksena on asteittain kannattaa sidosryhmät käyttämään uutta tikettijärjestelmää. Käyttöönnotossa ei onnistuttu täydellisesti, sillä suoria sähköpostiviestejä käytetään yhä laajasti, mutta jos edes valtaosa selkeässä toimenpiteistä saadaan ohjattua tikettijärjestelmään tai viestintäkanavalle, parantaa se tilanneteen seurattavuutta huomattavasti. Toki itse selvitystyön suorittamisessa suora, mahdollisimman tehokas kommunikaatio esimerkiksi puhelulla on tärkeä työkalu.

### 6.3 Palautteen kerääminen

Robotiikan sidosryhmille lähetetyllä kyselyllä haettiin samalla vastausta erityisesti kahteen aiheeseen: robotiikan arvoihin sekä kommunikaatioon robotiikasta. Erityisesti kommunikaatio puolen kehitykseen oli tarkoitettu kysymys kuusi: ”Alkuvuodesta on alettu ottamaan käyttöön keskitettyä RPA viestintäkanava sekä tikettijärjestelmää suorien sähköposti- ja elektronisten viestien sijasta. Näetkö niiden parantavan robotiikkaan liittyvää kommunikaatiota?” Vastaukset ovat kuvassa 8.



**Kuva 8** Uuden viestintämallin vaikutukset

Vastauksia on 11 ja enemmistö näki muutokset positiivisina. RPA tiimi kanavien osalta yleisin vaihtoehto oli ”Tulee parantamaan”. Tämä selittynee osin sillä, että käyttöönotto on asteittaista ja vaatii aikaa. Kuitenkin se nähdään hyödyllisenä muutoksena. Sen sijaan tikettijärjestelmän hyöty on välittömämpi ja tässä tyypillisin vastaus onkin ”Vähän parantanut”. Ylipäätään

viestintäjärjestelmän merkitys lienee suurempi ylläpidon ja RPA-tiimin näkökulmasta, sillä robotin käyttäjille ei suuresti ole väliä, mitä kautta he saavat yhteyttä tukeen, kunhan viesti menee perille. Robotiikkatiimi taas saa näitä vastaavia viestejä niin useasta lähteestä, että ilman systeemiä on vaikea pysyä ajan tasalla.

Viestinnän kehittämiseen kysyttiin myös avoimia ideoita. Yksi vastaajista piti viestintäkanavasta, juuri siksi että kaikki asianomaiset saavat tiedon kerralla, kun taas toinen suosisi edelleen suoraa kontaktia. Yksi vastaus esitti selviä jatkokehitysideoita: ”-- Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa sekä poikkeustilannefoorumi.” Lisäksi saatiin laajemmin tilanteeseen kantaa ottava vastaus ” --ei olla tehty kovinkaan selvää linjausta siitä, että robotiikka sulautetaan osaksi kokonaisvaltaisempaa tehostamista--” ja ” Varsinkin sisäisen robotiikan osalta kytkeytyminen muuhun järjestelmien ja toiminnan kehitykseen on aika lailla ilmiselvää--”. Tämä heijastelee kirjallisuutta. Jotta robotiikkaa saadaan todella skaalattua, on tärkeä saada yhteistyö mm. johdon ja IT:n kanssa toimimaan.

#### **6.4 Parannettu ehdotus**

Radikaaleja muutoksia ensin rakennettuun kommunikaatioalustaan ei palautteen perusteella tehdä. Sen käyttöönottoa jatketaan sisällyttämällä yhä enemmän osallistujia siihen. Vain käyttämällä tätä uutta järjestelmää tarpeeksi, saadaan se vakioitua. Alkuperäisen ajatuksen mukaan yleisöä ei olla kuitenkaan kasvattamassa radikaalisti, ei esimerkiksi koeta järkeväksi lisätä kaikkea robotiikkaa käyttävän yksikön jäseniä sen kanavalle, jos käytännössä informaatiosta hyötyy lähinnä yksikön johtaja, robot owner ja tietyssä tilanteessa asianomainen tiimin jäsen.

Joitakin lisäominaisuuksia päätetään kuitenkin lisätä. Yleissivulle, jonne siis kaikista yksiköistä päästään käsiksi, lisättiin wiki. Tähän on tarkoitus lisätä relevantteja sanastoa ja sen määritelmiä niin robotiikan kuin mahdollisesti alan yleisestä substanssiosaamista. Toisena ominaisuutena lisätään kalenteri, johon voidaan merkitä säännöllisten prosessien vakioidut suoritusajat, poikkeamat, prosessin suorittamisen tarpeet, sekä muita olennaisia päivämääriä, kuten päivitysten ajankohtia tai yhteisten tapaamisten aikoja.

Kaiken kaikkiaan tämä ratkaisu keskittyy hyvin käytännön läheisiin asioihin ja päivittäisen robotiikkaan. Kuitenkin palautteesta nähdään, että paljon laajempiakin teemoja on nostettu esille. Tällaisen suurempaa mallia käsitellään luvussa kuusi, mutta tämänlaisen muutosprojektin suunnittelu ja toteutus jää tämän työn rajauksen ulkopuolelle.

## 7 TEKNINEN MALLI

### 7.1 Lähtötilanne

Robottiikan tilannetta ja tarvetta käytäntöjen yhtenäistämiseksi on pohdittu jo pitkään ja ongelmia ja parannusehdotuksia myös dokumentoitu. Ensisijaisena lähteenä tässä on käytetty yrityksen konsulttien koostamaan listaa kehitys- ja parannuskohteista yrityksen sisäisessä robotiikassa. Aiheet sivuavat laajempia teemoja, mutta ovat keskittyneet tekniseen toteutukseen, ennen muuta näiden yhtenäistämiseen. Tässä listattuja aiheita ovat esimerkiksi Orchestratorin käytännöt, serverikäytännöt, lupakäytännöt, frameworkin käytännöt, projektien läpivientimalli ja ylläpito, sekä oman toiminnan kehittäminen (kohdeyritys 2020a). Tässä työssä on keskitytty erityisesti käytäntöihin Servereiden ja frameworkin osalta, mutta teemat sitoutuvat yhteen monelta osin.

Orchestratorin osalta pohdittiin miten logiviestejä tulisi vakioida, mitkä prosessit tulisi siirtää jonoihin, miten Orchestratorista saataisiin kerättyä dataa ja miten testi ja tuotantoympäristöön erityisiin toisistaan Orchestratorissa. Logeihin ehdotettiin standardointia, joka onkin laajasti otettu käyttöön. Jonoihin liittyviä kysymyksiä on, mitkä prosessit siirretään jonoihin ja mitä ei kannattaisikaan siirtää jonoon millä perusteella siirto tehtäisiin ja onko se hyödyllistä. Mielipiteet tästä vaihtelivat suuresti. Tuotanto ja testiympäristö ja erottamisessa harkittiin eri keinoja oikeuksien hallintaan ja eri kansioita. Lisäksi harkittiin pääsyn antamista Orchestratoriin laajemmin yrityksessä. (kohdeyritys 2020a)

Useilla servereillä prosesseja pyöri eri tiimeille. Koettiin että myös näiden järjestäminen olisi tärkeää. Miten servereiden käyttäjätunnukset ja robotit jakautuvat eri yksiköiden tiimien kesken ja miten servereiden käyttöastetta seurataan. Missä vaiheessa jollekin tiimille pyydetään kokonaan uusi serveri tai käyttäjätunnus, jos se nähdään tarpeelliseksi. (kohdeyritys 2020a)

Vielä projektin alussa REF:ejä oli erilaisia, usein kuitenkin niin, että saman tekijän käytännöt pysyvät samana. Näiden eroista seuraa erot transaktiodatan muodostuksessa, raportoinnissa ja siten ohjeistuksessa loppukäyttäjälle raporttien tulkitsemisessa. Selvitystyötä tehdään tapaus kerrallaan. Lisäksi roboteissa tarvitsisi olla kommentointia. Vaihtoehtona punnittiin myös, että

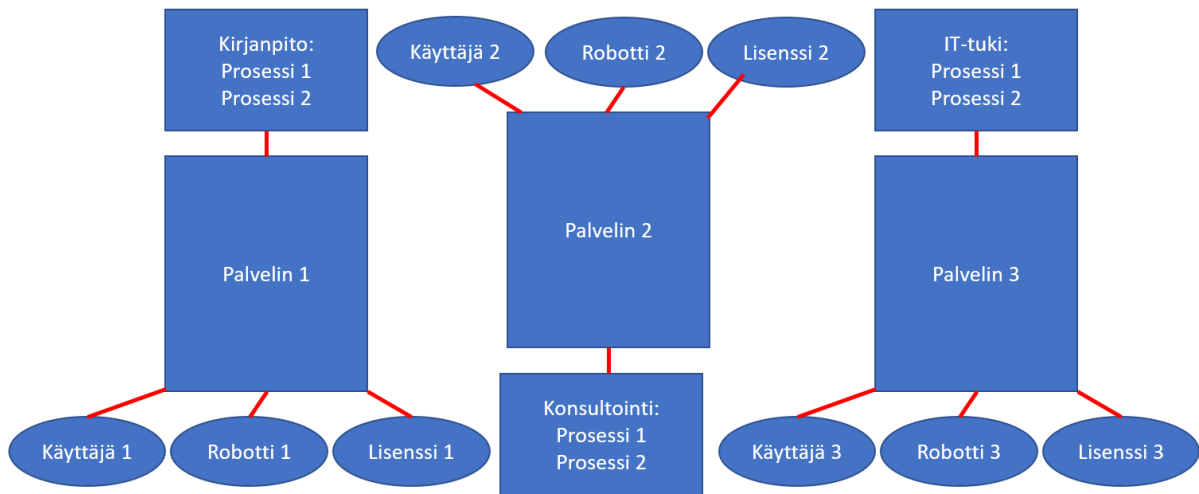
hyväksytään tilanne erilaisilla REF:eillä. Ennakoitua vakioidun REF:in käytössä olivat se, että kuka tahansa voisi muokata toisen prosesseja ja prosessien välille saataisiin vakio-osia. Mitä pidemmälle mallia vakioitaisiin, sitä parempi. REF:iin liittyy myös raporttien standardointi, eli miten kaikista prosessien lähettämistä raporteista saataisiin samanlaisia toimintaperiaatteeltaan. (kohdeyritys 2020a)

Vaikka dokumentissa on esitetty useita ratkaisuehdotuksia näihin erilaisiin ongelmiin ei suuria edistysaskeleita ollut asioiden tiimoilta tapahtunut diplomityöprojektin alkaessa. Syitä tähän lienee lukuisia, joista suurimpana priorisointi ja resurssit. Kun on kädet täynnä arjen pyörittämisessä, ei voida pysähtyä miettimään, miten tilannetta saisi parannettua. Kaikkiin dokumentissa esitettyihin haasteisiin tämä diplomityön ei ole tarkoituskaan vastata, mutta jos edes muutama keskeinen näistä saataisiin ratkottua, ja ennen todellista skaalausvaihetta luotua kestävä standardit toteutuksille, oli siis se suuri harppaus eteenpäin.

## **7.2 Ensimmäinen ratkaisuehdotus**

Ratkaisun lähtökohtana oli, että radikaaleja muutoksia pyritään välttämään jokaisen muutoksen saattaessa aiheuttaa ennakoimattomia etävaikutuksia. Siten keskityttiin jo olemassa olevien taipumusten konkretisointiin. Kuten ylläpidon teoriassa todettiin, on toteutusten alkuperäisellä tekemisellä ja myöhemmillä muutoksilla suuri ero. Mallin tarkoituksena on olla toimiva myös alusta asti rakennettavana, eikä vain jälkikäteen implementoituna. Tiedostorakenteiden muuttaminen on verrattain pieni muutostoimenpide tuotannossa oleviin robotteihin, sillä tarvitsee muuttaa vain tietoa siitä, mistä sijainnista tietynlainen tiedosto luetaan. Itse ydinprosessi, jota tietyssä käyttöliittymässä toteutetaan, ei silloin muutu.

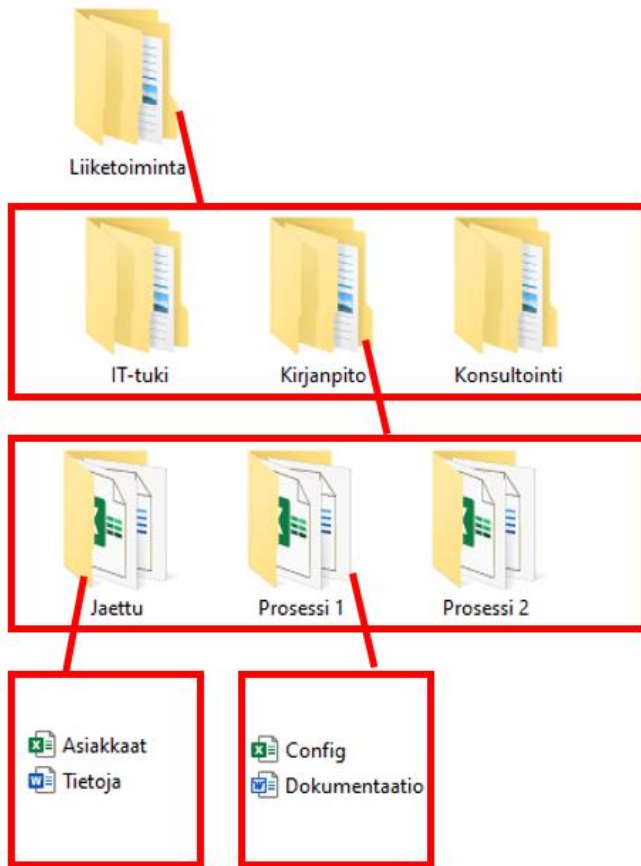
Vaikka näin jo pääosin olikin, päätettiin tiettyjen liiketoiminta-alueiden taikka tiimien käyttämien robottien ympäristöt järkevöittää. Siten tietyllä yksiköllä olisi omistettu tuotantoympäristö eli palvelin, käyttäjä ja allokoitu robotti. Servereiden sovittaminen on kuvattu kuvassa 9.



**Kuva 9** Servereiden rakenne

Kuvassa 9 on valaistu kuvitteellisen yrityksen robottipalvelimien, käyttäjien, robottien ja lisenssien malli. Yksiköiksi on valikoitunut liiketoiminta-alueet eli kirjanpito, konsultointi ja IT tuki. Vastaavaa jakoa käytetään siis tiedossa rakenteissa ja viestintä kanavissa. Tietyn kokoisilla yksiköllä kuten liiketoiminta-alueella tai suuremmilla tiimeillä on tietty palvelin, käyttäjä, robotti ja lisenssi. Palvelimien tulee olla mahdollisimman identtisiä keskenään, sillä pienetkin erot niissä saattavat johtaa robottien epäonnistumiseen. Kullekin yksikölle on myös vastaava testiympäristö, jossa robottia voidaan testata ja ajaa ylläpidon tarkoituksissa. Testiympäristön ei tarvitse olla oma serverinä, mutta niiden tulee olla mahdollisimman identtisiä tuotantoympäristön kanssa.

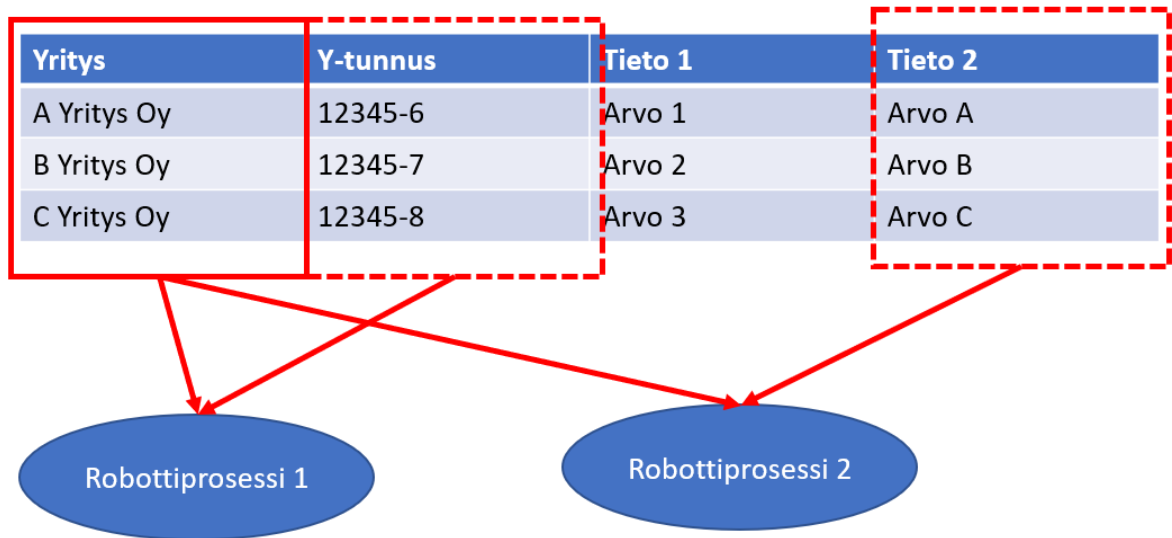
Etenkin tietyn liiketoiminta-alueen sisällä robottien vaatimat tiedot, parhaana esimerkkinä yritys- tai asiakkuuslista ovat samoja lukuisiin prosesseihin. Yksi kerrallaan rakennettuna kussakin robotissa oli oma tiedostosijaintinsa, jossa nämä halutut tiedot olivat siis kerättyä. Tämä johtaa siihen, että kun tietylle tiimille olennainen listaus muuttuu, pitää sama tieto muuttua useassa paikassa. Vaihtamalla nämä tiedostojen osoitteet jaettuun ajankohtaiseen listaan voidaan muutokset tehdä keskitetysti ja tehokkaasti. Keskitetyn tiedostorakenteen idea on esitetty kuvassa 10.



**Kuva 10** Hierarkkinen tietorakenne

Kuvassa 10 kuvitteellisen yrityksen toiminta jakautuu IT-tuen, kirjanpidon ja konsultoinnin liiketoiminta-alueisiin. Näiden alle on koottuna yksikön jaetut tiedot esimerkiksi lista asiakkaista ja muita mahdollisia tietoja. Kunkin robottiprosessin konfigurointitiedosto ja dokumentaatio löytyvät vastaavasti nimetyn kansion alta.

Monet prosessit perustuvat siihen, että asioita tehdään per yritys. kuitenkin tarkalleen vaaditut tiedot vaihtelevat esimerkiksi tietyssä prosessissa saatetaan tarvita vain yrityksen nimi, kun taas toisessa tarvitaan y tunnusta. Eri robotit on rakennettu käyttämään erilaisia tiedostopohjia, kuten Exceleitä, ja jotta nämä hieman eri tavalla rakennetut listat saadaan yhtenäistettyä, tulee tiedonlukuvaihetta muokata roboteissa esimerkiksi filtteröimällä halutut sarakkeet. Kun itse prosessivaihe kuitenkin alkaa oikean formaatin tiedoilla, ei näiden hakupaikalla ole väliä. Periaate on havainnollistettu kuvassa 11.



**Kuva 11** Jaettujen tietojen suodattaminen eri prosesseille

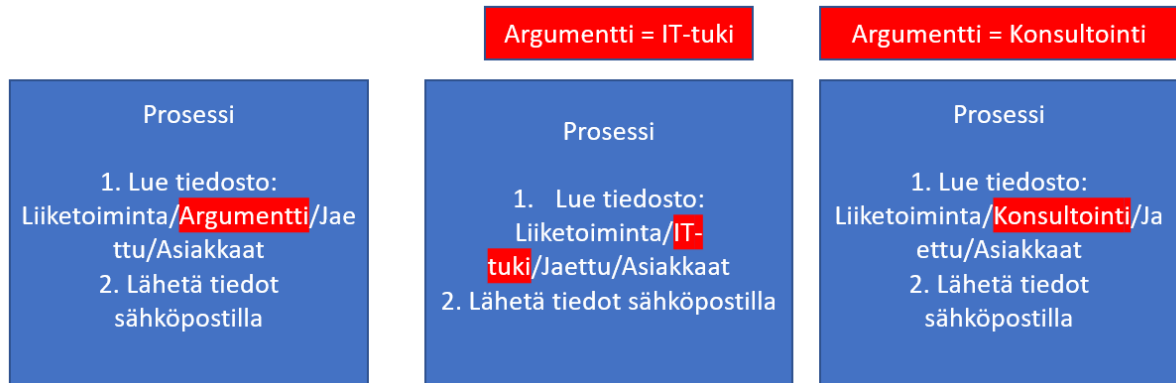
Kuvassa 11 on kolmen kuvitteellisen yrityksen lista, joille kullekin löytyy tiedot y-tunnuksesta sekä tieto 1 ja 2. Robottiprosessi yksi tarvitsee tietoja yrityksestä ja sen y-tunnuksesta, kun taas robottiprosessi 2 tarvitsee yritystä ja tietoa 2. Käytännössä robotti ohjataan lukemaan koko taulukko, josta sitten poistetaan turhat sarakkeet.

Kuten kuvassa 10 nähtiin, on kunkin prosessin konfigurointitiedosto tallennettuna sen kansion sisälle hierarkiassa. Konfiguroinnin avulla saadaan roboteille määriteltyä ideaalisesti kaikki vakioarvot, eli sellaiset tiedot, joita ei kerätä varsinaista prosessia suorittaessa, vaan ovat pysyviä. Tällöisiä ovat mm. tiedostosijainnit, vakiot kuten toistojen määrät, viiveet tai laskureiden maksimiarvot esimerkiksi kierroslaskureita varten. Perinteisessä rakenteessa konfigurointitiedosto on robotin kansiorakenteessa. Tällöinkin sen arvojen muuttaminen on paljon vaivattomampaa, kuin varsinaisen ”koodin”. Säilyttämällä tiedostoa kuitenkin etäsijainnissa, nämä tiedot saadaan tarpeen mukaan muutettua ilman uuden robotin version julkaisemista. Konfiguraatitiedostoja tulee myös ylipäätään käyttää mahdollisimman laajasti robotin ohjailemiseen.

Lisäksi malliin kuuluu laaja argumenttien käyttö skaalaamisen mahdollistajana. Kun lähtöarvoa muutetaan, saadaan esimerkiksi vaihdettua yritystä, jolle robotti tekee tiettyä prosessia. Jos tehtävä ja järjestelmät ovat esimerkiksi muuten samat, mutta tunnukset ja käytetyt tiedosto



eroavat, tai vaikka prosessissakin olisi pieniä eroja, saadaan nämä kaikki toteutettua argumenteilla. Argumenttien käyttöä on havainnollistettu kuvassa 12.



**Kuva 12** Argumenttien käyttö

Kuvassa 12 on robottiprosessi, joka lukee tiedoston tietyistä sijainnista, vertaa kuva 9, ja lähettää sen tiedot sähköpostilla. Tiedoston osoitteeseen on sisällytetty argumentti ja suorittamalla sama prosessiin argumentin eri arvoilla, ”IT-tuki” tai ”Konsultointi” eri tiedostot luetaan. siten samaa robottia voidaan käyttää samaan prosessiin, mutta eri tiedoilla. Jos myöhemmin liiketoiminnan alueet kasvavat tai muuttuvat, voidaan identtinen kansio rakenne lisätä uudelle yksikölle.

Ensimmäinen ratkaisuehdotus tehtiin siis melko välittömään tarpeeseen ja ennen laajempaa teorian tutustumista. Se pyrkii ratkaisemaan joitakin hyvin käytännönläheisiä ongelmia puuttuvista käytännöistä. Ennen muuta se pyrkii lisäämään joustavuutta samalla luoden järjestystä. Se myös tavallaan edistää uudelleenkäyttöä skaalaamista, siten ettei erillisten robottien kokonaismäärät ole yhtä korkeita, kuin jos jokainen toteutuisi olisi erillinen. Lisäksi skaalaaminen onnistuu ideaalisesti tiedostorakenteita ja konfiguraatitiedostoa muokkaamalla. Ehdotus ei kuitenkaan vielä määritä varsinaista ylläpidon operointimallia, ja ongelmien korjaaminen jatkuukin sillä vielä melko ”välitön korjaus”-tyyliin. Seuraavaksi tuleekin kerätä palautetta ja suunnitella suuremmalla tasolla ylläpidän operointia.

### 7.3 Palautteen kerääminen

kyselyn vastaukset ja varsinkin vapaa palaute ottavat osin kantaa myös enemmän teknisiin seikkoihin, mutta erityisesti tätä ajatellen toteutettiin haastatteluja RPA tiimin sisällä (liite1). Haastattelussa kysyttiin aluksi haastateltavan taustaa ohjelmistorobotiikkaan. Sitten kysyttiin yleisellä tasolla minkä nähdään edistävän robotiikan skaalaamista ja ylläpitoa, ja toisaalta mitkä asiat asettavat tälle esteitä. lisäksi kysyttiin teknisistä yksityiskohdista, jotka liittyvät esitettyyn mallin. lopuksi kysyttiin, oliko haastateltavalla muuta lisättävää aiheeseen. Vastauksista nähdään, että kukin vastaajista on robotiikan teknisiä osaajia ja heillä on useamman vuoden kokemus toteutuksien tekemisestä ja kokonaisuuden ohjaamisesta.

Teknisistä kysymyksistä, jotka ovat olennaisin palautteen lähde tämän mallin teknisten aspektin kehittämiseksi, ensimmäinen oli, että olisiko yhtenäinen ja hierarkkinen kansiorakenne ylläpitoa ja skaalasta helpottava tekijä kaikki vastaajat olivat näiden kannalla. Perusteluissa oli vähän eroa: *”Olisi pitänyt olla alusta asti-- että kaikki olisi tehty jonkin tietynlaisen standardin mukaan—”, ”Kun sitten siinä näkee sen kokonaiskuvan niistä roboteista ja mitä ne syövät, mitä ne tekevät.”* Muistutettiin kuitenkin siitä, ettei tällainen rakenne voi olla täysin kiveen hakattu sillä muutostarvetta voi ilmentua liiketoiminnan vaatimuksien mukaan: *” --täytyy huomioida se, että se tarve tulee sieltä prosessista ja loppuasiakkaalta eli se, että pystytäänkö jollain tietyllä mallilla toimimaan, riippuu siitä” ja ” Tämä on muuttuva kuvio kuitenkin koko ajan”.*

Toiseksi kysyttiin, että asemalle laitettavista konfiguraatitiedostoista. Kaksi vastaajista piti tätä hyvänä ideana, mutta kolmas ei, sillä oletuksella, että pääsy tiedoston olisi myös muilla kuin robotiikan kehittäjillä: *” Onko se niin että vaan kehittäjä pääsee niihin käsiksi, vai niin että myös se substanssiosaaja tai robot owner pääsee niihin käsiksi. Se lisää vika-alttiutta sille prosessille. Sanoisin että ei, jos on niin, että ei pelkästään kehittäjä pääsee sinne.”*

Hierarkkinen kansio rakenne ja että konfiguraatio tiedostot muodostavat yhdessä selkeään mutta joustavan kokonaisuuden, sillä vaikka organisaatorakenteet ja robotin käyttäjiltä tulevat muutostarpeet muuttaisivat kansioden rakennetta rajustikin, saadaan muutokset robottiin muuttamalla konfiguraatio tiedoston arvoja. *” helpottaa ylläpitoa ja poistaa ylläpidon tarvetta, kun ei tarvitse tehdä varsinaista robotin muutosta.”* Jos konfiguraatitiedostojen sijainti

muuttuu, on tämänkin vaihtaminen robotille verrattain helppo muutos, vaikka vaatiikin koodi tason työskentelyä.

Viimeinen konkreettinen ehdotus koski jonojen käyttöä. nämä ovat Orchestratorin mahdollistava keino hallita robottien työnjakoa (UiPath 2021d). Näiden laajempaa käyttöä oli projektin aikana pohdittu, joten niiden oletettuja vaikutuksia ylläpidon näkökulmasta tuli selvittää. Yhden haastateltavan mielestä, vaikka muuten hyvänä ominaisuutena jonot nähtiinkin, ei se ylläpito on merkittävästi vaikuttanut: ”*En suoralta kädeltä itse asiassa näe sitä. riippuu varmasti tapauskohtaisesti mutta pääsääntöisesti merkittävää eroa ylläpitoon en näe. Ehkä virhetilanteessa saadaan kerättyä tietoa helpommin... Sanon tähän, että ei.*” Voikin olla, kuten toinen haastateltu sanoi, että jonojen hyöty tulee siitä, että robottien seuranta ja läpinäkyvyyttä saadaan parannettua robottien loppukäyttäjille: ”*Ne jonot, kun sinne saadaan prosessivastaaville ja muille tarpeellisille henkilölle rajattu näkyvyys, niin voivat itse siellä tarkastella, eikä mitään kauheita erillisiä raportteja tarvittaisi.*” Samoilla linjoilla oli kolmaskin vastaaja: ”*Jos nyt muutaman mainitsee, niin parantaa läpinäkyvyyttä prosessi omistajille, pääsee katsomaan niitä transaktioita, kun se robotti tekee reaaliajassa, pääsee suoraan vikatilanteessa lähettämään kehittäjille ilmoitusta.*”

Myös robotiikka-asiantuntijoiden vastauksissa yleisesti korostuvat suuresti enemmän kommunikaation ja yhteistyöhön viittaavat teemat, mutta teknisiin toteutuksiin liittyviä mielipiteitä saatiin kerättyä laveammista kysymyksistä. Teknisiä tekijöitä, jotka edistävät robotiikan skaalaamista ja ylläpitoa olivat asiantuntijoiden mielestä toimiva IT arkkitehtuuri, mahdollisimman hyvin määritellyt, vakioidut ja dokumentoidut prosessit, palveluiden modulaarisuus, joustavat robotit ja koostettu tieto helposti saatavilla erilaisista järjestelmistä. ”*Tarvitsee olla myös hyvä tekninen infrastruktuuri. Juuri tämä projektinhallinta ja versionhallinta, ja mitä uusia projekteja toteutetaan ja miten niitä valvotaan-- infra tarvitaan siihen, että näitä robotteja voidaan ylipäänsä tehdä.*” Palveluiden modulaarisuus vaikuttaa, etenkin skaalaamiseen. ”*--asioita voi tehdä useammalla tavalla mutta ei kuitenkaan ole ääretöntä määrää vaihtoehtoja. Tietyn määrän vaihtoehdot automatisoitaisiin kaikki.*” Teknisiä asioita, jotka vaikeuttaisivat robottien ylläpitämistä, eivät asiantuntijat hirveästi esittäneet. Sen voi nähdä kertovan siitä, että teknisiin asioihin on yleensä löydettävissä toteutettavia ratkaisuja. Hyvä nosto siitä mikä kuitenkin aiheuttaa ylläpito kuormaa olivat

nykyään yleistyneet SaaS palvelut: ” *SaaS ohjelmistot mitä käytetään, niin niillä on ihan oma päivitys ryhmissä ja sieltä tuodaan niitä toiminnallisuuksia ja ominaisuuksia ilman että loppuasiakas, kuten robotin käyttäjä voi juurikaan niihin vaikuttaa. Se voi tuoda aika paljon päivitystarvetta kuin aikaisemmin. On-premise tuotteissa pystyttiin muutoksia hallitsemaan ja aikatauluttamaan halutulla tavalla, mutta nykyisin yleiset SaaS-palvelut voivat sinne vaan tuodaan uusia ominaisuuksia sen enempiä kyselemättä.*”

Asiantuntijoiden mielipiteistä saa kuvan, että vaikka aluksi esitelty toimintamallin esittelemät ratkaisut ovat jo sinänsä edistysaskel, ne eivät yksin riitä. Jo ylläpidon näkökulmasta tarvitaan muutostenhallintaan lisää yhtenäisiä käytäntöjä, jolla ylläpitoa saadaan määrätietoisesti toimimaan erilaisella mallilla. Eli auttaa, että toteutukset ovat järjestelmällisiä ja joustavia, mutta vaikka etäkonfiguraatio tekeekin joidenkin muutosten toteuttamisesta helpompaa, se ei ratkaise ydinongelmaa muutoksen ajoittaisesta ja usein vaikeasti ennakoitavasta tarpeesta. Kuten yksi haastateltavista sanoo: ” *Se on vaan ohjelmistorobottien luonteessa, että ne hajoavat ja niitä on aina päivitettävä ja ne tarpeet muuttuvat kuukausien ja vuosien myötä.*”

#### **7.4 Parannettu ehdotus**

Ensivaiheessa esitetty ratkaisuehdotus todettiin edelleen toimivaksi juuri sopivana yhdistelmänä järjestelmällisyyttä ja joustavuutta. Etäkonfiguraatitiedostojen on parempi kuitenkin pysyä vain harjaantuneiden robotiikan osaajien muokattavissa. Teknistä yhtenäistämistä ja ylläpitoa ja skaalausta varten tulee kuitenkin lähteä lisäämään. Konkreettisella teknisillä ratkaisuilla voidaan myös tehdä itse ylläpitotyöstä helpompaa, kun muuttamisesta tehdään yhä joustavampaa.

Pohjautuen kirjallisuudessa esiintyneeseen kirjastomukaiseen lähestymistapaan ylläpidosta ja asiantuntijan esittämään ideaan robottitoteutusten modulaarisuudesta pyritään komponenttikirjasto saamaan käytännöksi robotti toteutuksessa. Tällaisia yksiköitä, joita sitten voidaan jatkuvasti testata ja kehittää, on varmasti paljon helpompi ylläpitää kuin täysin irrallisia toteutuksia.

REF:iin luodaan vakiomalli ja sitä aletaan käyttämään uusissa toteutuksissa, jo aiemmin kuvatuista syistä (kohdeyritys 2020a). Vanhoja toteutuksia ei kuitenkaan lähdetä päivittämään siihen, ellei saman toteutuksen muuttamiselle ole muita painavia syitä.

Kun robottien määrä kasvaa, kasvaa myös työn määrä toimintamallien tehostamisesta huolimatta. Siten robottien seurannan siirtämistä niiden käyttäjien vastuulle lisäämällä voidaan päästä yli rajallisesta RPA-osaajien määrästä organisaatiossa. Tämän voi mahdollistaa esimerkiksi jonojen käyttäminen. Tämä tarkoittaa, että robottien käyttäjiä tulee kouluttaa käyttämään Orchestratoria, mikä vaatii omat resurssinsa, mutta hyöty on vaivan arvoinen, yksi haastatellun sanoin: ”--Mitä parempi käsitys loppukäyttäjällä on robotin toiminnasta ja mitä paremmin pystytään kommunikoimaan sitä muutostarpeesta, niin se helpottaa sitä ylläpitotyötä -- jos ne robotin käyttäjät tuntee prosessiin ja robotin toiminnan, he pystyvät antamaan jo esimerkki käyttötilanteen ja mahdollisesti miettimään testitavankin tälle ongelmalle. Eli loppukäyttäjän osaaminen on sellainen asia, mikä helpottaa ylläpitoa”.

Kun robotien käyttäjät ymmärtävät käytössään olevien robottien toimintaperiaatteita, niiden tapaa pelkistää prosessi täsmällisin säännöin ja tarkkaan määritellyssä ympäristössä toimivaksi algoritmiksi, ja heille annetaan keinot seurata robottiensa toimintaa reaaliajassa tai jälkikäteen, ymmärtävät he robotin poikkeustilanteita eli eroja prosessissa jopa paremmin kuin RPA-kehittäjä. Tällöin robotin käyttäjät voivat myös ennakoida muutoksien tarvetta ja arvioida mahdollisia ongelmia tai parannuskohteita ja viestiä niistä robotiikan kehittäjille.

## 8 QFD ANALYYSI ROBOTIIKAN ARVOISTA JA TOTEUTUKSESTA

QFD-analyysi, engl. quality function deployment, on tärkeää tuotekehityksen menetelmä, jossa käännetään asiakkaan vaatimukset tuotteen tai palvelun ominaisuuksiksi (Carnevali & Miguel 2008). Tässä työssä menetelmää käytetään vertaamaan suunnitellun mallin ominaisuuksia robotiikan sidosryhmien esittämiin arvoihin robotiikasta. Taulukossa 1 on esitetty QFD analyysin tulokset.

**Taulukko 1** QFD-analyysi

|       |                                       | Toteutuksen osa-alueet |                 |                     |               |                    |                |              |
|-------|---------------------------------------|------------------------|-----------------|---------------------|---------------|--------------------|----------------|--------------|
| Arvot |                                       | Tikettijärjestelmä     | Viestintäkanava | Servereiden rakenne | Kansiorakenne | Argumenttienkäyttö |                |              |
|       | Tärkeys                               |                        |                 |                     |               |                    | Arvon sopivuus |              |
|       | LUOTETTAVUUS                          | 9,79                   | 1               | 3                   | 9             | 3                  | 3              | 185,93       |
|       | ONNISTUMISOSUUS                       | 6,23                   | 1               | 3                   | 3             | 1                  | 3              | 68,49        |
|       | NOPEUS                                | 5,62                   | 1               | 3                   | 3             | 0                  | 3              | 56,19        |
|       | STANDARDIT                            | 4,24                   | 1               | 3                   | 9             | 9                  | 9              | 131,31       |
|       | RAPORTOINTI                           | 6,28                   | 1               | 3                   | 3             | 9                  | 9              | 157,08       |
|       | MAHDOLLISIMMAN VÄHÄN FALSE NEGATIVEJA | 3,64                   | 3               | 1                   | 1             | 1                  | 1              | 25,45        |
|       | MAHDOLLISIMMAN VÄHÄN FALSE POSITIVEJA | 5,60                   | 1               | 3                   | 1             | 1                  | 1              | 39,20        |
|       | SIJOITUKSEN PALAUTUS (ROI)            | 3,76                   | 1               | 3                   | 1             | 9                  | 9              | 86,47        |
|       | YLLÄPITO                              | 4,10                   | 9               | 3                   | 3             | 9                  | 9              | 135,38       |
|       | Tarjoaman tärkeys                     |                        | 89,34           | 140,47              | 205,88        | 210,25             | 239,56         | 885,50       |
|       | <b>Tarjoaman suhteellinen tärkeys</b> |                        | <b>10 %</b>     | <b>16 %</b>         | <b>23 %</b>   | <b>24 %</b>        | <b>27 %</b>    | <b>100 %</b> |

Taulukossa 1 on vasemmalla listattuna robotiikan arvot kyselyn mukaisesti. Ylhäällä taas on listattuna mallin osa alueita. Tärkeys sarake kertoo, miten tärkeänä tiettyä arvoa pidettiin vastanneiden joukossa. Se on määritetty summalla kunkin vastaajan antaman prioriteetin käännteislukujen summalla, jolloin yksi ”prioriteetti 1” vastaus antaa yhden pisteen, ”prioriteetti 2” puoli pistettä ja niin edelleen. tyhjäksi jääneet arvot ovat laskettu nolllalla. Kukin mallin elementti on saanut arvon 0, 1, 3 tai 9 riippuen siitä kuinka paljon sen voi nähdä edes auttavan arvon toteutumista. Tällöin nolla tarkoittaa, ettei ole lainkaan vaikutusta ja yhdeksän, että se

vaikuttaa suoraan. Vertailukelpoisuuden vuoksi yksittäisten vastaajien ehdottamat arvot on jätetty tässä pois.

Tikettijärjestelmän tarkoituksena on siis vastaanottaa selkeitä toimenpiteitä, kuten virheiden selvittämistä ja ennalta määriteltyjä muutospyyntöjä. Tapaus kerrallaan ylläpito toimenpiteet parantavat toteutuksen luotettavuutta onnistumisen osuutta ja muita tekijöitä. Koska loppukäyttäjälle on usein selvää, että jotakin olisi voitu tehdä, mutta robotti on keskeyttänyt toimenpiteen virheilmoituksella, jäävät false negativet helpommin kiinni. Suoraviivainen ja merkittävä vaikutus tikettijärjestelmällä on kuitenkin ylläpitoon. Viestintäkanavalla viestinnällä sen sijaan on epäsuoraa vaikutusta kaikkiin arvojen osa alueisiin. Aktiivisella viestinnällä luotettavuutta, onnistumisosuutta jne. saadaan parannettua. Verrattuna ylläpitotoimenpiteisiin on näkökulma proaktiivisempi virheiden korjaamisen sijaan. Etsimällä uusia helppoja laajennuksia ja parannuksia voidaan vaikuttaa esimerkiksi sijoituksen palautukseen. False negativet jäävät kuitenkin pienempään rooliin kuin tikettien kanssa.

Servereiden rakenne vaikuttaa ensisijaisesti luotettavuuteen ja standardointiin taikka yhtenäistämiseen. Toteutusten tiedetään toimivan oikein tietyssä ympäristössä, ja kun ympäristöt ovat identtisiä ja selkeästi järjestettyjä, luotettavuus paranee. Servereiden, lisenssien, ja robottien järjestelmällisyys lisäävät myös suoraan yhtenäistämistä. Sopiva ympäristö myös vaikuttaa onnistumisia osuuteen, nopeuteen ja ylläpitoon. Koska false negativet ja false positivet puolestaan seuraavat useammin prosessista, ei ympäristöllä ole vaikutusta yhtä paljon tähän.

Myös kansiorakenteet vaikuttavat suoraan ja suuresti toteutusten yhtenäistämiseen. Kun tiedostoja jaetaan myös toteutusten läpi ja tietorakenne on selkeä parantaa se myös huomattavasti raportointia. Sijoituksenpalautusta parannetaan helpolla skaalaamisella ja selkeyden vuoksi myös ylläpito helpottuu. Tähän osa alueeseen kuuluvat myös, että konfiguraatio tiedostot ja nyt parantavat skaalautuvuutta ja joustavuutta omalta osaltaan. Onnistumisosuuksiin kansiorakenteet eivät kovin paljoa vaikuta. Nopeuden kannalta etärakenteet saattavat olla myös pieni haitta verrattuna suoraan suoritusympäristössä luettaviin tietoihin.

Argumenttien käyttö myöskin yhtenäistää toteutuksia ja parantaa raportointia, sijoitusten palautusta sekä helpottaa skaalaamista ja ylläpitoa. Se onkin vahvasti yhteydessä kansio rakenteiden kanssa. Ylläpito helpottuu erityisesti sen takia, että lukuisten hyvin samankaltaisten toteutusten sijaan tarvitaan vain yksi. Mahdollisena vaarana on, että liian monet poikkeukset ja eroavuudet argumenteista riippuen tekevät kokonaisuudesta sekavan.

Tarjoaman suhteellinen tärkeys kertoi sen, kuinka paljon kokonaismallin osa alueet vaikuttavat arvojen toteutumiseen toisiinsa verrattuna. Suuremmat merkitykset ovatkin analyysissä argumenteilla ja kansiorakenteilla. Näiden avulla saavutettu selkeys ja läpinäkyvyys sekä ylläpidon ja skaalaamisen helpottaminen ovatkin vaikuttavia toimenpiteitä.

Viestintä mallin komponentit jäävät merkitykseltään pienemmiksi. Koska monet muut lähteet osoittavat viestinnän ja kommunikaation olevan niin merkittävässä asemassa organisaatioiden muutoksessa ja sitä myös robotiikan käyttöönotossa ja yleistymisessä, voi siis olla, että näiden kannalta ehdotetussa mallissa olisi parantamisen varaa. Toisaalta vastaajien lisäävät arvot, kuten robotiikan läpinäkyvyys, ovat jääneet pois tästä analyysistä ja niihin mallin viestintään liittyvät osa alueet olisivat voineet vaikuttaa enemmän. Vaikutusten arviointi perustuu myös vapaaseen, subjektiiviseen arvioon mikä voi suuresti vaikuttaa tuloksiin.



## 9 TULOKSET JA NIIDEN ARVIOINTI

### 9.1 Työn keskeiset tulokset

#### -Toimintamalli

Konkreettisena työn tuotoksena saatiin aikaiseksi toimintamalli robotiikan päivittäisen toiminnan selkiyttämiseksi ja yhtenäistämiseksi. Malliin suurin onnistuminen on siinä, että se on nyt olemassa. se keskittää robotiikkaan liittyvää kommunikaatiota ja ratkoo ainakin joitakin ylläpidon haasteita tarjoten siten teknisesti kestävämmät keinot robotiikan skaalaamiseen.

Keskeisenä ajatuksena on yhtenäinen todellista liiketoimintaa vastaava rakenne. Järkevän kokoisen yksiköiden, kuten liiketoimintayksiköiden tai tiimien, mukaan tehty jaottelu niin tuotanto- ja testausympäristön, robottien rakenteiden ja käyttämien tietojen, kuin viestintäkanavien suhteen lisää tehokkuutta uudelleenkäyttämällä komponentteja ja välttämällä toistoa. Malliin sisältyneet ja sen käsittelemät aiheet eivät ole kuitenkaan vielä kokonaisvaltainen ratkaisu organisaation robotiikkaan. Se keskittyy päivittäisen toiminnan näkökulman, niin viestintään kuin robottilaivueen hahmottamiseen ja ylläpitoon, ja näidenkin suhteen on vielä parantamisen varaa, kuten QFD:n tuloksista voidaan nähdä.

#### -Yhteistyö ja kommunikaatio

Niin kirjallisuus, kyselyn tulokset kuin asiantuntijoiden haastattelutkin vahvistavat kommunikaation ja yhteistyön keskeistä asemaa. Se jopa ylittää tekniset vaatimukset: *”On siis erittäin tärkeää, että yhteistyö sujuu. Ilman sitä ei onnistu. Se on jopa tärkeämpää tai yhtä tärkeää kuin hyvä infra. Totta kai se infra tarvitaan siihen, että näitä robotteja voidaan ylipäänsä tehdä, mutta nostaisin kuitenkin se kommunikoinnin itse ykkössijaan.”*

Robotiikka ei toimi tyhjiössä vaan sitoo yhteen ja yhdistää organisaatiota yhteisessä tavoitteessa lisätä niin työn tuottavuutta kuin mielekkyyttä. Jotta robotiikan skaalaaminen ja organisaation todellinen muutos onnistuvat, tarvitaan erityisesti johdon ja IT-arkkitehtuurin tuki. Käytännössä robotit saadaan ihmistyöntekijöiden käyttöön ja avuksi rakentamalla luottoa robotiikkaan.

Tämä vaatii ennen muuta yhteistyötä robotiikan kehityksen ja substanssiosaajien välillä. Tämä helpottuu, kun eri alojen asiantuntijat ymmärtävät toisiaan. Kun RPA-kehittäjä ja vaikkapa taloudenasiantuntija ymmärtävät toistensa erikoisosaamisalueita, saadaan koko toiminta läpinäkyvämmäksi ja esimerkiksi Orchestratoria ja jonoja hyödyntämällä robottien hallintaa ja valvontaa jaettua suuremmalle joukolle ihmisiä.

-Vastaukset tutkimuskysymyksiin

Toimintamallin ja tutkimuksen löydösten pohjalta voidaan vastata tutkimuskysymyksiin. Tutkimuksen keskeinen tarkoitus oli siis selvittää, miten jatkuvaa ja kasvavaa robotiikkaprosessien laivuetta on tehokasta hallita. Yksinkertainen vastaus tähän oli kehittää skaalattavissa oleva toimintamalli. Siten alakysymykset jakautuvat täsmentämään, millainen tämän mallin tulisi olla.

Ensimmäinen kysymyksistä koski sitä, miten seurannasta ja ylläpidosta tehdään mahdollisimman suoraviivaista. Mallin pohjalta tarkoin määritetyt ympäristöt, kansiorakenteet ja argumenttien käyttö nähtiin tässä tehokkaina keinoina, sillä ne selkeyttävät robottien toimintaa, sekä tehostavat ylläpitotoimenpiteitä, kun muutettavia robotteja on suhteellisesti vähemmän ja monissa muutostarpeissa riittää tiedostojen muokkaus. Tikettijärjestelmällä tarvituista toimenpiteistä on myös helpompaa pysyä ajan tasalla ja viestintäkanavilla tavoitetaan oikeat yhteyshenkilöt. Osaamisen ja ymmärryksen lisääminen robottien käyttäjien keskuudessa ja seurannan siirtäminen heille olisi kuitenkin merkittävin keino ylläpidon suoraviivaistamiseen ja tehostamiseen robotiikkaa skaalattaessa.

Toinen alakysymys teknisten toteutusten yhtenäistämisestä kytkeytyy vahvasti ensimmäiseen, sillä sen tarkoituksena on seurannan, ylläpidon ja skaalaamisen tehostaminen. Tähän vastasi mallista erityisesti ympäristöjen, kansiorakenteiden ja argumenttien käyttö. Lisäksi parannetusta ehdotuksesta yhtenäistetty REF ja kirjastot vastaavat tähän. Mitä enemmän toteutuksia saadaan yhtenäistettyä, sen parempi tehokkuuden kannalta, mutta liiketoiminnan joustavat ja vaihtelevat tarpeet rajoittavat tätä.

Kolmas alakysymys koski viestintää, miten sidosryhmien kesken kommunikoidaan tuloksista, ongelmista ja kehityskohteista. Tähän kehitetty malli loi käytännön tasolla avuksi asiakasportaalit, viestintäkanavat sekä tikettijärjestelmän. Nämä selkeyttävät ja tehostavat toimintaa varsinkin RPA-tiimin kannalta. Yhteistyö ja ymmärrys sidosryhmien kesken on kuitenkin ratkaisevassa asemassa. Kun kehittäjä ja prosessin osaja ymmärtävät toisiaan, saadaan luotua hyviä ja luotettavia toteutuksia ja toiminnasta läpinäkyvää. Johdon sekä IT-arkkitehtuurin tukea vaaditaan varsinkin, kun robotiikkaa halutaan skaalata organisaatiossa.

## 9.2 Tulosten arviointi

Työ on siis toteutettu kirjallisuuskatsauksella, kyselyllä ja haastatteluilla. Tarkoituksena oli löytää yleisesti järkeviä keinoja ja olennaisia tekijöitä ohjelmistorobotiikan ylläpidon, valvonnan ja skaalaamisen järjestämiseen. Tutkimus on kuitenkin toteutettu taloushallinnon yritykseen. Kyselyyn vastanneita ja haastatteluun osallistuneita oli kuitenkin melko rajallinen määrä ja heidän taustansa yhtenäinen. Voi siis olla, että jollakin toisella alalla ja ympäristössä tilanne olisi hyvin erilainen. Kuitenkin tulosten ollessa niin yleismaallisia kuin, että toiminnalla kannattaa olla jonkinlainen malli ja että yhteistyö ja kommunikaatio sidosryhmien kanssa ovat avain onnistumiseen, on todennäköistä, että ne pätevät muuallakin.

Teknisissä näkökulmissa oltiin hyvin UiPath keskeisiä. Käytettiin tämän RPA ohjelmiston termejä, lähteitä ja dokumentaatiota, mutta sen ollessa markkinajohtaja, on myös todennäköistä, että lukuisat muut toimittajat rakentavat vastaavia malleja, joilla voisi toteuttaa samoja ideoita. Toisaalta esiteltiin muita liiketoiminnan prosessien tuottavuuden parantamisen menetelmiä, ja samojen elementtien, esimerkiksi yhteistyön, kommunikaation ja kokonaisvaltaisten ratkaisujen, merkitys kuvattiin suureksi myös.

Käyttöön otettujen toimintamallien ja käytäntöjen toteaminen onnistuneeksi on vielä tässä vaiheessa ennaikaista. Varsinkin viestinnän suhteen uusi toimintatapa on ollut aktiivisesti käytössä eri yksiköille vaihtelevan määrän aikaa, eikä vielä milläkään selvästi vakiintunut. Tarkat käytännöt ja tottumukset ovat siis vielä asettumassa, mutta toisaalta ratkaisujen iteratiivisuus onkin alusta alkaen ollut tarkoitus.

### 9.3 Jatkotoimenpiteet ja suositukset

Työssä jo saavutetut mallit ja toimintatavat koskevat siis lähinnä päivittäistä toimintaa. Ne pyrkivät hakemaan joitakin teknisiä seikkoja ja tehostamaan päivittäistä kommunikaatiota käytössä olevien robottien tilasta, ongelmatilanteista, jatkokehityksestä ja uusien automaatio kohteiden kartoittamisesta. Malli konkreettiseen skaalaamiseen ja liiketoiminnan automaation myötä muuttamisen toteuttamiselle strategisella, kokonaisvaltaisella tasolla jäi sen ulkopuolelle.

Tällainen toimintasuunnitelma olisi varmasti monelle jo selkeästi robotiikkaa käyttöön ottaneille organisaatiolle varsin hyödyllinen. Mallin olennaisena osana olisi luultavasti selvittää, miten yrityksen RPA-osaamista saadaan lisättyä tarpeen kasvaessa. suurin hyöty tietyllä yrityksellä saataisiin ehkä sillä, että yrityksessä jo työskentelevät teknologia orientoituneet työntekijät perehtyivät ja kouluttautuisivat ainakin ymmärtämään, ellei jopa käyttämään robotiikkaa.

Käytössä olevan robotiikan määrä tuskin ainakaan kymmenkertaistuu hetkessä, joten hyvä lähtöpiste on alkaa kouluttamaan ohjelmistorobotiikan perusteita ainakin osalle henkilöstöstä. Kun edes robotinkäyttäjille annetaan pääsy ja perehdytys Orchestratoiin tai vastaavaan robotiikan hallintajärjestelmään, auttaa siirtämään robotiikan valvontaa laajemmalle joukolle ihmisiä, mikä puolestaan suuntaisi usein rajallisten, varsinaisten robotikehittäjien työtä uusiin toteutuksiin, mikä puolestaan lisää robotiikkaa ja sen hyötyjä organisaatiossa.

## 10 YHTEENVETO

Lukuisat organisaatiot ja tutkimukset ovat todenneet robotiikan hyödyn. Sen ajatus toteuttaa sääntöpohjaisia rutiinitehtäviä ihmistä jäljentaen on ilmeisen käytännönläheinen. Kuitenkin, kun näiden toteutusten määrä kasvaa, on huomattu, ettei saadut hyödyt yllä odotetulle tasolle. Pääsyytä tälle on muun muassa, että ideaalisen kehitysprosessit ovat rajallisia, ylläpidon määrä kasvaa ja osaamista ei ole tarpeeksi saatavilla. Tämän pullonkaulan ylittämiseksi tarvitaan ennen muuta selkeä toiminnan malli. Robotiikkaa tulee toteuttaa tavalla, joka mahdollistaa suuren joustavuuden ja skaalaamisen. Ideaalisia automaation kohteita ei löydy kuin rajallinen määrä, jos robotiikan hyötyjä lasketaan prosessi kerrallaan. Tarvitaan siis kokonaisvaltaisempaa tarkastelua ja vahvaa yhteistyötä organisaation eri yksiköiden välillä, sillä vaikka yksittäisen prosessin hyödyt eivät olisi suuria, laajasti automatisoidut tehtävät parantavat tuottavuutta.

Työn konkreettisena tavoitteena oli luoda toimintamalli päivittäisiin robotiikan teknisiin ylläpidon ja skaalaamiseen tarpeisiin sekä kommunikaation sidosryhmien kanssa. Tässä keskeiseksi ajatukseksi nousi hierarkkisen todellista liiketoimintaa heijastelevan mallin hyödyntäminen niin tietorakenteiden yhtenäistämiseen kuin viestinnän keskittämiseen. Lisäksi toteutettiin teknisiä keinoja, joilla robottien joustavuutta ja ylläpitoa etenkin muutostarpeiden suhteen saatiin lisättyä.

Kaiken sitovaksi teemaksi nousi kuitenkin selvästi yhteistyö ja kommunikaatio. Kun automaation edistäminen on koko organisaation yhteinen tarkoitus ja tahtotila, ja eri sidosryhmät ymmärtävät toisiaan, helpottuvat niin päivittäisten ongelmatilanteiden ja haasteiden ratkaiseminen, jatkokehitys, uusien toteutuskohteiden löytäminen kuin laajempi organisaatioon muutos kohti tuottavampaa ja mielekkäämpää työtä. Tämän kannalta konkreettinen toimi on kasvattaa robotiikkaosaamista työntekijöiden keskuudessa, jotta heitä palvelevan robotiikan toiminta olisi läpinäkyvää ja luotettavaa ja he pystyisivätkin hoitamaan valtaosan robotiikan vaatimasta valvonta- ja ylläpitotyöstä itse. Silloin robotiikan kehittäjät voivat keskittyä tekemään uusia toteutuksia, viemään automaatiota organisaatiossa pidemmälle ja siten parantamaan työyhteisön roboteista saamaa hyötyä ennestään.

## LÄHTEET

Aguirre, S. & Rodriguez, A. 2017. 'Automation of a Business Process Using Robotic Process Automation (RPA): A Case Study', in *Applied Computer Sciences in Engineering*. Cham: Springer International Publishing. pp. 65–71.

Antonoaie, V., Irimeș, A., Chicoș, L-A., Drăgoi, M.V. & Oancea, G. 2017. ERP processes automation in corporate environments. MATEC web of conferences. 946001–

Bickman, L. & Rog, D. J. 2009. *The SAGE handbook of applied social research methods*. 2nd ed. Los Angeles SAGE.

Byrne, D. S. & Ragin, C. C. 2009. *The SAGE handbook of case-based methods*. 2nd ed. Los Angeles. SAGE.

Cabello, R., Escalona, M. J. & Enríquez, J. G. (2020) 'Beyond the Hype: RPA Horizon for Robot-Human Interaction', in *Business Process Management: Blockchain and Robotic Process Automation Forum*. Cham: Springer International Publishing. pp. 185–199.

Carnevalli, J. A. & Miguel, P. C. 2008. Review, analysis, and classification of the literature on QFD—Types of research, difficulties, and benefits. *International journal of production economics*. 114 (2), 737–754.

De, B. & Doda, R. 2017. *API Management: An Architect's Guide to Developing and Managing APIs for Your Organization*. Berkeley, CA: Apress L. P.

Dov Te'eni 2001. Review: A Cognitive-Affective Model of Organizational Communication for Designing IT. *MIS quarterly*. 25 (2), 251–312.

Dumas, M. La Rosa, M., Mendling, J., Reijers, H. A. 2018. *Fundamentals of Business Process Management*. 2nd ed. 2018. Berlin, Heidelberg: Springer Berlin Heidelberg.

G2 2021. Best Robotic Process Automation (RPA) Software [Verkkosivu].[viitattu 12.6.2021].

Saatavissa:

<https://www.g2.com/categories/robotic-process-automation-RPA>

Grubb, P. & Takang, A. A. 2003. Software maintenance: concepts and practice. 2nd ed. Singapore: World Scientific Publishing Co. Pte. Ltd.

G. Schuetz, C. 2015. Multilevel Business Processes Modeling and Data Analysis. 1st ed. 2015. Wiesbaden: Springer Fachmedien Wiesbaden.

Heijari, Mikko 2020. Ohjelmistorobotiikka suomalaisessa yrityskentässä. LUT yliopisto

Hofmann, P. Samp, C, Urbach, N. 2020. Robotic process automation. Electronic markets. 30 (1), 99–106.

IBM 2020. CIO perspective: To be great at IT, you must be great at BPM.

Kauppila 2020 Älykkäiden teknologioiden onnistunut käyttöönotto taloushallinnossa. LUT yliopisto

Kohdeyritys 2020. Sisäinen dokumentti

Kohdeyritys 2021 Asiakasportaali

Korol Julitta 2017. ‘Excel Macros’, in Microsoft® Excel® 2016 - Programming by Example with VBA, XML, and ASP. Mercury Learning and Information.

Mattila, Lotta 2019. Ohjelmistorobotiikan käyttöönotto taloushallinnossa. LUT yliopisto.

Moe, N. B. Dingsøyr, T., Dybå, T., 2010. Agile Software Development: Current Research and Future Directions. 1. Aufl. Berlin, Heidelberg: Springer-Verlag.

Microsoft 2021a. Microsoft Teams [Verkkosivu].[viitattu 12.6.2021]. Saatavissa: <https://www.microsoft.com/en-ww/microsoft-teams/group-chat-software>

Microsoft 2021b. Power Automate [Verkkosivu].[viitattu 12.6.2021]. Saatavissa: <https://flow.microsoft.com/en-us/>

Mäkelä, Jussi 2019. Ohjelmistorobotiikkaliiketoiminnan muodostuminen ja sen hinnoittelussa esiintyvien haasteiden kartoitus, Tampere University

Jablin, F. M. & Putnam, L. L. 2000. *New Handbook of Organizational Communication: Advances in Theory, Research, and Methods*. Los Angeles: SAGE Publications.

Poussa, Henri 2020. *Challenges of scaling robotic process automation*. LUT yliopisto.

Rutschi, C. & Dibbern, J. 2020. 'Towards an Understanding of Scaling the Software Robot Implementation', in *Information Systems Outsourcing*. Cham: Springer International Publishing. pp. 453–466.

Tanner, G. & Otto, K. 2016. Superior-subordinate communication during organizational change: under which conditions does high-quality communication become important? *International journal of human resource management*. 27 (19), 2183–2201.

Taulli, T. 2020. *The Robotic Process Automation Handbook: A Guide to Implementing RPA Systems*. 1st edition. Berkeley, CA:

Syed, R. & Wynn, M. T. 2020. 'How to Trust a Bot: An RPA User Perspective', in *Business Process Management: Blockchain and Robotic Process Automation Forum*. Cham: Springer International Publishing. pp. 147–160.

Syed, R. Suriadi, S., Adams, M., Bandara, W., Leemans, S. J.J., Ouyang, C., Hofstede, A., H.M 2020. *Robotic Process Automation: Contemporary themes and challenges*. Computers in industry. 115103162–.



Trust radius 2021. Robotic Process Automation (RPA) Software Overview [Verkkosivu].[viitattu 12.6.2021]. Saatavissa: <https://www.trustradius.com/robotic-process-automation-RPA#overview>

TIBCO Software 2021. Product documentation TIBCO ActiveMatrix BusinessWorks [Verkkosivu].[viitattu 12.6.2021]. Saatavissa: <https://docs.tibco.com/products/tibco-activematrix-businessworks>

Tuominen Tinja 2020. Ohjelmistorobotiikan vaikutus yrityksen taloushallintoon. LUT yliopisto.

UiPath 2020a. UiPath-Scaling-RPA\_whitepaper

UiPath 2020b. Step-by-Step Guide – Upgrading your RPA CoE to Scale Automation

UiPath 2021a. kotisivu, linkki:

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

UiPath 2021b. Playbook: Continuous Automation, Continuous Testing

UiPath 2021c. Attended vs Unattended [Verkkosivu].[viitattu 12.6.2021]. Saatavissa: <https://docs.uipath.com/robot/docs/attended-vs-unattended>

UiPath 2021d. Orchestrator Introduction, [Verkkosivu].[viitattu 12.6.2021]. Saatavissa: <https://docs.uipath.com/orchestrator/docs/introduction>

UiPath 2021e. Robotic Enterprise Framework [Verkkosivu].[viitattu 12.6.2021]. Saatavissa: <https://docs.uipath.com/studio/docs/robotic-enterprise-framework>

Vanhala, M., Heilmann, P., Salminen, H. 2016. Organizational Trust Dimensions as Antecedents of Organizational Commitment. *Knowledge and Process Management*, vol. 23, iss. 1, pp. 46-61. DOI: 10.1002/kpm.1497

## Liite 1 haastattelukysymykset ja haastateltavat

### Haastattelukysymykset:

1. Haastateltavan tausta / kokemusrobotiikasta
2. Mitkä asiat vahvistavat ja edistävät robotiikan skaalausta ja ylläpitoa
3. Mitkä asiat tekevät robotiikan skaalausta ja ylläpitoa vaikeammaksi äskeisten vastakohtien taikka puuttumisen lisäksi?
4. Olisivatko nämä ylläpidon kannalta helpottavia toimenpiteitä? Kyllä – ei, miksi
  - Yhtenäinen ja hierarkkinen kansiorakenne robottien käyttämistä tiedoista
  - Etäasema-konfigurointitiedosto
  - Jonojen käyttö
5. Muuta

### Haastateltavat:

1. Haastateltava A, Robotiikka ja AI asiantuntija sekä esimies, 10.5.2021
2. Haastateltava B, RPA projektipäällikkö, 11.5.2021
3. Haastateltava C, RPA konsultti ja teknologiavastaava, 12.5.2021

## Liite 2 kyselyn vastaukset

| ONNISTUMISOSUUS. | NOPEUS.      | ROBOTTI SAA           | STANDARDIT.               | ROBOTIT                                       | RAPORTOINTI.           | MAHDOLLISIMMAN | MAHDOLLISIMMAN | MAHDOLLISIMMAN | SIJOTTUKSEN | ROBOTIN       | JOS MIELESI TUULEE  | RPA VESTINTÄKANAVA | TIKETTIJÄRJESTELMÄ | MITEN KOMMUNIKAATIOMALLI VOIDAAN EDLLEEN |  |
|------------------|--------------|-----------------------|---------------------------|---|------------------------|----------------|----------------|----------------|-------------|---------------|---|--------------------|--------------------|--|--|
| Mahdollisimman   | korkea osuus | prosessista onnistuu. | tehtävän hoidettua paljon | nopeammin, kuin ihmiset manuaalisella työllä. | ymmärtänyt käyttänyt.  | positiivinen   | negatiivinen   | positiivinen   | (ROI).      | mutuokse      | muuta tärkeitä  | raportointi        | raportointi        | parantaa?                                |  |
|                  |              |                       | yhentäisi, eli            | roboetta on                                   | jaaneista ja           | ilmoittaa      | ilmoittaa      | ilmoittaa      | rahalinen   | onnistuu      | asioita robotiikkaan  | epäonnistuu        | epäonnistuu        | epäonnistuu                              |  |
|                  |              |                       | raportointi toimii        | prosessista on                                | ongelmasta/yrityksestä | epäonnistuu.   | säästö on      | merkittävä     | ketterä.    | prioriteetti. |   |                    |                    |  |  |
|                  |              |                       | samanlaisesti.            | selkeää.                                      |                        |                |                |                |             |               |   |                    |                    |  |  |
| Priority 2       | Priority 4   | Priority 1            | Priority 7                | Priority 3                                    | Priority 2             | Priority 1     | Priority 2     | Priority 1     | Priority 3  | Priority 1    | Robottien työnoo on asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. (1) Robottikalenteri ja vastuuhenkilöt (2) sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi (3). | Tulee parantamaan  | Vähän parantanut   | Tulee parantamaan                        | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi. |
| Priority 4       | Priority 7   | Priority 1            | Priority 2                | Priority 5                                    | Priority 3             | Priority 2     | Priority 3     | Priority 2     | Priority 6  | Priority 5    | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi.                | Tulee parantamaan  | Vähän parantanut   | Tulee parantamaan                        | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi. |
| Priority 6       | Priority 1   | Priority 2            | Priority 3                | Priority 2                                    | Priority 2             | Priority 5     | Priority 3     | Priority 2     | Priority 2  | Priority 7    | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi.                | Tulee parantamaan  | Vähän parantanut   | Tulee parantamaan                        | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi. |
| Priority 2       | Priority 3   | Priority 6            | Priority 3                | Priority 2                                    | Priority 7             | Priority 6     | Priority 7     | Priority 6     | Priority 4  | Priority 5    | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi.                | Tulee parantamaan  | Vähän parantanut   | Tulee parantamaan                        | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi. |
| Priority 1       | Priority 7   | Priority 4            | Priority 5                | Priority 1                                    | Priority 5             | Priority 3     | Priority 7     | Priority 3     | Priority 3  | Priority 4    | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi.                | Tulee parantamaan  | Vähän parantanut   | Tulee parantamaan                        | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi. |
| Priority 3       | Priority 2   | Priority 7            | Priority 4                | Priority 3                                    | Priority 6             | Priority 1     | Priority 5     | Priority 1     | Priority 5  | Priority 4    | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi.                | Tulee parantamaan  | Vähän parantanut   | Tulee parantamaan                        | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi. |
| Priority 7       | Priority 6   | Priority 4            | Priority 5                | Priority 6                                    | Priority 5             | Priority 7     | Priority 5     | Priority 5     | Priority 7  | Priority 6    | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi.                | Tulee parantamaan  | Vähän parantanut   | Tulee parantamaan                        | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi. |
| Priority 3       | Priority 1   | Priority 4            | Priority 4                | Priority 6                                    | Priority 5             | Priority 7     | Priority 5     | Priority 5     | Priority 7  | Priority 7    | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi.                | Tulee parantamaan  | Vähän parantanut   | Tulee parantamaan                        | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi. |
| Priority 2       | Priority 6   | Priority 2            | Priority 2                | Priority 2                                    | Priority 4             | Priority 1     | Priority 1     | Priority 1     | Priority 2  | Priority 3    | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi.                | Tulee parantamaan  | Vähän parantanut   | Tulee parantamaan                        | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi. |
| Priority 2       | Priority 4   | Priority 3            | Priority 3                | Priority 1                                    | Priority 3             | Priority 2     | Priority 2     | Priority 2     | Priority 2  | Priority 2    | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi.                | Tulee parantamaan  | Vähän parantanut   | Tulee parantamaan                        | Robottien työnoo asetamisen ja tulosten hallinnoinnin yhtenäistämisen eri prosessien välillä. Robottikalenteri ja vastuuhenkilöt sekä päivityskalenteri eri sidosryhmien kanssa (2) sekä poikkeustilannefoorumi. |