

INTEGRAATIO OSANA PALKKAPROSESSIN AUTOMAATIOITA



Ammattikorkeakoulututkinnon opinnäytetyö

Tietojenkäsittelyn koulutus, Hämeenlinnan korkeakoulukeskus
syksy, 2021

Petri Rantanen

TIIVISTELMÄ

Opinnäytetyön tarkoituksena oli selvittää, mitä on integraatio ja liiketoimintaprosessien automaatio sekä miten niitä voidaan hyödyntää nykyaikaisessa palkanlaskennassa. Työssä käsitellään myös palkanlaskentaa prosessina sekä esitellään moderni tapa tehdä integraatiota integraatioalustalla. Työn toimeksiantajana oli Accountor Services Oy.

Opinnäytetyön teoriaosuudessa kerrotaan, mitä on integraatio ja esitellään integraatiomallit, mitä tarkoittaa liiketoimintaprosessien automaatio, mikä on palkkahallinnon kokonaisprosessi, miten integraatioita ja automaatioita käytetään palkkaprosessissa sekä esitellään integraatioalustat ja niiden toimitusmuodot. Lisäksi esitellään käytännön osan työvälineenä käytetty Friends-integraatioalusta. Opinnäytetyö on toiminnallinen.

Johtopäätöksenä voidaan todeta, että automaatioiden osuus sujuvassa liiketoimintaprosessissa on merkittävä. Integraatiot ovat edelleen isossa roolissa, kun suunnitellaan automaattisia prosesseja ja niiden sujuvuus sekä toimintavarmuus entistä tärkeämpiä. Automatisoidut prosessit vapauttavat ihmisten aikaa tuottaa prosessissa lisäarvoa tuottavia palveluita ja tehtäviä.

Työn tuotoksena valmistui työn tilaajan käyttöön tiedostosiirtointegraatio Friends-integraatioalustaa käyttäen sekä tiedostosiirtojen hallintaan Sharepoint Power Apps -sovelluksena toteutettu käyttöliittymä.

Avainsanat Automaatio, integraatio, liiketoimintaprosessi, palkanlaskenta

Sivut 61 sivua ja liitteitä 1 sivua

Author Petri Rantanen

Year 2021

Subject Integration as part of automation in a payroll process

Supervisors Lasse Seppänen

ABSTRACT

Purpose of this thesis is to find out what is system integration and business process automation and how one can use them in modern payroll. Thesis also introduces payroll as a process and a modern way to do integrations with integration platform. The thesis was commissioned by Accountor Services Oy.

Thesis defines what is system integration and what are integration models, what means business process automation, what is payroll process, how integrations and automation can be used in payroll process and what is integration platform. In the thesis integration platform Frends is also introduced. Frends has been used to configurate practical part of the thesis. The thesis is practical.

The conclusion was that automations are a vital part of the fluent business process. Reliable and flowing integrations have a big and meaningful role when planning the process automation. Automated processes give humans more time to do value adding tasks and services.

As a practical part of the thesis a new file transfer integration and a user interface for ordering and updating file transfers was planned for the Accountor Services Oy. The file transfer integration was made with integration platform Frends and the user interface with Sharepoint Power Apps.

Keywords Automation, system integration, business process, payroll

Pages 61 pages and appendices 1 pages

Sanasto

| | |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| .NET | Microsoftin kehittämä ohjelmistokomponenttikirjasto |
| Automaatio | Koneiden, laitteiden ja järjestelmien toimintaa ilman ihmisen suoranaista vaikutusta |
| AI | Artificial Intelligence eli tekoäly |
| API | Application Programming Interface eli ohjelmointirajapinta |
| BPA | Business Process Automation eli liiketoimintaprosessien automaatio |
| BPM | Business Process Management eli liiketoimintaprosessien hallinta |
| C# | Ohjelmointikieli |
| CSV | Comma-Separated Values -tiedostomuoto |
| DPA | Digital Process Automation eli digitaalinen prosessiautomaatio |
| EAI | Enterprise Application Integration eli järjestelmäintegraatio |
| EiPaaS | Enterprise Integration Platform as a Service eli suuryrityksille suunnattu pilvipohjainen integraatiopalvelu |
| HTML | Hypertext Markup Language -merkintäkieli |
| Hyperautomaatio | RPA, EAI ja AI yhdistelmä |
| Integraatio | Ohjelmisto on yhdistetty toiseen ohjelmistoon tietojen siirtämistä varten |
| IPA | Intelligent Process Automation eli älykäs prosessiautomaatio |
| iPaaS | Integration Platform as a Service eli pilvipohjainen integraatiopalvelu |
| JSON | Javascript Object Notation tiedon esitysmuoto |
| ML | Machine Learning eli koneoppiminen |
| NuGet | Microsoftin kehitysympäristöön tehty pakettienhallintaohjelma |
| Prosessi | Loogisesti toisiinsa liittyvät toiminnot ja niiden saavuttamiseen tarvittavat resurssit, joiden avulla saadaan aikaan asetettuja tavoitteita vastaavat tulokset |
| Rajapinta | Rajapinnan avulla voidaan tehdä pyyntöjä ohjelmistolle, josta halutaan noutaa tai tuoda tietoja |

| | |
|------|---------------------------------------------------------------------------------------|
| REST | Representational State Transfer -protokolla ohjelmointirajapintojen toteuttamiseen |
| RPA | Robotic Process Automation eli ohjelmistorobotiikka |
| RSS | Really Simple Syndication. XML-kieleen perustuva syötemuoto |
| SOA | Service Oriented Architecture eli palvelukeskeinen malli |
| SOAP | Simple Object Access Protocol eli palvelukeskeinen arkkitehtuuri |
| SSIS | SQL Server Integration Services eli tietojen integrointityökalu |
| XML | Extensible Markup Language -merkintäkieli |

Sisällys

| | | |
|-------|-------------------------------------------------------------------|----|
| 1 | Johdanto | 1 |
| 2 | Integraatio | 2 |
| 2.1 | Point-to-point-integraatiomalli | 3 |
| 2.2 | Hub and spoke -integraatiomalli | 4 |
| 2.3 | Palveluväylä eli Enterprise service bus | 5 |
| 2.4 | SOA, SOAP ja REST | 6 |
| 2.5 | API-rajapinnat | 7 |
| 3 | Liiketoimintaprosessien automaatio | 8 |
| 3.1 | RPA | 9 |
| 3.2 | Älykäs automaatio | 10 |
| 3.3 | Hyperautomaatio | 11 |
| 4 | Palkanlaskennan kokonaisprosessi | 12 |
| 4.1 | Henkilöstöhallinnon tehtävät | 12 |
| 4.2 | Palkkahallinnon ja palkanlaskennan tehtävät | 13 |
| 4.3 | Palkanlaskennan prosessi | 13 |
| 5 | Integraatiot ja automaatio palkkaprozessissa | 15 |
| 5.1 | Integraatiot ja rajapinnat palkkaprozessin automaatiossa | 15 |
| 5.1.1 | HR-järjestelmäintegraatio | 16 |
| 5.1.2 | Integraatiot työajanseurantajärjestelmistä | 16 |
| 5.1.3 | Palkkojen raportointiin liittyvät integraatiot | 17 |
| 5.2 | RPA ja tekoäly palkkaprozessin automaatiossa | 17 |
| 6 | Integraatioalustat ja toimitusmuodot | 19 |
| 6.1 | Integraatioalustojen toimitusmuodot | 20 |
| 6.2 | Integraatioalustojen tietoturva | 21 |
| 7 | Frends-integraatioalusta | 23 |
| 7.1 | Frends-integraatiot | 24 |
| 7.2 | Frends-agentit ja -liittimet | 26 |
| 7.3 | Frends-integraatioalustan toimitusmuodot | 27 |
| 7.4 | Frends tietoturva | 28 |
| 8 | Kehittämistyön tavoite ja tarkoitus | 29 |
| 9 | Yleistiedostosiirtointegraatiot Frends-integraatioalustalla | 31 |
| 9.1 | Käyttöliittymä | 33 |
| 9.2 | Frends-ylläpitointegraatio | 34 |

| | | |
|------|-----------------------------------------------------|----|
| 9.3 | Frends-tiedostosiirtointegraatio | 39 |
| 9.4 | Tuotoksen tietoturva ja käyttäjäystävällisyys | 44 |
| 10 | Jatkokehitys, johtopäätökset ja pohdinta | 46 |
| 10.1 | Johtopäätökset..... | 48 |
| 10.2 | Pohdinta | 49 |
| 11 | Yhteenveto | 51 |
| | Lähteet..... | 52 |

Kuvat

| | |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Kuva 1. Point-to-point-integraatiomalli (Flashnode, 2021). | 3 |
| Kuva 2. Hub and spoke -integraatiomalli (Flashnode, 2021). | 4 |
| Kuva 3. ESB-integraatiomalli. 1. Lähtö- ja kohdejärjestelmät. 2. Tiedonsiirtokerros. 3. Rajapinnat. 4. Tiedon käsittelyt ja muunnokset. 5. Prosessihallinta (Flashnode, 2021)... | 5 |
| Kuva 4. Ohjelmistorobotin käyttökohteita (Kääriäinen, ym., 2018). | 10 |
| Kuva 5. Hyperautomaation osa-alueet (Pursiainen, 2020). | 11 |
| Kuva 6. Integraatioalustan toiminnan mallikuva (Toivanen, HiQ, n.d -b). | 19 |
| Kuva 7. Esimerkki Frends-prosessin vuokaaviosta (HiQ, n.d -b). | 23 |
| Kuva 8. Esimerkki Frends-prosessiautomaatiosta (Frends, n.d -a). | 24 |
| Kuva 9. Esimerkki Frends-tiedostosiirrosta (Frends, n.d -a)..... | 25 |
| Kuva 10. Malli yksinkertaisesta prosessista. Prosessi alkaa ja sen aikana suoritetaan kolme tehtävää, jonka jälkeen prosessi päättyy. | 26 |
| Kuva 11. Malli eksklusiivisesta päätöksestä, jossa suoritetaan tehtävä 2 tai 3, riippuen tehtävän 1 tuloksesta. | 26 |
| Kuva 12. Yleissiirto integraation kokonaisprosessi..... | 32 |
| Kuva 13. Sharepoint Power Apps -käyttöliittymä Frends-integraatioiden tilaamiseen.. | 34 |
| Kuva 14. Frends-ylläpitointegraatio ensimmäiset tehtävät. | 35 |
| Kuva 15. Frends-ylläpitointegraatio toisen vaiheen tehtävät. | 36 |
| Kuva 16. Kansio-oikeuksien tarkistamisen aliprosessi. | 37 |
| Kuva 17. Frends-integraatio 1 kolmannen vaiheen tehtävät. | 37 |
| Kuva 18. Frends-integraatio 1 prosessin viimeiset vaiheet..... | 38 |
| Kuva 19. Tiedostojen vertailun aliprosessi. | 38 |
| Kuva 20. Toteutettu Frends-yleissiirtointegraatio. | 40 |
| Kuva 21. Yleissiirtointegraation aloitus. | 41 |
| Kuva 22. For each -loopin muuttujien sijoittaminen ja siirron ehtojen tutkiminen. | 42 |
| Kuva 23. For each -silmukan sisällä oleva Scope, jossa tiedostosiirrot tapahtuvat. | 43 |
| Kuva 24. Catch-tehtävä ja Scope virnehallintaa varten. | 44 |

Liitteet

Liite 1 Friends-integraatioiden mallikuvat

Liite 2 Aineistonhallintasuunnitelma

1 Johdanto

Automaatiot ja integraatiot kasvattavat rooliaan jokapäiväisessä tekemisessä koko ajan enemmän ja enemmän. Automaatiolla voidaan joko suoraan korvata ihmisen tekemää työtä, siirtää sille toistuvia yksinkertaisia helposti monistettavia tehtäviä tai lisätä laatua tuomalla automaatio suorittamaan erityistä tarkkuutta vaativiin tehtäviin. Integraatioilla voidaan liittää kaksi keskenään erilaista ohjelmaa yhteen.

Palkanlaskenta ja palkkaprozessit sisältävät paljon toistuvia ja säännönmukaisia tehtäviä, jotka ovat erittäin otollisia automaation kohteita. Palkkojen lähetys pankkiin, viranomaisraportointi tai veron laskenta ovat hyviä esimerkkejä prosessin osista, joista löytyy standardit ja säännöt, miten ne tehdään. Palkkajärjestelmät itsessään sisältävät paljon automaatioita, mutta automaatio kokonaisprosessin ympärillä kasvaa koko ajan.

Tässä opinnäytetyössä selvitetään, mitä on liiketoimintaprosessien automaatio ja integraatio, esitellään palkkaprozessin pääpiirteet sekä nostetaan esiin kohteita palkkaprozessista missä automaatiota käytetään tai voidaan käyttää. Työ toteutetaan Accountor Services Oy:n palkkapalvelulle. Accountor palkkapalvelussa automaatiota on tuotu mukaan ammattilaisten päivittäisen työn tueksi ja vapauttamaan heidän aikaansa tulkintaa sekä erityisosaamista vaativiin tehtäviin.

Osana tätä muutosta ovat integraatiot. Integraatiot ovat toistaiseksi lähinnä olleet tekemässä taustaprosesseja, kuten esimerkiksi tiedostojen siirtoa paikasta A paikkaan B. Accountor Services Oy:ssä integraatiot ovat olleet aikaisemmin ICT-yksikön toteuttamia ja heidän hallinnassaan. Työn lopputuotoksena rakennetaan Accountor palkkapalvelun omaan hallintaan tiedostosiirtointegraatio ja näiden hallintaan käyttöliittymä.

Opinnäytetyöt tutkimuskysymyksiä ovat mitä on integraatio, mitä on liiketoimintaprosessien automaatio, miten ja missä integraatiota ja automaatiota voidaan hyödyntää palkkaprozessissa sekä miten tehdään Friends-järjestelmällä tietoturvallinen ja käyttäjäystävällinen integraatio?

2 Integraatio

EAI eli Enterprise Application Integration tarkoittaa sitä, että integraation avulla keskenään toimimattomat järjestelmät saadaan yhdistettyä toisiinsa (Flashnode, 2021).

Järjestelmäintegraatiolla yhdistettyjen järjestelmien välillä voidaan siirtää tietoa yhdestä järjestelmästä toiseen tai useaan järjestelmään. Integraatiota voidaankin kutsua tietovirraksi. (Toivanen, n.d -a) Integraatiossa keskenään keskustelevat järjestelmät on usein toteutettu erilaisin tekniikoin tai eri alustoilla. Integroinnin tarkoituksena onkin taata tiedonkulku järjestelmästä toiseen. (Haglund, 2021)

Integraatitoteutuksiin löytyy erilaisia malleja. Esimerkiksi Point-to-point-integraatio, Hub-and-Spoke-integraatio ja SOA, joka perustuu palveluihin. Viimeisin on kuitenkin enemmän tapa suunnitella integraatiota, kuin itse malli. (Flashnode, 2021) Muita malleja on muun muassa ESB eli Enterprise Service Bus ja erilaiset sisäiset tai julkiset rajapinnat eli API:t (Toivanen, n.d -a). Oikean mallin valintaan vaikuttavat integraation piiriin tulevien järjestelmien ominaisuudet ja mahdollisuudet muun muassa rajapintoihin sekä millaista tietoa niissä säilytetään ja tuleeko käyttöön jokin integraatioalusta (Flashnode, 2021).

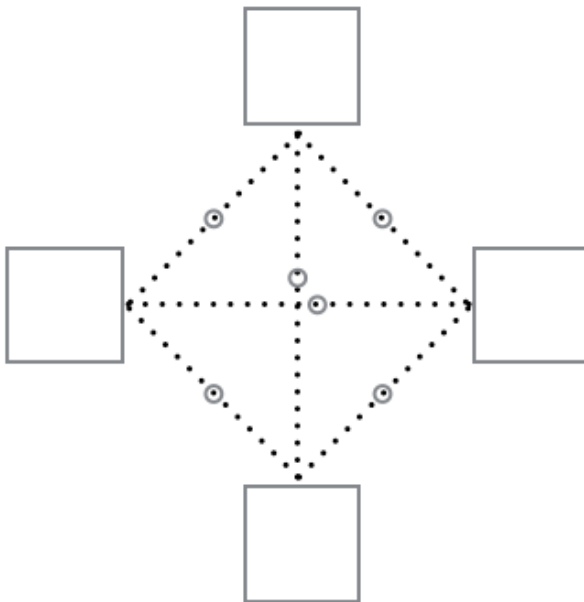
Integraation mallia ja toteutustapaa suunniteltaessa täytyy ottaa huomioon, kuinka monta järjestelmää integraatio piiriin tulee sekä millaiset ovat integraatio skaalautuvuus- ja kasvutarpeet. Integraatiosuunnittelussa onkin otettava huomioon, mitä dataa liikkuu, missä muodossa data on kussakin järjestelmässä, tarvitseeko integraatiossa tehdä datamuunnoksia, millaisia sääntöjä tarvitaan (triggerit, rajaukset ja viestintä) ja miten varmistetaan laadukas integraatio (tietoturva ja valvonta). (Flashnode, 2021)

Integraatiota voidaan rakentaa on-premises-järjestelmien ja pilvessä toimivien järjestelmien välille tai hybridimallina. On-premises-integraatioita ovat esimerkiksi lisenssiohjelmien väliset integraatiot sekä usein myös yrityksille rakennetut järjestelmät. Pilvi-integraatioita voivat olla esimerkiksi pilvessä toimivan verkkokaupan ja taloushallinto-ohjelman välinen integraatio. Hybridimallissa yhdistetään aiemmin hankittu on-premises-järjestelmä johonkin pilvipalveluun. (Flashnode, 2021)

2.1 Point-to-point-integraatiomalli

Point-to-point-integraatio (kuva 1) tarkoittaa kahden järjestelmän välistä suoraa integraatiota. Tässä toteutustavassa usean järjestelmän integraatiot ketjutetaan keskenään, kukin omalla liitännällä. Point-to-point-integraatio on edullinen rakentaa ja kustannustehokas. (Flashnode, 2021) Yhdestä yhteen integraatioissa ei käytetä erillisiä integraatioalustoja vaan ne rakennetaan suoraan itse ohjelmaan tai laitetaan erillisenä palvelimelle. Point-to-point-integraatio toimii tilanteissa, joissa halutaan rakentaa väliaikainen tai saman järjestelmätoimittajan kahden eri järjestelmän välinen integraatio. (Toivanen, n.d -a) Point-to-point-integraatiot ovat usein monimutkaisia ja heikosti hallittavia, koska ne sisältävät useita yksittäisiä integraatioita ja vanhaa tekniikkaa. Uusien järjestelmien liittäminen on hankalaa, koska jokainen uusi järjestelmä vaatii uuden integraation. Tämä heikentää point-to-point-integraatioiden skaalautuvuutta. (Flashnode, 2021)

Kuva 1. Point-to-point-integraatiomalli (Flashnode, 2021).

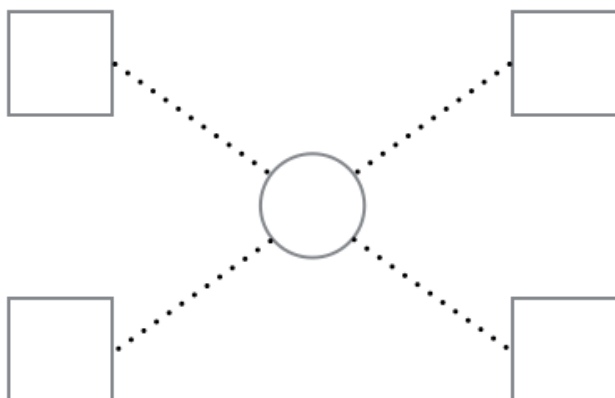


2.2 Hub and spoke -integraatiomalli

Hub and spoke -integraatiossa (kuva 2) järjestelmät eivät keskustele keskenään suoraan, kuten point-to-point-mallissa, vaan niiden välissä toimii niin sanottu viestinvälittäjä eli hub. Hubin tehtävänä on vastaanottaa viestit ja jakaa ne eteenpäin oikeaan osoitteeseen. (Flashnode, 2021) Tämä keskitetty integraatiomalli vaatii toimiakseen integraatioalustan, joka toimii datan liikuttajana, samalla valvoen datavirtoja. Keskitetty integraatio toimii yhdestä moneen -periaatteella, joka mahdollistaa datan valvonnan ja hallinnan integraatioalustalla. Keskitetty integraatio vähentää integraatioiden määrää, koska se pystyy jakamaan dataa useaan eri kohteeseen. (Toivanen, n.d -a) Koska järjestelmät eivät ole riippuvaisia toisistaan, voidaan niihin tehdä muutoksia, vaarantamatta integraation toimivuutta (O'brien, 2008).

Integraatioalustan toimiessa hubina, tarjoaa se mahdollisuuden muokata siirrettävää tietoa paremmin kuin suorissa integraatioissa. Esimerkiksi SFTP-protokollan avulla sisään tuotu tieto voidaan lähettää eteenpäin REST-kutsuna. Integraatioalustoissa voidaan integraatioita käynnistää eri tavoin. Esimerkiksi integraatio voi olla ajastettu tai se voi käynnistyä REST/JSON-kutsusta, tiedoston ilmestyessä lähtösijaintiin tai tietokannassa tapahtuneesta muutoksesta. (Toivanen, n.d -a) Hub and spoke -mallin heikkoudet löytyvät hubin käytöstä. Kaikki data liikkuu yhden ja saman hubin kautta, jolloin vikatilanteissa tiedonsiirto vaarantuu, datamuodot rajoittuvat hubin kyvykkyyksiin ja hajautetuissa organisaatioissa maantieteelliset välimatkat tuovat omat haasteensa. (Flashnode, 2021)

Kuva 2. Hub and spoke -integraatiomalli (Flashnode, 2021).



2.3 Palveluväylä eli Enterprise service bus

Palveluväylä eli ESB-integraatiomalli (kuva 3) tarkoittaa sitä, että integraatio pystyy välittämään dataa erilaisin tavoin. Integraatioalusta voi esimerkiksi laittaa saapuvan datan jonoon, ja toimittaa sen eteenpäin, kun kohdejärjestelmä on valmis ottamaan sitä vastaan. Jonojen lisäksi ESB-mallissa voidaan dataa toimittaa tapahtumapohjaisesti suoraan lähtöjärjestelmästä integraatioalustan REST/JSON-rajapinnan avulla kohdejärjestelmään. Kolmas tapa välittää tietoa palveluväylää pitkin on publish-subscribe-viestinvälitys. Tässä välitystavassa järjestelmä, joka haluaa vastaanottaa tietoa, ilmoittautuu kuuntelijaksi eri viesteille aiheen (topic) tai sisällön (content) perusteella. Palveluväylä reitittää sen kautta kulkevat viestit sen mukaan, mitä kukin kuuntelija kaipaa. (Toivanen, n.d -a)

Palveluväylämalli poistaa hub and spoke -integraatioissa sisältyviä ongelmia, koska se osaa itse hajauttaa integraation eri vaiheet omikseen. Tällöin siinä on sisällä ikään kuin useita eri hubeja. ESB-alustalla voidaan muokata tietoa ja sitä pystytään myös valvomaan sekä pilkkomaan osiin. Tieto voidaan jakaa muun muassa vikailmoituksiin, käsittelyyn tai muunnoksiin riippuen mitä osaa milloinkin tarvitaan. Palveluväylän suurimmat edut liittyvät hajautukseen. Päivitysten tekeminen on kevyempää ja vikojen korjaukset voidaan tehdä vain vialliseen osaan. Lisäksi ESB on hyvin skaalautuva ja helposti muokattavissa tarpeiden mukaan. Jos ESB-mallin liitetään useita eri järjestelmiä, kasvaa se helposti todella suureksi ja raskaaksi. (Flashnode, 2021; Goel, 2021)

Kuva 3. ESB-integraatiomalli. 1. Lähtö- ja kohdejärjestelmät. 2. Tiedonsiirtokerros. 3. Rajapinnat. 4. Tiedon käsittelyt ja muunnokset. 5. Prosessihallinta (Flashnode, 2021).



2.4 SOA, SOAP ja REST

SOA (Service Oriented Architecture) eli palvelukeskeinen malli ei ole varsinaisesti integraatioiden toteutustapa vaan suunnittelutapa, jossa jokainen erillinen liiketoimintaprosessi suunnitellaan toimimaan omana palvelunaan. Julkiset API:t voivat käyttää näitä palveluita suoraan, jolloin ei tarvita erillisiä järjestelmiä tätä varten. (Flashnode, 2021; Goel, 2021) SOA:n ongelmana on usein ollut pyrkimys ratkaista koko prosessikokonaisuus, joka tekee siitä raskaan. SOA:n raskaudesta johtuen on kehitetty mikro- ja minipalvelut, joissa kokonaisuus ajatellaan pienempiin osiin. Nämä osat ovat yleisesti liiketoimintafunktiokohtaisia eivätkä tarvitse toimiakseen muita järjestelmiä (mikropalvelut) tai niillä on vain vähän kytköksiä muihin järjestelmiin (minipalvelut). (Toivanen, n.d -a)

SOA tarvitsee toimiakseen jonkun web-palvelun. Näitä ovat muun muassa SOAP- ja REST-tekniikat. (Chakray, n.d) SOAP eli Simple Object Access Protocol on jo pitkään käytössä ollut standardi. REST (Representational State Transfer) on luotu yksinkertaisemmaksi vaihtoehdoksi SOAPille. (Smartbear, 2020) Toisin kuin SOAP, REST ei ole standardi, vaan arkkitehtuurinen tapa jakaa tietoa webissä (Pratt, 2021).

SOAPissa tietoa siirretään aina XML-muotoisena. Siirtoprotokollana käytetään useimmiten http:tä. (Korhonen, 2018; Smartbear, 2020) Myös muita protokollia voidaan käyttää. Esimerkiksi SMTP (Simple Mail Transfer Protocol) ja JMS (Java Message Service) toimivat SOAPissa. (Korhonen, 2018) SOAP standardissa tieto liikkuu XML-muodossa. Tiedon pyytävä muodostaa pyynnöstään XML-paketin, joka toimitetaan siirtoprotokollaa pitkin tiedon tarjoajalle. Tarjoaja purkaa XML:n ensin luettavaan muotoon ja muodostaa sen jälkeen vastauksen, joka kootaan taas XML:ksi ja lähetetään takaisin vastauksena saatuun pyyntöön. (Moore, 2001) Koska SOAP käyttää vain XML-paketteja, on se usein monimutkainen. SOAPin käyttöä voidaan helpottaa valitsemalla sellainen ohjelmointikieli, joka pystyy käsittelemään XML-kieltä. SOAPin parhaita puolia on sen ongelmankäsittely. Siihen on sisään rakennettu toiminto, joka palauttaa tiedon pyytäjälle ongelman ratkaisemiseen tarvittavat tiedot. (Smartbear, 2020)

REST on URL-pohjainen kevyempi versio SOAPista. Siinä missä SOAPin kutsuun voidaan vastata vain XML:llä, REST kutsu hyväksyy vastauksia myös CSV-, JSON- tai RSS-muodossa. (Smartbear, 2020) Myös HTML-esitysmuoto toimii RESTiä käytettäessä. RESTissä siis esitysmuotoa ei ole sidottu, kuten SOAPissa, mutta siirtotapana voidaan käyttää ainoastaan http:tä. SOAPissa siirtotapa oli vapaammin valittavissa. (Korhonen, 2018) RESTin parhaita puolia ovat läpinäkyvyys, luotettavuus ja skaalautuvuus (Pratt, 2021).

2.5 API-rajapinnat

Application Programming Interface (API) eli ohjelmointirajapinta on joukko toimintoja ja säädöksiä, joilla määritellään, miten tietoja tai palveluilta tarjotaan yhdestä ohjelmistosta muille sovelluksille tai järjestelmille (Siegel, 2020; avoinrajapinta.fi, 2014). Rajapinta voi olla avoin kaikille käytettäväksi tai se voi olla rajattu tilaajan hallitsemaksi. Avoimet rajapinnat ovat julkisia eikä niissä ole rajoituksia käytölle. Rajapinta, jossa järjestelmän tilaaja määrittelee oikeudet rajapinnan käytölle ja levitykselle, on tilaajan hallitsema rajapinta. Rajapinta voi olla datarajapinta tai toiminnallinen rajapinta. Datarajapinnassa siirretään vain tietoa järjestelmästä toiseen, kun taas toiminnallisella voidaan päivittää toisessa järjestelmässä olevia tietoja. (avoinrajapinta.fi, 2014) Rajapintoja on toteutettu aikaisemmin SOAPin avulla XML-rajapintoina, mutta helppokäyttöisempi ja kevyempi RESTin JSON-rajapinta on nykyään yleisempi (Siegel, 2020).

3 Liiketoimintaprosessien automaatio

Automaatio on joukko teknologioita, jotka tekevät automatisoituja prosesseja tai työvaiheita (Ventä, ym., 2018, s. 12). Automaatiota voidaan käyttää kaikkeen valmistuksesta ylläpitoon, suunnittelusta johtamiseen ja jopa myyntiin ja markkinointiin. Automaatiota ovat muun muassa robotiikka, telemetria ja tietoliikenne, langattomasti toimivat sovellukset, järjestelmäintegraatiot jne. (ISA, n.d; Kehal, 2019) Automaation etuja ovat kustannustehokkuus, laadun parantuminen, ajan säästö ja luotettavuus (Kehal, 2019).

Liiketoimintaprosessi on sarja toisiinsa liittyviä tehtäviä tai toimia, joiden tavoitteena on tuottaa jokin konkreettinen tulos. Liiketoimintaprosessi saa alkunsa tarpeesta ja päättyy tämän tarpeen tyydyttämiseen. Liiketoimintaprosessin eri vaiheita voi suorittaa joko ihminen tai jokin kone/järjestelmä. (Bourgeois & Bourgeois, 2014; Dvorak, 2020)

Liiketoimintaprosessien automaatio (Business Process Automation, BPA) tarkoittaa sitä, että liiketoimintaprosessin tehtäviä suoritetaan automaation avulla. Liiketoimintaprosessien automaatio ei ole järjestelmäsidoista, vaan siinä tarkistellaan kokonaisprosessia ja pyritään löytämään sieltä tehtävät, joiden automatisointi tuo suurimmat hyödyt. (Mustonen, n.d; Varon, 2021; McDermott, 2021) Liiketoimintaprosessien hallinta (Business Process Management, BPM) yhdistää automaation liiketoimintaprosessiin. Liiketoimintaprosessien hallinnassa tarkistellaan prosessia kokonaisuutena alkupisteestä loppupisteeseen. Prosessianalyysillä löydetään prosessista ne kohdat, joissa automaatiolla voidaan parantaa prosessia ja helpottaa ihmisen työtä. (Tirkkonen, 2019; Rock & Dwyer, n.d)

Liiketoimintaprosessien automaatio koskee koko organisaatiota alhaalta ylös asti. Alimmalla tasolla automaatiot suorittavat perustason tehtäviä, esimerkiksi raportointiin ja laskutukseen liittyen. Ylemmällä tasolla automaatio tukee yrityksen päätöksentekoa, tuottamalla esimerkiksi tietoa markkinoista ja kilpailijoista. (Ylén, ym., 2010, s. 44)

Liiketoimintaprosessien automaatiot voidaan jakaa neljään kategoriaan: 1)

Perusautomaatiot, 2) Prosessiautomaatiot, 3) Integraatioautomaatiot ja 4) Tekoälyyn

perustuvat automaatiot. Perusautomaatioilla tarkoitetaan yksinkertaisia tehtäviä, kuten

esimerkiksi sisäinen viestintä. Prosessiautomaatiot voivat olla esimerkiksi tiedostosiirtoja eri osastojen välillä. Integraatioautomaatioissa koneet ja järjestelmät suorittavat työtehtäviä

samalla tavalla kuin ihmiset. Tekoäly on näistä neljästä kaikista kehittynein ja sillä voidaan muun muassa varmistaa tuotannon sujuvuus. (Henderson, 2021)

Palkanlaskennassa ja integraatioissa käytettäviä automaation välineitä ovat RPA, tekoäly ja hyperautomaatio. Seuraavissa luvuissa tarkastellaan näitä tarkemmin.

3.1 RPA

Ohjelmistorobotiikka tarkoittaa ohjelmaa, joka käyttää yrityksen omia järjestelmiä erilaisten tehtävien suorittamiseen, aivan kuin ihminenkin tekisi. Ohjelmistorobotti suorittaa pääosin tehtäviä, jotka sijoittuvat yrityksen tukitoimiin tai ydintehtäviin. (Kääriäinen, ym., 2018, s. 8; Kaarlejärvi & Salminen, 2018, s. 53) Ohjelmistorobotti voi esimerkiksi syöttää dataa järjestelmään, kopioida dataa järjestelmästä toiseen tai suorittaa erilaisia laskentatehtäviä (kuva 4). Ominaista RPA:n suorittamille tehtäville on suuri toistettavuus. Niiden suorittamistavassa on vain vähän, jos ollenkaan poikkeamaa, eivätkä ne vaadi ihmisten päätöksentekokykyä. (Pratt, 2021; Kaarlejärvi & Salminen, 2018, ss. 53-54)

Ohjelmistorobotiikan suurimpia hyötyjä ovat laadun parantuminen, ihmistä nopeampi tehtävien suorittaminen, kustannustehokkuus ja pidemmät työtunnit. Robotti ei tarvitse vuorokautista lepoa kuten ihminen. (Pratt, 2021; Kaarlejärvi & Salminen, 2018, ss. 53-54)

Kuva 4. Ohjelmistorobotin käyttökohteita (Kääriäinen, ym., 2018).

| Työtehtävän kategoria | Selite |
|----------------------------------|---------------------------------------------------------------------------------------------------------------|
| Raportointi | Erialaisten raporttien ja yhteenvetojen kokoaminen järjestelmistä |
| Tarkistus ja testaus | Tietojen oikeellisuuden tarkistaminen, järjestelmätestaukset |
| Tiedon esikäsittely | Kerätään, työstetään ja lajitellaan tietoa myöhempää käyttöä varten ihmiselle prosessin seuraavissa vaiheissa |
| Tiedon päivittäminen | Ylikirjoitetaan vanhaa tietoa uudella tai poistetaan vanhaa tietoa kokonaan. Tiedon laadun ylläpitäminen |
| Tiedon siirtäminen | Siirretään tai kopioidaan tietueita järjestelmästä toiseen, massatallennukset, arkistoinnit |
| Tiedon syöttäminen järjestelmään | Syötetään uusia tietueita järjestelmiin, esimerkiksi luodaan uusi asiakkuus tai työntekijä |
| Tiedon täsmäyttäminen | Verrataan kahden tai useamman tietolähteen tietoja keskenään |
| Viestin lähetyks | Massapostitukset, sähköpostien lähetyks, muistutukset, selvityspyynnöt |

3.2 Älykäs automaatio

Kun yhdistetään IT-automaation välineitä, jotka hyödyntävät koneoppimista ja tekoälyä, saadaan älykästä automaatiota (Intelligent Automation, IA). Koneoppimisen avulla tekoälyä käyttävät laitteet tai ohjelmat voivat tehdä itsenäisiä päätöksiä ja toimia. (McHugh, n.d)

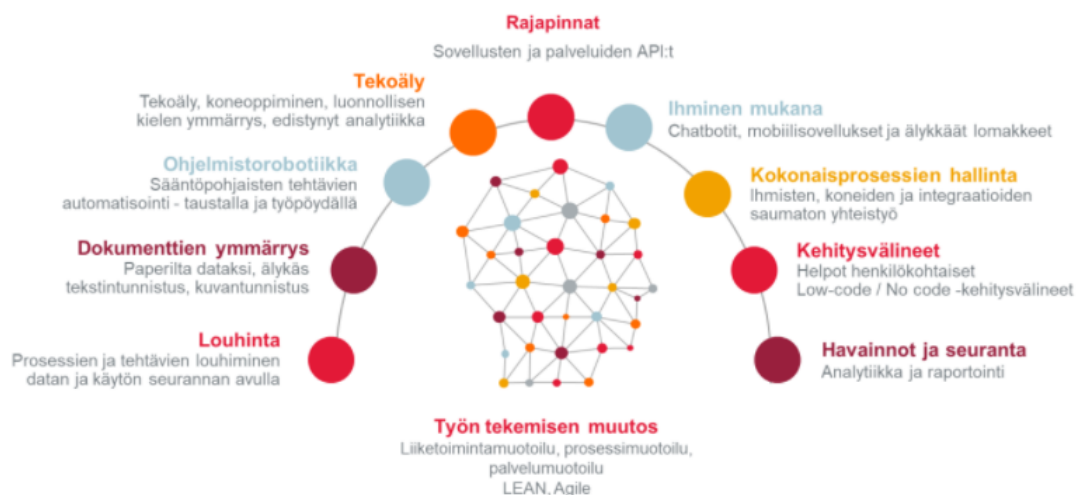
Koneoppiminen (Machine Learning, ML) tarkoittaa sitä, että laite tai järjestelmä ei tarvitse ihmistä oppiakseen uutta. Koneoppiminen perustuu laajoihin datamassoihin, algoritmiin ja laajaan lähdekoodiin. Näiden avulla kone pystyy keräämään dataa ja tekemään päätelmiä, miten sen kuuluu milloinkin toimia. (Frankenfield, 2020; Kaarlejärvi & Salminen, 2018, ss. 59-60)

Tekoäly eli Artificial Intelligence (AI) tarkoittaa teknologiaa, joka sisältää oppimista. Tekoälyn avulla koneet, laitteet, ohjelmat ja järjestelmät osaavat toimia tilanteen vaatimalla tavalla. (Kääriäinen, ym., 2018) Tekoälyä hyödyntävät laitteet pystyvät ihmisen kaltaiseen ajatteluun ja saavuttavat tällä tavoin niille asetetut tavoitteet. Vain yksinkertaisiin yhden aktiviteetin tehtäviin pystyvää tekoälyä sanotaan heikoksi tekoälyksi. Vahvaa puolestaan on sellainen, joka pystyy suorittamaan tehtäviä, jotka ovat vaativampia ja edellyttävät ihmismäistä toimintaa. (Frankenfield, 2021)

3.3 Hyperautomaatio

Kun yhdistetään järjestelmäintegraatiot, RPA, tekoäly ja työkulkujen automaatio, saadaan hyperautomaatio. Hyperautomaatiota on kutsuttu myös älykkääksi prosessiautomaatioksi (Intelligent Process Automation, IPA) tai digitaaliseksi prosessiautomaatioksi (Digital Process Automation, DPA). (Toivanen, n.d -b; Eisner, 2020) Hyperautomaatio koostuu rajapinnoista, tekoälystä, ohjelmistorobotiikasta, datan käsittelystä (ymmärrys ja louhinta), prosessien hallinnasta, kehitysvälineistä, seurannasta ja ihmisistä (kuva 5). Voidaankin puhua kokonaisprosessien automatisoinnista. (Pursiainen, 2020) Koska hyperautomaatio käyttää useaa erilaista teknologiaa hyväkseen, poistaa se rajoitteita, joita esimerkiksi ohjelmistorobotiikan käytössä on ollut (Eisner, 2020).

Kuva 5. Hyperautomaation osa-alueet (Pursiainen, 2020).



4 Palkanlaskennan kokonaisprosessi

Usein palkkahallinto sisällytetään osaksi talous- tai henkilöstöhallintoa. Palkkahallinto on merkittävä osa yrityksen raportointia, kirjanpitoa ja reskontraa. Järjestelmänäkökulmasta tarkisteltaessa, palkanlaskentaa voidaan suorittaa joko kirjanpito-ohjelmien sisällä oleville palkkahallinto-osioilla tai erillisillä palkkahallinto-ohjelmilla. Mikäli palkkahallintoa suoritetaan erillisellä ohjelmalla, sisältää se yleensä integroinnit muihin tarvittaviin järjestelmiin. Palkkaprosessi sisältää useita sellaisia työvaiheita, joiden siirtäminen manuaalisesta tekemisestä automaattiseksi ja digitaaliseksi vähentää näihin toimiin tarvittavan työajan määrää. (Lahti & Salminen, 2014, s. 135-136)

Palkkaprosessi koostuu käytettävästä järjestelmästä riippumatta hyvin pitkälle samoista elementeistä. Palkanlaskentaprosessi voidaan jakaa palkkakaussittain, kuukausittain ja vuosittain tapahtuviin tehtäviin. Palkkakaussittain ja kuukausittain tehtäviä tehtäviä ovat palkanlaskenta, palkkakirjanpito ja viranomaisraportointiin liittyvät toimenpiteet. Vuosittaiset tehtävät liittyvät järjestelmän ylläpitoon, esimerkiksi palkasta pidettävien maksujen prosenttien ja verottomien korvausten sekä luontoisetujen hintojen päivittäminen sekä vuosittain tapahtuvaan raportointiin. (Mattila, Orlando & Parninen, 2020, s. 306)

Palkanlaskennan kokonaisprosessin tehtävistä osa saattaa kuulua henkilöstöhallinnolle ja osa palkkahallinnolle. Henkilöstöhallinnon tehtäviin kuuluu rekrytointi, työntekijätietojen ylläpito ja työajan seuranta. Palkkahallinnon vastuulla on noudattaa lakeja, asetuksia ja sopimuksia sekä maksaa palkat. (Kouhia-Kuusisto, Mikkonen, Syvänperä, Turunen, 2017, s. 10-11)

Seuraavissa luvuissa esitellään palkanlaskennan kokonaisprosessin kannalta olennaisia tehtäviä HR- ja palkkahallinnon kannalta.

4.1 Henkilöstöhallinnon tehtävät

Lahti ja Salminen (2014, s. 137) listaavat henkilöstöhallinnolle kuuluviin tehtäviin henkilöstöstrategia ja -politiikka, rekrytointi, työterveyshuolto ja hyvinvointi, osaamisen johtaminen sekä koulutusten suunnittelu ja toteuttaminen, resurssi- ja lomasuunnittelu, työehtosopimusasiat, työsuojeluasiat, eläke- ja sairausvakuutusasiat, palkitseminen,

henkilöstötietojen ylläpito sekä sidosryhmäsuhteet ja ostopalvelusopimukset. Kouhia-Kuusisto, Mikkonen, Syvänperä ja Turunen (2017, s. 10) lisäävät edelliseen vielä työaikojen seurannan ja yhteistoimintamenettelyistä vastaamisen.

4.2 Palkkahallinnon ja palkanlaskennan tehtävät

Palkkahallinnon tehtävälistaan kuuluvat palkkojen määrittäminen ja sopimusten tulkinta, palkkakustannusten seuranta, esimiesten avustaminen, arkistointi, työtodistusten ja hakemusten laatiminen, tilastointi ja muu raportointi sekä henkilöstötietojen ylläpito (Lahti & Salminen, 2014, s. 137). Palkkahallinnon on taattava, että palkanlaskennassa noudatetaan lakeja, asetuksia ja sopimuksia sekä, että palkkojen maksu tapahtuu oikein ja ajallaan (Kouhia-Kuusisto, Mikkonen, Syvänperä, Turunen, 2017, s. 10-11).

Palkanlaskenta on osa palkkahallintoa ja henkilöstöhallintoa. Sen tehtäviin kuuluu palkkojen laskenta teknisesti oikein. Palkanlaskennassa on oltava tuntemus laeista, säädöksistä, palkanlaskentaohjelmista ja näihin tapahtuvista muutoksista. ((Kouhia-Kuusisto, Mikkonen, Syvänperä, Turunen, 2017, s. 11) Palkanlaskenta hoitaa yrityksen palkkojen laskennan ja maksun, verojen ennakoperinnän, palkkakirjanpidon, viranomaisraportoinnin ja tilitykset, työaika- ja muiden palkkatapahtumien keruun, työaikatapahtumien tulkinnan, työntekijätietojen ylläpidon sekä arkistoinnin. (Lahti & Salminen, 2014, s. 137)

4.3 Palkanlaskennan prosessi

Palkkojen maksua edeltää se, että palkanlaskenta suorittaa neljä sille kuuluvaa ydintehtävää. Nämä neljä tehtävää ovat aineiston kerääminen (palkka- ja työaika-aineisto), kerätyn aineiston tulkinta, varsinainen palkkojen laskenta ja palkkakirjanpidon muodostus sekä raportointi. (Lahti & Salminen, 2014, s. 138)

Aineisto kerääminen pitää sisällään muun muassa työsuoritusten ja loma- ja poissaolotapahtumien kirjaamista palkkajärjestelmään (Lahti & Salminen, 2014, s. 138–139). Työsuoritukset voidaan toimittaa palkanlaskentaan esimerkiksi työaikakorteilla ja ne sisältävät muun muassa ylityötuntien määrän (Ferguson, 2019). Aineisto voidaan toimittaa

muodossa, jossa palkanlaskennan tehtäväksi jää työajan laskenta alku- ja loppuajan perusteella, lounastauko huomioiden (Beasley, 2019).

Työaikatapahtumien tulkinta tarkoittaa saadun aineiston muokkaamista muotoon, jotta varsinainen palkanlaskenta voidaan suorittaa. Tulkinnan tuloksena palkkajärjestelmässä on jokaisella palkansaajalla kerätyn aineiston mukaiset tapahtumat oikeilla palkkalajeilla lakien ja säännösten mukaisesti. (Lahti & Salminen, 2014, s. 139)

Palkanlaskenta vaiheessa lasketaan jokaiselle palkansaajalle maksettava nettopalkka, ennakonpidätyksen määrä ja muut vähennykset (Lahti & Salminen, 2014, s. 140). Raporttien perusteella palkat voidaan tarkistaa tässä vaiheessa ennen siirtymistä palkkojen maksatukseen (Ferguson, 2019). Tarkistukset voidaan myös tehdä palkkajärjestelmän käyttöliittymässä (Lahti & Salminen, 2014, s. 140).

Raportointi eroaa sen mukaan, onko se palkkakaudesta, kuukaudesta tai vuosittain raportointia (Lahti & Salminen, 2014, s. 140). 1.1.2019 käyttöön otettu tulorekisteri on muuttanut palkkojen raportointia merkittävästi. Palkanmaksaja ei enää tee jokaiselle viranomaiselle erillistä ilmoitusta maksetuista palkoista, vaan tietojen käyttäjät hakevat ne itse tulorekisteristä. Erillisiä vuosi-ilmoituksia ei enää tehdä, vaan palkkatiedot ilmoitetaan jokaisesta palkanmaksusta erikseen. Ilmoituksen määräpäivä on viisi päivää maksupäivästä. Työnantajan erillisilmoitus on annettava palkanmaksua seuraavan kuukauden 5. päivänä. (Verohallinto, 2021)

Palkka-ajo-, kuukausi- tai vuosiraportointia saattavat tarvita myös organisaation sisäiset sidosryhmät, kuten HR- tai taloushallinto (Ferguson, 2019). Maksutiedoston lähettäminen pankkiin ja palkkalaskelmien lähettäminen palkansaajille kuuluvat myös palkkakaustain tehtävään raportointiin (Lahti & Salminen, 2014, s. 140-141).

5 Integraatiot ja automaatio palkkaprozessissa

Palkanlaskentaprosessi sisältää huomattavan määrän erilaisia työtehtäviä ja sen automatisoinnilla saavutetaan merkittäviä säästöjä esimerkiksi työhön käytettävässä ajassa (Lahti & Salminen, 2014, ss. 135-136). Automatisoimaton palkanlaskenta on hidasta ja huomattavan paljon alttiimpaa virheille, kuin automatisoitu palkanlaskenta (Ferguson, n.d - a). Kehittyvät teknologiat palkanlaskentajärjestelmissä, robotiikassa ja tekoälyssä, ovat mahdollistaneet automaation kasvattamisen palkkaprozessissa (Ruokonen, 2021).

Palkanlaskennan automaatio lähtee liikkeelle prosessin kehittämisestä. Prosessia pitää järkevöittää ja selkeyttää, jotta automaatiosta saadaan paras hyöty irti. (Ruokonen, 2021) Palkanlaskentaprosessia automatisoitaessa, on otettava huomioon kaikki sen eri vaiheet aina työntekijöiden ja esimiesten toimenpiteistä palkanlaskennan toimiin sekä talouden raportoinnista viranomaisraportointiin saakka (Lahti & Salminen, 2014, s. 138). Palkanlaskentaprosessin automatisoinnin hyötyjä ovat tehokkuuden lisääntyminen, alhaisemmat kustannukset, virheiden vähentyminen, virheistä johtuvan selvitystyön vähentyminen ja nopeampi prosessi (Fiedler, n.d; White, 2020; Cronquist, 2020).

5.1 Integraatiot ja rajapinnat palkkaprozessin automaatiassa

Palkkajärjestelmään tarvitsee monesti tuoda tietoa ulkoisista järjestelmistä, jotta työntekijöille voidaan maksaa palkat oikein ja ajallaan. Vastaavasti palkkajärjestelmä tuottaa usein sellaista tietoa mitä tarvitaan muissa järjestelmissä. Nämä liitokset eri järjestelmien välillä voidaan hoitaa integraatioilla. Palkkajärjestelmä liitetään usein integraatiolla HR-, työajanseuranta- tai kirjanpitojärjestelmiin. (Beasley, 2020) Integraatiot eri järjestelmien välillä hoidetaan taustaprosesseina, jolloin manuaalista työtä ei tarvita, vaan ohjelmistot kommunikoivat keskenään ja kulloinkin tarvittava tieto liikkuu järjestelmien välillä (Ruokonen, 2021).

Järjestelmäintegraatioita voidaan tehdä joko tiedostopohjaisella siirrolla tai API-rajapinnan avulla. Tiedostopohjaisessa siirrossa lähtöjärjestelmä muodostaa tiedoston, joka siirretään kohdejärjestelmään. Esimerkiksi HR-dataa siirrettäessä, voidaan siirtää jokaisella siirrolla koko henkilöstödata tai vain muuttuneet tiedot. API-rajapinta on tiedostopohjaista

nopeampi tapa tehdä integraatioita, koska data päivittyy kohdejärjestelmään reaaliaikaisesti. (Xabeire, n.d)

5.1.1 HR-järjestelmäintegraatio

HR- ja palkkajärjestelmissä tarvitaan usein samaa tietoa. Integroimalla HR- ja palkkajärjestelmä toisiinsa, vältetään manuaaliselta tietojen tallentamiselta kahteen eri järjestelmään, vähennetään erillisten paperisten dokumenttien tarvetta, mahdollistetaan automaattinen datan päivitys ja pystytään tuottamaan laadukkaampia raportteja. (Silven-Hoell, n.d) Integroiduissa järjestelmissä saadaan kaikki työsuhteen elinkaareen liittyvä HR-data liikkumaan HR- ja palkkajärjestelmän välillä. Kun uusi työntekijä palkataan, siirtyvät hänen tietonsa palkanlaskentaan. Kun työsuhteessa tapahtuu muutoksia (esimerkiksi palkan muutos tai organisaatiomuutos), päivittyvät ne myös palkkajärjestelmään. Integraatiossa voi olla mukana myös erillinen rekrytointijärjestelmä, jolloin uusien henkilöiden tiedot siirtyvät sieltä tarvittaviin järjestelmiin. Osa tiedoista, kuten esimerkiksi verokortti ei useinkaan löydy rekrytointi- tai HR-järjestelmistä. (Lahti & Salminen, 2014, ss. 142-143) Palkan maksaja voi verottajan tarjoamaa VeroAPIa pitkin hakea henkilön verotustiedot suoraan verottajalta (Verohallinto, 2020).

5.1.2 Integraatiot työajanseurantajärjestelmistä

Työaikatietojen (työtunnit, lomat ja poissaolot) saaminen palkkajärjestelmään laskentaa varten on yksi palkkaprosessin vaikeimmista ja tärkeimmistä tehtävistä. Jotta tiedot voidaan siirtää palkanlaskentaan, tulee esimiesten hyväksyä ne työntekijöiden tuntikirjausten jälkeen. Tässä tyypillisesti sähköisessä vaiheessa voidaan vielä tehdä tarpeen mukaan korjauksia aineistoon. (Lahti & Salminen, 2014, ss. 138-139) Työaikojen kirjaamista varten on tarjolla erilaisia sovelluksia, jotka toimivat eri alustoilla (esim. web- tai mobiilisovellus) (Kosinski, 2021). Lisäksi löytyy työpaikolle asennettavia järjestelmiä, jotka toimivat esimerkiksi sormenjälkitunnistuksella tai erillisellä tagilla (Ferguson, n.d -b). Tuntikirjauksia on mahdollista myös tehdä suoraan joihinkin palkkajärjestelmiin (Ruokonen, 2021).

Osa työaikatietojen keräämistä on niiden tulkinta. Tulkintavaiheessa työntekijöiden tekemistä työaikakirjauksista muokataan palkkadataa. Työtunneista tulkitaan ylityöt ja tapahtumat käännetään palkkalajeiksi. Tämä toimenpide voidaan tehdä joko työajanseuranta- tai palkkajärjestelmässä. (Lahti & Salminen, 2014, ss. 139, 147)

5.1.3 Palkkojen raportointiin liittyvät integraatiot

Palkanlaskennasta toimitetaan runsaasti erilaista raportointia muun muassa palkansaajalle, viranomaisille ja yrityksen sisäisille toiminnolle. Kun palkanlaskenta valmistuu, lähetetään palkanlaskentajärjestelmästä maksutiedosto maksuliikennettä hoitavaan järjestelmään, josta se edelleen siirtyy pankkiin. Palkansaajalle lähetetään jokaisesta palkanmaksusta palkkalaskelma. Palkkalaskelmien yleisimmät toimitustavat ovat verkkopalkkalaskelma verkkopankkiin tai Postin iPost-palvelun kautta joko paperisena tai eKirjeenä toimitettava laskelma. (Lahti & Salminen, 2014, ss. 140-141)

Palkanlaskennan valmistuttua, yrityksen sisäistä raportointia varten palkkajärjestelmästä tuotetaan raportointia kirjanpitoon ja mahdollisesti myös muihin raportointijärjestelmiin. Kirjanpitoon toimitetaan tiedot palkkakustannuksista, näistä lasketut sivukulut sekä varaukset (esimerkiksi lomapalkkavarauus). Palkoista tehdään palkkakaussittaista ja kuukausittaista raportointia verottajalle, eläkeyhtiöille, vakuutusyhtiöille, Kelaan ja ay-liitoille. (Lahti & Salminen, 2014, ss. 140-142) Viranomaisraportointi tulorekisteriin palkoista ja ansiotuloista on mahdollista hoitaa rajapinnan kautta tai tulorekisterin asiointipalvelussa (Verohallinto, 2021).

5.2 RPA ja tekoäly palkkaproessin automaatiassa

Palkkaprosessista löytyy paljon tehtäviä, jotka ovat hyviä kohteita ohjelmistorobotiikalle, niiden luonteen vuoksi. Tällä tavoin voidaan palkkaprosessissa lisätä tehokkuutta ja vapauttaa ihmisten aikaa muihin tehtäviin. (Pal, 2021) Nämä toistuvat ja rutiiniomaiset tehtävät toistuvat palkanlaskennan arjessa päivittäin, viikoittain tai kuukausittain. Tällaisia tehtäviä ovat erilaiset täsmäytykset (kirjanpito), tarkistukset (poissaolot, henkilötiedot, maksatus), ilmoitukset (verottajalle, sidosryhmille) tai tietojen siirtäminen järjestelmästä

toiseen, silloin kuin integraatio ei ole mahdollista. (Honkanen, n.d; Kaarlejärvi & Salminen, 2018, s. 53)

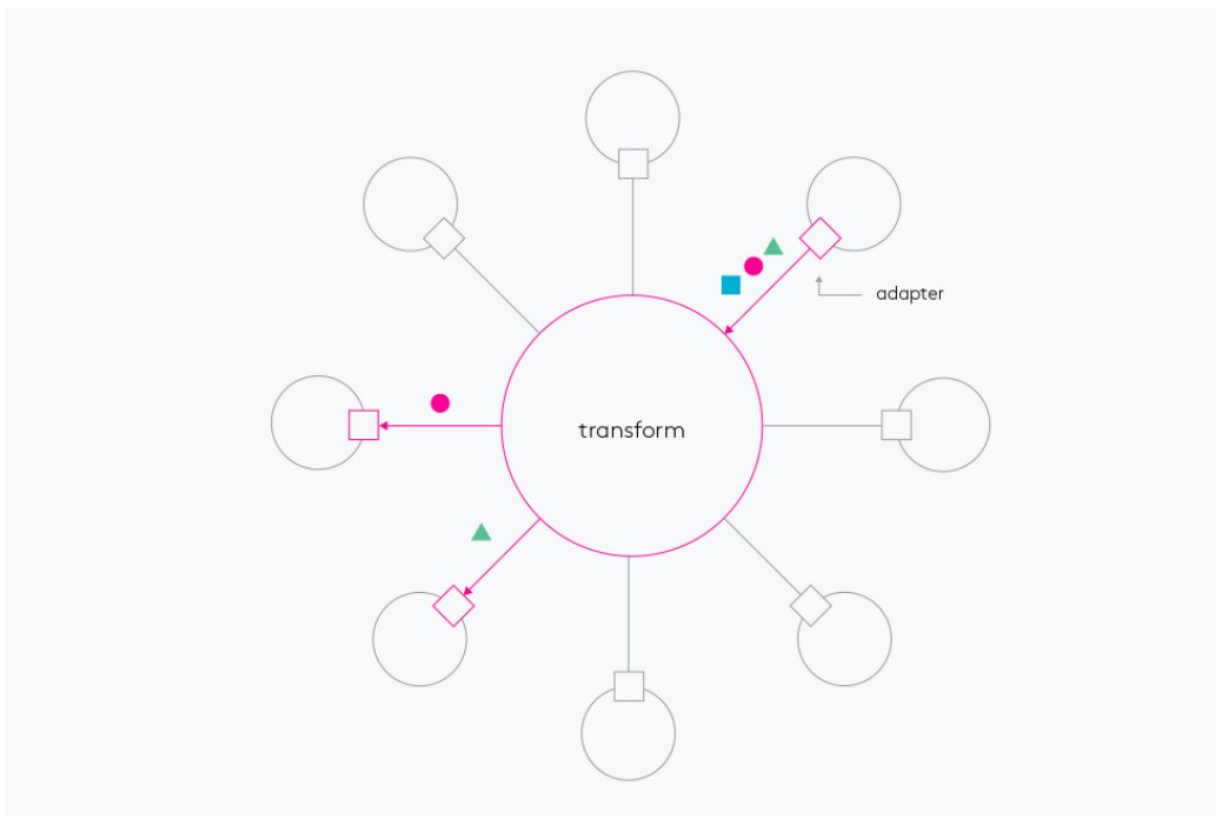
Palkanlaskennan kokonaisprosessia tarkisteltaessa, ohjelmistorobotiikasta saadaan apua koko työsuhteen elinkaaren varrelle aina rekrytoinnista työsuhteen päättymiseen saakka. Ohjelmistorobotiikkaa hyödyntämällä voi HR rekrytointivaiheessa kartoittaa sopivia kandidaatteja laittamalla botti tutkimaan työnhakijoiden CV:tä. Työntekijän sisäänotossa RPA:n avulla voidaan tilata työntekijälle käyttäjätunnukset, laitteet, sähköpostiosoitteet ja muu tarvittava työvälineistö. Lisäksi ohjelmistorobotiikka voidaan käyttää muun muassa palkka-, henkilö- ja työsuhtetietojen ylläpidossa. Palkanlaskennan prosesseissa RPA voi esimerkiksi tarkistaa henkilöiden poissaolotietoja, verrata tuntipalkkaisen henkilön suunniteltuja työaikoja toteutuneisiin, avustaa kuukausittaisissa tarkistuksissa ja täsmäytyksissä sekä ylläpitää tietoja palkkajärjestelmässä. Työsuhteen päättyessä ohjelmistorobotti voi laskea lopputilin maksuun tarvittavia tietoja sekä sulkea työntekijän tunnukset ynnä muita sen kaltaisia tehtäviä. (Pal, 2021; Makadia, 2020)

Tekoälyn ja koneoppimisen avulla voidaan palkanlaskennan automaatiota viedä vielä askel pidemmälle. Osa palkanlaskennan ja HR:n työtä on vastata erilaisiin kyselyihin, joita tulee työntekijöiltä ja esimiehiltä. Näihin voidaan avuksi ottaa tekoälyn ohjaama chatbotti, joka osaa jo itsellään vastata osaan kysymyksistä tai mahdollisesti ohjata kysyjän suoraan oikean henkilön puheille, jolloin yhteistyö palkka- ja henkilöstöhallinnon sekä työntekijöiden välillä paranee. Tekoälyn avulla voidaan paremmin ehkäistä palkoissa esiintyviä virheitä. Esimerkiksi työntekijöiden tuntikirjauksia voidaan tarkistaa ja mahdollisesti jopa korjata tekoälyn avulla. Tekoäly voi myös tarkistaa palkanlaskennan tuloksia ja etsiä sieltä poikkeamia vertaamalla nykyistä dataa historiaan. Tekoälyllä voidaan myös tehdä henkilöstöhallinnolle ennusteita tulevasta resurssitarpeesta ja näin yritys voi paremmin varautua tulevaan. Rekrytointivaiheessa tekoäly pystyy käsittelemään hakemuksia ja valikoimaan joukosta parhaat kandidaatit avoimna olevaan tehtävään tai jopa ehdottaa työntekijälle palkankorotusta tehokkuuteen perustuen. Tekoälyllä voidaan myös tarkistella käytössä olevaa prosessia ja saada sieltä mahdolliset pullonkaulat sekä muut prosessia hidastavat osa-alueet esiin reaaliaikaisesti. (Kugasia, 2019; Sen, 2021; Radin, 2017; Silva, 2020)

6 Integraatioalustat ja toimitusmuodot

Integraatioalusta on ohjelma, jolla toteutetaan integraatioita. Integraatioalustaan voidaan kytkeä useita eri ohjelmia, jotka on rakennettu erilaisilla teknologioilla. Integraatioalustan avulla eri ohjelmat kytkeytyvät saumattomasti toisiinsa ja niiden liikennettä voidaan sekä valvoa että hallita. Integraatioalustalla voidaan yhdestä järjestelmästä tuleva data muuntaa kohdejärjestelmälle sopivaan muotoon (kuva 6). Samoin voidaan tehdä muunnoksia protokollaan, jolla dataa liikutellaan. Esimerkiksi lähtöjärjestelmästä tuodaan tietoa XML-muodossa SFTP-protokollaa käyttäen integraatioalustalle, jossa se muunnetaan JSON-muotoon ja lähetetään REST-kutsuna eteenpäin. Integraatioalustat voivat kytkeytyä niin pilvessä kuin on-prem asennuksina toimiviin järjestelmiin. (Toivanen, n.d -b; Shankar, 2020; Toivanen, 2019)

Kuva 6. Integraatioalustan toiminnan mallikuva (Toivanen, HiQ, n.d -b).



Integraatioalusta voivat toimia myös API-rajapintojen suoritusalustoina. Integraatioalusta voi esimerkiksi konnektoreiden ja adaptereiden avulla hakea konesalissa olevassa järjestelmästä tietoa ja viedä sitä pilvessä sijaitsevaan järjestelmään. Integraatioalustoilla voidaan toteuttaa myös prosessiautomaatiota. Esimerkiksi RPA:n suorittaessa automaatiota käyttöliittymän

takana, integraatioalusta voi suorittaa prosessia rajapintojen päällä. RPA voidaan liittää osaksi integraation työkulkua, jolloin saadaan automatisoitu liiketoimintaprosessi. Integraatioalustaa ei voida kuitenkaan käyttää prosessiautomaatiossa, jos rajapinta puuttuu tai automaatio suoritetaan vain yhden käyttöliittymän sisällä. (Toivanen, n.d -b; Toivanen, 2019)

Integraatioalustoissa ja niiden ominaisuuksissa on hyvin paljon vaihtelua. Full-code-alustat vaativat käyttäjältä ohjelmointiosaamista. Low-code-alustat vaativat vähän protokolla- ja standardiosaamista. No-code-alustat ovat täysin valmiita ja ne osaavat suoraan sovelluksen rajapinnat. Full-code-alustojen käyttö vaatii koodaajia, jotka tuntevat käytössä olevan alustan, jonka vuoksi niiden haittana on muun muassa ongelmatilanteiden ratkominen. Koodia osaamaton henkilö ei pysty itse ongelmaa ratkaisemaan, vaan siihen tarvitaan käytännössä aina toimittajan ohjelmoija. No-code-alustat toimivat valmiiden adaptereiden pohjalta. Jos integraatioon kytkettävissä ohjelmissa on tehty määrittämiä tai asiakaskohtaisia muokkauksia siten, että valmisadapteri ei toimi suoraan, tarvitaan integraatioalustaan muutoksia. (Toivanen, n.d -b) Low-code käyttää sekä low-code- että no-coden-tekniikoita. Se sisältää malliin pohjautuvaa suunnittelua ja kehitystä. Low-code-alustat ovat helpompia ottaa käyttöön ja ylläpitää kuin full-code tai no-code. (Toivanen, n.d -b; Gartner, n.d -a)

6.1 Integraatioalustojen toimitusmuodot

Kun otetaan käyttöön integraatioalusta, voidaan se joko rakentaa itse tai ostaa palveluna. Jos integraatioalusta rakennetaan itse, jää kaikki ylläpito ja kehitysvastuu yritykselle itselleen. Kun se ostetaan palveluna, saadaan jatkuvasti kehittyvä ja varmatoiminen integraatio. (Tähtinen, 2016) Palveluna ostettavista integraatioalustoista puhutaan usein iPaaS-mallina eli Integration Platform as a Service -mallina. iPaaS pohjautuu Paas-malliin, jossa perinteisen lisenssien ostamisen (SaaS eli Software as a Service) sijasta, ostetaan käyttöoikeus ohjelmaan. iPaaS-mallissa siis vuokrataan ohjelma pilvestä omaan käyttöön. Näissäkin alustoissa on eroja muun muassa skaalautuvuudessa. Osa perustuu perinteisempään palvelinmalliin ja osa serverless eli palvelimettomaan malliin. Näiden erona on se, että kun halutaan lisätä kapasiteettia ja tehoja, täytyy palvelinmalliin lisätä tehoja perinteiseen tapaan palvelintehoa kasvattamalla. Serverless-palveluissa tehot skaalautuvat

integraation vaatimusten mukaan, jolloin käytössä on aina oikea määrä kapasiteettia. (Ryhänen, 2019)

EiPaaS (Enterprise Integration Platform as a Service) on iPaaS-mallin alalaji. iPaaS muuttuu EiPaaSiksi silloin, kun integraatiopalvelu on suunniteltu palvelemaan enterprise-tason yrityksiä. Tällöin tarvitaan korkeampaa saavutettavuutta ja virreehallintakykyä, turvallisuutta, palvelutasosopimuksia sekä teknistä tukea kuin iPaaS-mallin integraatioissa. (Gartner, n.d -b) EiPaaS tarjoaa yrityksille laajemmat mahdollisuudet räätälöidä omia integraatiotaan esimerkiksi data formaattien ja käytettävien protokollien osalta. EiPaaS kuitenkin vaatii syvällisempää teknistä osaamista ja suurempia investointeja kuin iPaaS käyttö. (Ferguson, 2019)

6.2 Integraatioalustojen tietoturva

Perinteisissä point-to-point-integraatioissa tietoturva nojaa integraation piirissä oleviin järjestelmiin sekä tiedonsiirtotapaan. Haavoittuvuusriskit kasvavat, mikäli järjestelmien tietoturvassa on puutteita tai tiedonsiirto tapahtuu suojaamattomalla yhteydellä.

Integraatioalustan käyttö itsessään tuo lisäturvaa integraatioihin. (Keskiruokonen, 2020)

Kun integraatio hoidetaan integraatioalustalla, saadaan lisäturvaa alustan omista tietoturva-asetuksista ja käyttäjähallinnasta. Integraatioalustalle määritellään omat tietoturvaan liittyvät asetukset samoin tavoin kuin lähtö- ja kohdejärjestelmiin. Alustalle annetaan käyttöoikeudet jokaiselle käyttäjälle hänen roolinsa mukaisten tarpeiden perusteella. Siirrettävää dataa voidaan myös hallita keskitetymin integraatioalustalla. Dataa voidaan validoida ja tätä kautta tarkkailla sen laatua. Alustan virhetilanteiden hallinnalla voidaan nopeasti reagoida havaittuihin ongelmiin. (Keskiruokonen, 2020)

Integraatioalusta lisää tietoturvaa integraatioissa viidellä eri tavalla:

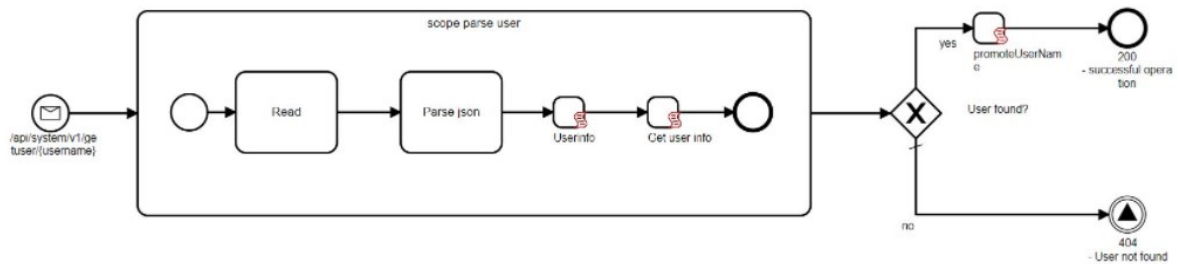
1. Keskitetty yhteyksien hallinta yhdestä pisteestä – kaikki integraatiot kulkevat yhden pisteen kautta.
2. Datat hallinta – mitä dataa integraatioissa liikkuu.
3. Tietoturvallisten siirtoprotokollien käyttö.

4. Haavoittuvuuksien nopeampi neutralisointi – tietoturvuhan kohdalla yhteydet saadaan katkaistua yhdestä työkalusta ja palautettua uhan neutraloimisen jälkeen nopeasti takaisin.
5. Käyttäjätunnusten ja salasanojen hallinta – mm. salasanojen keskitetty päivitys vain yhteen järjestelmään. (Choudhuri, 2019)

7 Friends-integraatioalusta

Friends on nykyaikainen hybridi-integraatioalusta, jolla voidaan toteuttaa integraatioita tiedostosiirroista hyperautomaation asti. Lisäksi Friendsillä onnistuu API-rajapintojen toteutus ja hallinta. Friends on EiPaaS-mallilla toimitettava low-code-ratkaisu integraatioiden tekemiseen, jossa integraatiota rakennetaan BPM-notaatiolla. BPMN eli Business Process Model and Notation on liiketoimintaprosessien kuvaamiseen käytetty visuaalinen kieli. Friendsissä rakennetaan vuokaaviomaisia työkulkuja (kuva 7) valmiiden komponenttien (Friends-taskit) avulla. (HiQ, n.d -a; HiQ, n.d -b; Friends, n.d -a)

Kuva 7. Esimerkki Friends-prosessin vuokaaviosta (HiQ, n.d -b).



Friends-integraatioiden suorittamisen hoitaa agentti, joka keskustelee käyttöliittymän kanssa ja suorittaa sille määritellyjä integraatioita. Agentti itsessään ei sisällä mitään älyä, vaan niiden ohjaaminen tapahtuu Friendsin Control Panesta eli kontrollikeskuksesta. Samaisessa keskuksesta tehdään myös kehitys, monitorointi ja hallinta. Agenteja voidaan asentaa omaan konesaliin, pilveen tai kontitusympäristöön. (Friends, n.d -a; HiQ, n.d -b)

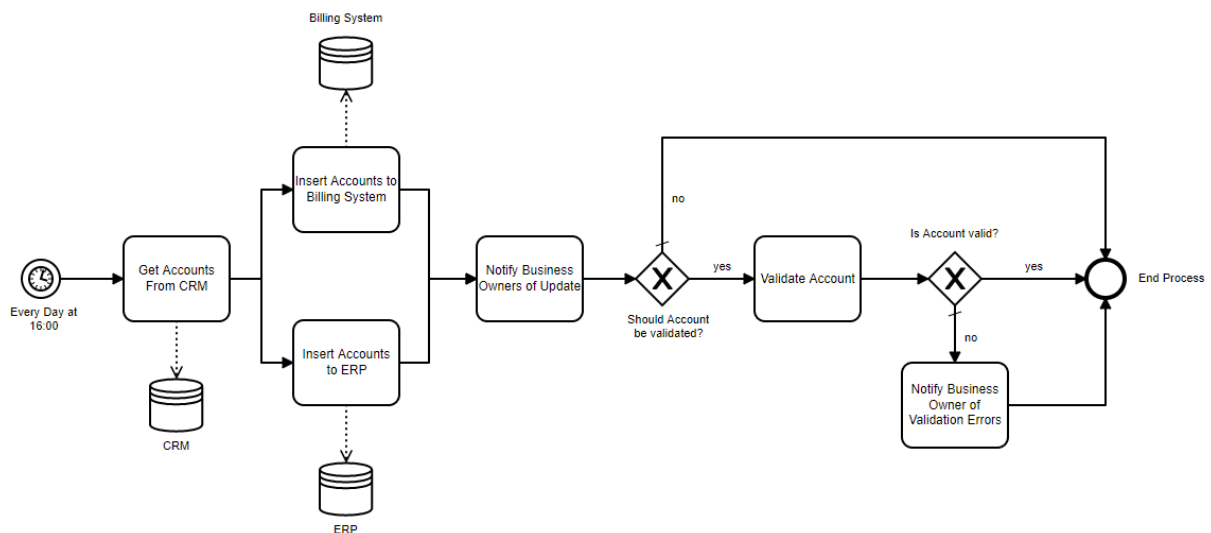
Friends sisältää oman versionhallinnan, deployment frameworkin ja monitoroinnin. Monitoroinnin avulla virheiden selvittäminen ja häiriöiden hallinta helpottuu. Lisäksi monitoroinnin avulla voidaan tutkia ajohistoriaa. Monitorointi on toteutettu visuaalisesti kontrollikeskuksesta ja prosessien ajojen onnistumisia on mahdollista seurata liikennevaloilla. Jokainen prosessin task eli tehtävä voidaan tarkistaa monitoroinnissa omanaan, jolloin nähdään vihreällä mitkä ovat onnistuneet ja punaisella ne mitkä epäonnistuneet. (HiQ, n.d -b)

7.1 Friends-integraatiot

Friends-integraatioalustalla voidaan toteuttaa prosessiautomaatioita, sovellusintegrointia, tiedostosiirtoja, datamuunnoksia ja API-rajapintoja. Prosessiautomaatioiden suunnittelu Friendsillä on luontaista johtuen BPMN 2.0 -standardin käytöstä. Prosessiautomaatiot voivat sisältää päätöksiä, liiketoimintalogiikkaa, silmukoita ja virheenkäsittelyä.

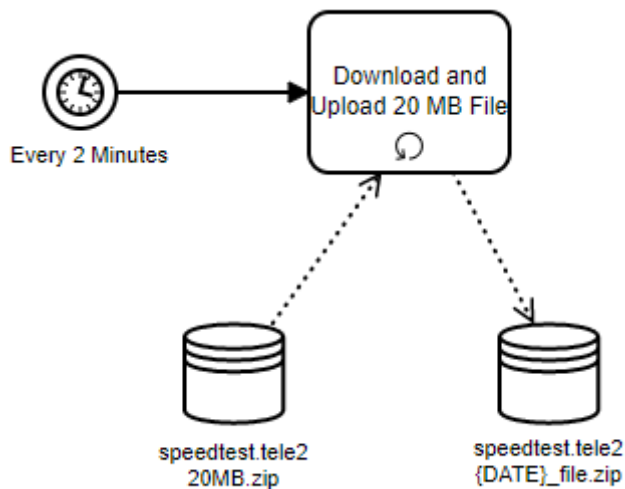
Prosessiautomaatioissa jokainen erillinen aktiviteetti suoritetaan omanaan ja ne sisältävät useimmiten useita erilaisia skenaarioita (kuva 8). Prosessiautomaatioihin voidaan liittää mukaan ohjelmistorobotiikka, koneoppimista ja tekoälyä. (Friends, n.d -a)

Kuva 8. Esimerkki Friends-prosessiautomaatiosta (Friends, n.d -a).



Sovellusintegraatioissa Friends voi noutaa tai vastaanottaa dataa lähtöjärjestelmästä, päivittää tai syöttää dataa kohdejärjestelmään ja yhdistää tietoja eri lähteistä. Friendsin tiedostosiirto tehdään Cobalt-tehtävällä. Tiedostosiirtoja voidaan tehdä FTP-, FTPS- ja SFTP-protokollien lisäksi Windows- ja Linux-pohjaisilla tiedostoratkaisuilla. Tiedostosiirto-ominaisuuksia (kuva 9) ovat mm. automatisoitu varmuuskopiointi ja varmuuskopioiden siivous ja tiedostojen uudelleen nimeämiset. Lisäksi Friends sisältää toimintoja erilaisten tilanteiden käsittelyyn (esimerkiksi tiedosto löytyy jo kohdekansiosta tai tiedosto on lukittu). (Friends, n.d -a)

Kuva 9. Esimerkki Frends-tiedostosiirrosta (Frends, n.d -a).



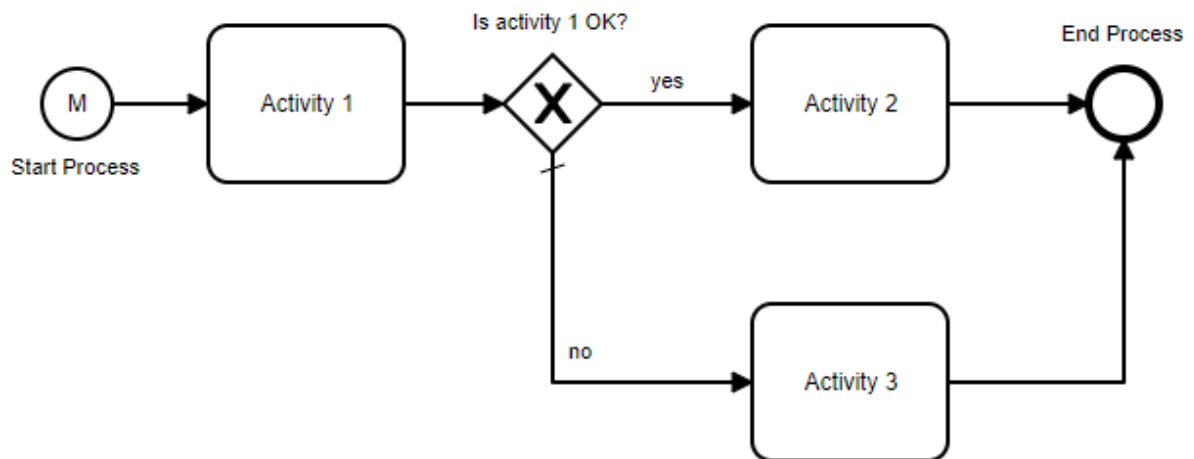
Frends voi käsitellä muun muassa JSON-, XML-, CSV-, EDI- ja kiinteä leveyksisiä tiedostoja. Tämä tarjoaa erittäin laajat mahdollisuudet tehdä erilaisia datamuunnoksia Frendsillä. Tiedostoja voidaan muuntaa muodosta toiseen tai tiedostosta löytyvällä tiedolla voidaan suorittaa esimerkiksi SQL-käskyjä. Frendsillä voidaan myös suunnitella, julkaista ja hallita API-rajapintoja. Frends käyttää OpenAPIa omassa API-hallinta toiminnallisuudessaan. Frends APIen avulla voidaan siirtää dataa tai ne voivat toimia myös toiminnallisina rajapintoina. Tällöin esimerkiksi APIa pitkin tulee pyyntö, joka käynnistää Frends-prosessin. (Frends, n.d - a)

Frends-prosessi sisältää aina alun, vähintään yhden tehtävän ja lopun. Prosessin aikana voidaan tehdä erilaisia päätöksiä tai silmukoita ynnä muita Frendsin tehtäviä. Esimerkiksi yksinkertaisessa prosessissa (kuva 10) siirrytään tehtävästä toiseen ja prosessissa loppuu vasta, kun kaikki tehtävät on suoritettu. Inklusiivisen päätöksen sisältävässä prosessissa (kuva 11) suoritetaan päätöksen jälkeen vain toinen jäljellä olevista tehtävistä. Lisää esimerkkejä Frends-prosesseista liitteessä 1.

Kuva 10. Malli yksinkertaisesta prosessista. Prosessi alkaa ja sen aikana suoritetaan kolme tehtävää, jonka jälkeen prosessi päättyy.



Kuva 11. Malli eksklusiivisesta päätöksestä, jossa suoritetaan tehtävä 2 tai 3, riippuen tehtävän 1 tuloksesta.



7.2 Friends-agentit ja -liittimet

Friends-prosessien suorittava osa on Friends-agentti. Yksittäinen Friends-prosessi voi käyttää agentteja useasta eri asennuksesta. Esimerkiksi yksi agentti voidaan hakea pilvestä ja toinen omasta konesalista. Friends-agenteilla on kolme kerrosta; hallintakerros, käynnistyskerros ja suorittava kerros. Hallintakerros keskustelee Friends-kontrollikeskuksen kanssa Azure Service Bus -jonojen kautta. Jonojen avulla voidaan päivittää agentin tietoja, käynnistää prosesseja manuaalisesti, käyttöönottaa työnkuluja ja raportoida monitorointiin dataa prosessista.

(Friends, n.d -a)

Käynnistyskerroksessa voidaan määrittellä, miten Friends-prosessi käynnistyy. Prosessi voi käynnistä esimerkiksi API-kutsulla tai ajastetusti. Kun prosessi on käynnistetty, suorittava kerros ajaa prosessin läpi ja palauttaa hallintakerroksen avulla monitorointitiedot takaisin kontrollikeskukseen. (Friends, n.d -a)

Friends-prosesseja rakennetaan tehtävien varaan. Nämä tehtävät toimivat Friends-prosesseissa liittiminä (Connectors). Friends käyttää Friendsin omia avoimen lähdekoodin liittimiä, kaupallisia liittimiä ja itserakennettuja tai kustomoituja liittimiä. Avoimen lähdekoodin liittimistä käytössä on REST, http, tiedostot, SQL, SFTP, FTPS, FTP, Service Bus, XML, JSON ja CSV. Kaupalliset liittimet toimivat plug-and-play periaatteella ja niitä löytyy todella paljon. Joukossa on liittimiä erilaisiin tietokantoihin (MariaDB, MongoDB, MySQL), toiminnanohjausjärjestelmiin (Dynamics NAV, Salesforce, Odoo), sosiaalisen median palveluihin (Facebook, Youtube, Twitter) jne. Omia liittimiä voidaan rakentaa .NET- ja C#-kielillä tai voi käyttää kolmansien osapuolien valmiita kirjastoja NuGet-paketeilla. (Friends, n.d -a)

7.3 Friends-integraatioalustan toimitusmuodot

Friends-agentteja voidaan asentaa käyttöön pilvestä, omaan konesaliin tai hybridimallina (pilvi + oma konesali). Kun agentteja asennetaan pilvestä, voidaan ne asentaa Microsoft Azureen joko virtuaaliselle Windows-palvelimelle tai Azure Kubernetes -palveluun, AWS:ään virtuaaliselle Windows-palvelimelle tai Amazon Elastic Container -palveluun, Google Cloudiin virtuaaliselle Windows-palvelimelle tai Google Kubernetes -moottorille tai yksityiseen pilveen Windows-palvelimelle tai Docker- tai Kubernetes-pohjaiseen kontitukseen. (Friends, n.d -a)

Kun Friends-agentit asennetaan omaan konesaliin, voidaan ne asentaa mille tahansa Windows 2012- tai tuoreemmalle palvelimelle. Agentit voidaan myös asentaa joko Docker- tai Kubernetes-kontituksena omalle palvelimelle. Hybridimallissa voidaan esimerkiksi asentaa järjestelmien ja datalähteiden yhteyksiin käytettävät agentit omaan konesaliin ja käyttää API-agentteja pilvestä. (Friends, n.d -a)

7.4 Friends tietoturva

Friends käyttää Azure Security Centerin parhaita käytäntöjä tietovuotojen aktiiviseen monitorointiin. Friends pystyy 36 tunnin sisällä vuodosta havaitsemaan vuodon ja sen laajuuden. Friendsiin pystytään tekemään laajamittainen datan käsittelyn auditointi Friendsin oman monitorointityökalun avulla. Monitoroinnin perusteella voidaan selvittää missä ja miten henkilödataa kussakin tilanteessa käytetään. Friendsistä voidaan tarpeen vaatiessa poistaa kaikki historia henkilödataa käyttävästä prosessista. Henkilödataa käsitteleviä prosesseja voidaan myös siistiä siten, että Friends-prosessi käyttää lähtödatasta vain niitä henkilödataa käsitteleviä kenttiä, joita prosessin ajamisen osalta on tarve. (Friends, n.d -b)

GDPR (General Data Protection Regulation) eli yleinen tietosuoja-asetus otettiin keväällä 2018 käyttöön henkilötietojen käsittelyä säänteleväksi laiksi. Sen tarjoaa paremman suojan ja hallittavuuden jokaiselle henkilölle omien henkilötietojen osalta. (Tietosuoja.fi, n.d) Tässä opinnäytetyössä käsitellään palkkaproessia ja siihen liittyvää henkilödataa, joten tietosuoja on otettava tarkasti huomioon osana prosessia.

GDPR:n vaatimuksissa on kaksi toimijaa, jotka lainsäädännöllisesti vaaditaan; tiedon hallitsija ja tiedon prosessoija. Hallitsija on toimi, joka määrittelee henkilödatan käytön tarkoituksen, ehdot ja välineet. Prosessoija on toimi, joka käsittelee henkilödataa hallitsijan puolesta. Friends-integraatiossa palvelun tilaaja toimii hallitsijana ja Friends-integraatio prosessoijana. Tästä asetelmasta johtuen Friends-integraatiot täyttävät GDPR:n asettamat vaatimukset. (Friends, n.d -b)

ElfGroup on tehnyt Friends-integraatioalustalle kyberturvallisuustestauksen, jossa on todettu, että Friends ei sisällä haavoittuvuuksia. Testauksen tuloksena Friends on saanut CyberSafe Solution -sertifikaatin. (CyberSecurityExecutive, n.d)

8 Kehittämistyön tavoite ja tarkoitus

Accountor Services Oy:n palkkapalveluissa on tarve erillisille tiedostosiirtointegraatiolle sekä sisäisesti että ulkoisesti asiakkaalle tai kolmansille osapuolille. Tiedostoja siirretään esimerkiksi työajanseurantajärjestelmien, HR-järjestelmien ja henkilökuntaetuuksia hallinnoivien järjestelmien ja palkkajärjestelmän välillä. Tiedostosiirron tarve ilmenee, kun ei pystytä rakentamaan rajapintaa järjestelmien välille. Palkkapalveluissa tiedostosiirtoja on toteutettu muun muassa SSIS-paketein verkkolevyiltä SFTP-kansioihin tai toisinpäin. Tiedostosiirtoja on tehty asiakaskohtaisten tarpeiden mukaan sekä yleissiirrot, joilla voidaan tehdä tiedostosiirtoja usealle asiakkaalle samanaikaisesti sekä sisään- että ulospäin. Ulospäin olevat siirrot ovat toimineet lähtökansiorakenteen perusteella. SSIS-paketti on tiennyt tiedoston saapuessa tietyllä kaavalla rakennettuun kansioon, että tämä tiedosto pitää siirtää SFTP-kansioon. Sisäänpäin siirtoja on ollut ohjaamassa erillinen XML-ohjaustiedosto, jonka perusteella SSIS-paketti on tiennyt tarkistaa, onko tiedostoja siirrettävänä, mitä siirretään, mistä siirretään ja minne siirretään. SSIS-paketit on palkkapalvelulle rakentanut IT-yksikkö.

SSIS-paketein tehtyjen siirtojen isoin ongelma on ollut niiden näkymättömyys palkkapalvelulle. Mahdollisissa virhetilanteissa palkkapalvelussa ei ole nähty, missä vika on tai koska vika on alkanut. Vikaa ei ole voitu myöskään itse korjata, vaan korjaus on pyydetty IT-tuesta tikettinä. SSIS-pakettien valvonta on ollut puutteellista ja sitä on parannettu vuosien varrella, mutta se on siltikin vielä rajallinen. Ulospäin olevan siirron ongelma on ollut myös ajastus, joka ei ole toiminut riittävällä tasolla. Sisäänpäin olevan siirron SSIS-paketti on rakennettu siten, että vain osalla palkkapalvelun pääkäyttäjiä ja sovellusasiantuntijoita on ollut mahdollisuus lisätä sinne uusia tehtäviä.

Tämän opinnäytetyön tuotoksena on kehitysprojekti, jossa rakennettiin uudenaikainen ja palkkapalvelun kannalta omavarainen yleistiedostosiirtointegraatio, jolla tulevaisuudessa korvataan SSIS-paketit. Alkuvaiheessa tälle ratkaisulle siirretään uudet tulevat siirrot. Integraatio toteutettiin FrenDS-integraatioalustalla ja integraatioiden hallintaa varten rakennettiin erillinen käyttöliittymä Microsoft Sharepoint -työkaluilla. Kehitystyön suunnittelun pohjana käytettiin vanhoja siirtopaketteja sekä workshop-tapaamista Accountor Services Oy:n kolmen sovellusasiantuntijan kanssa. Tiedostosiirrot ovat tärkeä osa palkkaprosessia ja niiden tarve kokonaisprosessin automatisoinnin kannalta on merkittävä.

Nykyaikaisella työvälineellä tehtäessä, saadaan tiedostosiirroista maksimaalinen hyöty irti ja kokonaisprosessin sujuvuus kasvaa. Lisäksi Friends-integraatioalustalla saadaan kehitettyä tiedostosiirtojen valvontaa ja hallintaa sekä avattua siitä osa palkkapalvelulle itselleen.

9 Yleistiedostosiirtointegraatiot Friends-integraatioalustalla

Työn pohjana olivat edellisessä luvussa esitellyt vanhat SSIS-paketeilla toteutetut siirrot. Työn tilaaja asetti kolme vaatimusta: 1) moderni ja näkyvä, 2) käyttäjäystävällinen tapa ja 3) tietoturvallinen tapa toteuttaa yleistiedostosiirtoja integraatioalusta Friendsiä käyttäen. Työn toteutus aloitettiin käymällä läpi vanhojen siirtojen hyvät ja huonot puolet sekä pidettiin workshop Mepco-pääkäyttäjien ja -sovellusasiantuntijoiden kanssa. Saadun palautteen perusteella laadittiin suunnitelma kokonaisprosessista, jota lähdettiin Friendsin ympärille rakentamaan.

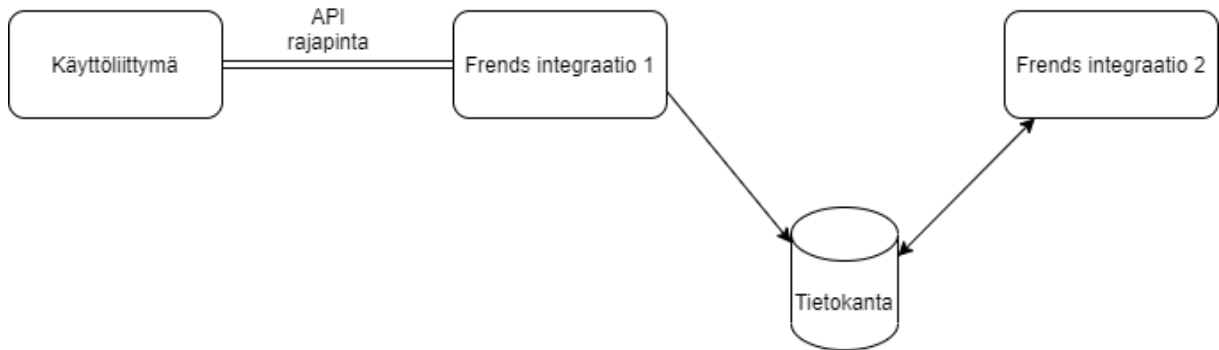
Vanhoista SSIS-paketeista ei integraation yksinkertaisen luonteen vuoksi löydetty juuri muita heikkouksia kuin näkymättömyys palvelulle, riittämätön ajastus ja virreehallinnan puutteellisuus. Workshopissa esiteltiin raakaversio suunnitelmasta, joka siinä kohtaa koostui kahdesta osasta eli käyttöliittymän rakentamisesta hallintaa varten sekä tiedostosiirtointegraation rakentamisen vanhojen siirtojen pohjalta Friendsillä. Pääkäyttäjiltä ja sovellusasiantuntijoilta saatiin uusina toiveina mahdollisuus datamuunnoksiin (tiedostojen nimet, tiedostomuodot, tiedostojen jakaminen useaksi tiedostoksi), paremmat ajastusmahdollisuudet (ajastus 24/7 tai heti kun tiedosto löytyy), siirron hallinnointi ja muokkausmahdollisuudet sekä testitiedoston ja yhteyksien testaaminen ennen tuotannon aloitusta.

Uusi yleissiirtojen kokonaisprosessi suunniteltiin näiden ennakkotietojen ja palautteiden pohjalta. Tämän opinnäytetyön tuotoksena on toteutettu järkeväksi kokonaisuudeksi katsottu osa tästä kokonaissuunnitelmasta. Tässä kappaleessa esitellään tehty kokonaissuunnitelma ja toteutetut osat. Toteuttamatta jääneet osat on kirjattu työhön kehitysehdotuksina.

Suunniteltu kokonaisprosessi (kuva 12) koostuu kolmesta osasta: käyttöliittymä ja kaksi erillistä Friends-integraatiota. Käyttöliittymän tarkoituksena on toimia yleissiirtojen tilaajien kanavana käytössä olevien siirtojen hallinnointiin sekä uusien tilaamiseen. Ensimmäinen Friends-prosessi hallinnoi Friends-integraatioalustalla yleissiirron piirissä olevia siirtoja. Sen tehtävänä on hakea tietoa käyttöliittymältä API-rajapintaa pitkin ja päivittää tietoja Friendsin SQL-relaatiotietokannassa. Lisäksi se testaa, että alustan käyttöoikeudet ovat riittävät

siirtojen tekemiseen sekä tutkii, onko käyttöliittymän kautta tehty muutoksia olemassa oleviin siirtoihin tai tilattu uusia. Jälkimmäinen Friends-integraatio on työvaihe, joka suorittaa aktiivisena olevat siirrot ajastetun aikataulun mukaisesti.

Kuva 12. Yleissiirto integraation kokonaisprosessi



Tämän työn tuotoksessa ei ole toteutettu API-rajapintaa eikä tietokannan käyttöönottoa. Tuotoksesta haluttiin rakentaa sellainen, joka voidaan ottaa käyttöön sellaisenaan ja jatkokehittääärkevin sekä hallittavin askelin eteenpäin. Jäljempänä esitellään tarkemmin käyttöliittymän ja Friends-integraatioiden osalta suunniteltu kokonaisuus ja toteutetut osat. Seuraavassa kappaleessa esitellään API-rajapinnan ja tietokannan osuus kokonaissuunnitelmassa.

API-rajapintaa käytettäisiin toiminnallisena sisäisenä rajapintana. Kun tilaaja käy tilaamassa uuden siirron tai tekee olemassa olevaan siirtoon muutoksia, kutsuu käyttöliittymä Friendsiä, joka noutaa tiedot rajapintaa pitkin. Noudon jälkeen Friends-integraatio 1 ajaa edellä esitetyt tarkistukset käyttöoikeuksista ja tietosisällöstä, jonka jälkeen se tallentaa uudet tai muutetut tiedot tietokantaan. Tietokantaan tallennetaan jokainen tiedostosiirto tehtävä omana tietueenaan. Tietue sisältää muun muassa tiedot siitä, kenen asiakkaan siirto on, mistä siirretään ja minne siirretään. Yksi tietue voi siis olla esimerkiksi Asiakas Oy:n työajanseurantajärjestelmästä tulevan työaikatiedoston siirtäminen SFTP-kansiosta Mepcoon. Yhdellä asiakkaalla on niin monta tietuetta, kuin on erilaisia siirtoja. Yksi tietue on Friendsille yksi suoritettava tehtävä.

Friends integraatio 2 lukee tietokannasta itselleen työlistalle ne siirrot eli tietueet, joita sen pitää missäkin ajossa suorittaa. Siirtoja rajataan ulos esimerkiksi aktiviteetin, ajankohdan ja

yleissiirroksi sopivuuden perusteella. Kun prosessi on suorittanut siirrot, palauttaa se tietokantaan erillisen tunnistetiedon, jos siirrossa on oikeasti suoritettu jotain.

Tunnistetiedon avulla voidaan ylläpitää tietokantaa ja tarkistaa aika ajoin, että kaikki siirrot, jotka on merkitty aktiivisiksi, myös oikeasti ovat sitä.

9.1 Käyttöliittymä

Käyttöliittymän välineeksi valittiin Microsoft Sharepoint. Sharepoint on yleisesti käytössä eri tiimeillä Accountorilla ja myös integraatioita varten on olemassa oma Sharepoint-kanava. Siksi oli luontevaa, että käyttöliittymä pyritään myös toteuttamaan sinne. Käyttöliittymä toteutettiin Power Apps -sovelluksena Sharepoint-luettelon päälle. Power Apps -sovelluksen lomakkeella voidaan avata tietoja luettelosta tai sinne voidaan tallentaa uutta tietoa. Suunnitelmassa käyttöliittymässä voitaisiin erikseen valita, halutaanko muokata olemassa olevaa vai tilata uusi siirto. Käyttöliittymällä voisi tilata myös muita kuin yleissiirroksi soveltuvia siirtoja. Käyttäjä ottaa kantaa tilausvaiheessa, millaisen siirron hän tarvitsee.

Käyttöliittymälle lisätyt kentät ovat:

- Siirron ID (automaattinen)
- Asiakkaan nimi
- Asiakkaan y-tunnus
- Siirron nimi
- Siirron tyyppi (in/out)
- Lähtökansio
- Kohdekansio
- SFTP tunnus
- Siirron viikonpäivä
- Siirron alkamisaika
- Siirron päättymisaika
- Yhteyshenkilön sähköposti
- Virheviestien sähköposti
- Lisätäänkö aikaleima? (kyllä/ei)
- Siirtosääntö (esim. *.csv)

- Tarvitaanko datamuunnoksia? (kyllä/ei)
- Datamuunnosten lisätiedot
- Aktiivinen (kyllä/ei)

Tämän työn tuotoksessa on toteutettu ainoastaan mahdollisuus tilata uusi siirto.

Käyttöliittymässä (kuva 13) on edellä esitetyt kentät ja tietojen tallennusta varten on rakennettu Sharepoint-luettelo.

Kuva 13. Sharepoint Power Apps -käyttöliittymä Friends-integraatioiden tilaamiseen.

The screenshot shows a configuration form titled "Friends siirrot". It includes the following fields and controls:

- * Siirron nimi: Text input field
- * Asiakkaan y-tunnus: Text input field
- * Lähtökansio: Text input field
- * SFTP käyttäjätunnus: Text input field
- Siirron alkamisaika: Text input field with placeholder "Syötä muodossa HH:mm (esim. 22:00)"
- * Yhteyshenkilöt - sähköpostit: Text input field
- Lisätäänkö tiedostonimeen aikaleima: Toggle switch (currently off), labeled "Ei käytössä"
- Tarvitaanko tiedostojen muokkausta?: Toggle switch (currently off), labeled "Ei käytössä"
- * Asiakkaan nimi: Text input field
- * Siirron tyyppi: Dropdown menu with "Etsi kohteita" selected
- * Kohdekansio: Text input field
- Siirron viikonpäivä: Dropdown menu with "Etsi kohteita" selected
- Siirron päättymisaika: Text input field with placeholder "Syötä muodossa HH:mm (esim. 22:00)"
- * Virheviestien sähköpostit: Text input field
- * Siirtosääntö: Text input field with placeholder "Esim. *.csv"
- * Aktiivinen: Toggle switch (currently on), labeled "Käytössä"
- Datamuunnosten lisätiedot: Large empty text area
- Tallenna: Button
- Tyhjennä: Button

9.2 Friends-ylläpitointegraatio

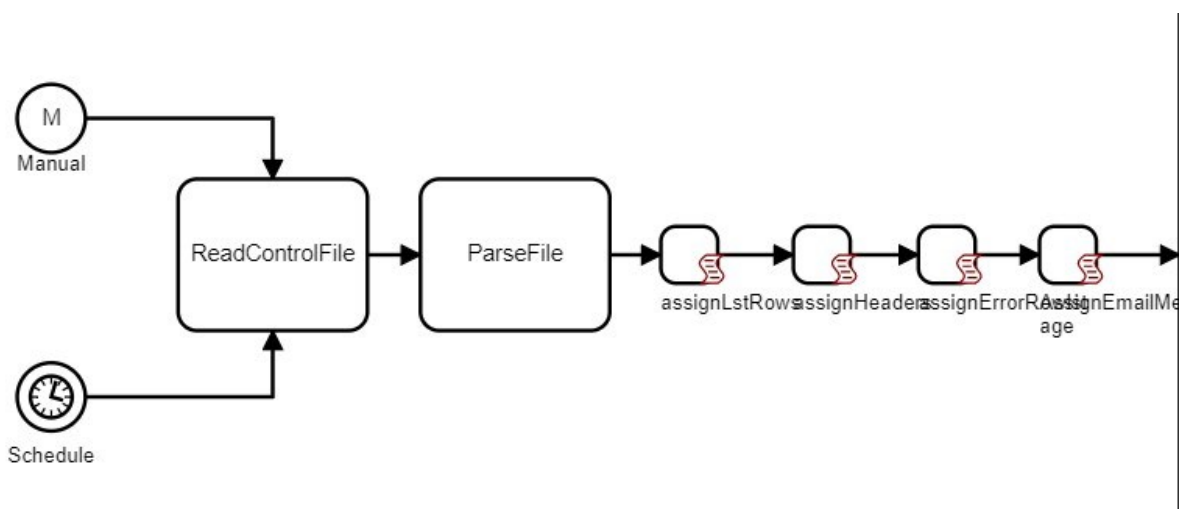
Ensimmäinen kahdesta Friends-integraatiosta suorittaa yleissiirtojen ylläpidollisia tehtäviä. Sen tehtävänä kokonaissuunnitelmassa on noutaa tietoa Sharepoint-luettelosta API-rajapintaa pitkin ja päivittää tietoja Friends-tietokantaan. Lisäksi se jokaisen noudon yhteydessä tarkistaa uusien ja muutettujen siirtointegraation kohteiden osalta, että Friendsillä on riittävät oikeudet suorittaa tiedostosiirtoja. Viimeisenä tämä integraatio erottelee uusista tilatuista siirroista ne, joihin tarvitaan vaikeampia ja laajempia datamuunnoksia omikseen sekä lähettää näistä sähköpostia integraatiotiimille, jotta niille voidaan määritellä omat asiakaskohtaiset prosessit.

Tämän työn tuotoksessa API-rajapintaa tai muuta automaatiota tietojen siirrolle Sharepointista Friendsiin, ei toteutettu. Myös Friends-tietokannan käyttöönotto rajattiin ulos. Sharepointiin käyttöliittymällä tallennetut tiedostosiirrot haetaan manuaalisesti CSV-tiedostona ja viedään Friends-verkkolevylle väliaikaistiedostona. Friends-prosessi noutaa väliaikaistiedoston verkkolevyltä ja tarkistaa siellä jokaisen rivin osalta, että Friendsillä on oikeudet sekä lähtö- että kohdekansioon. Tarkistuksen jälkeen onnistuneista riveistä kirjoitetaan uusi väliaikaistiedosto. Epäonnistuneet kirjoitetaan Friends-muuttujaan, jotka prosessin lopuksi lähetetään sähköpostilla integraatiotiimille.

Tarkistuksen jälkeen prosessi noutaa kopion tuotannossa olevasta CSV-ohjaustiedostosta ja vertaa sen jokaista riviä onnistuneista riveistä kirjoitettuun uuteen väliaikaistiedostoon. Mikäli joukosta löytyy muuttuneita tai uusia rivejä, kirjoitetaan näistä sähköposti integraatiotiimille, jotta he voivat vielä tarkistaa uudet rivit manuaalisesti. Lopuksi onnistuneista riveistä kirjoitettu väliaikaistiedosto tallennetaan uudeksi tuotannon ohjaustiedostoksi.

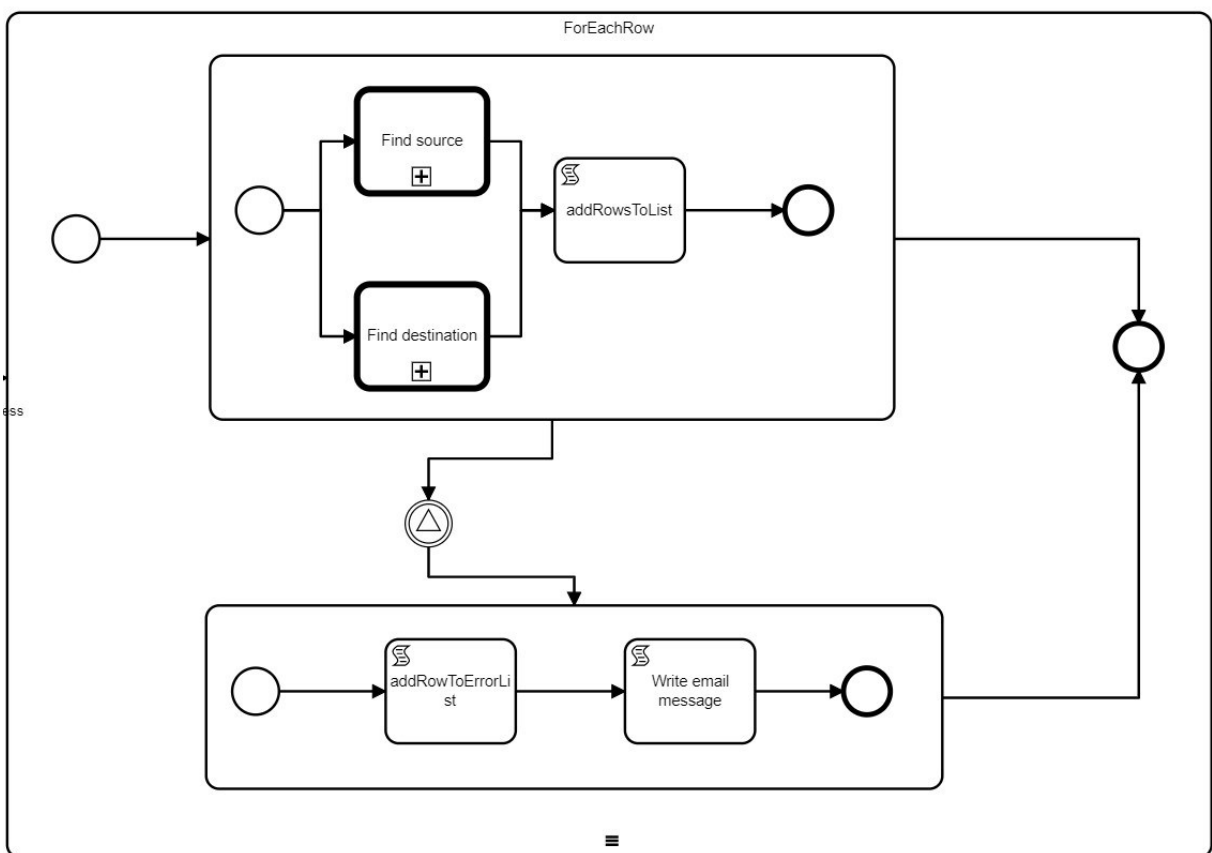
Prosessin tekoon on käytetty Friendsin valmiita tehtäviä. Prosessi alkaa (kuva 14) noudetun väliaikaistiedoston lukemisella (Read-tehtävä) ja tietojen parsimisella (Parse-tehtävä). Seuraavana alustetaan neljä myöhemmin käytettävää lista-muuttujaa (lstRows, lstRowHeaders, lstErrorRows ja emailMessage).

Kuva 14. Friends-ylläpitointegraatio ensimmäiset tehtävät.

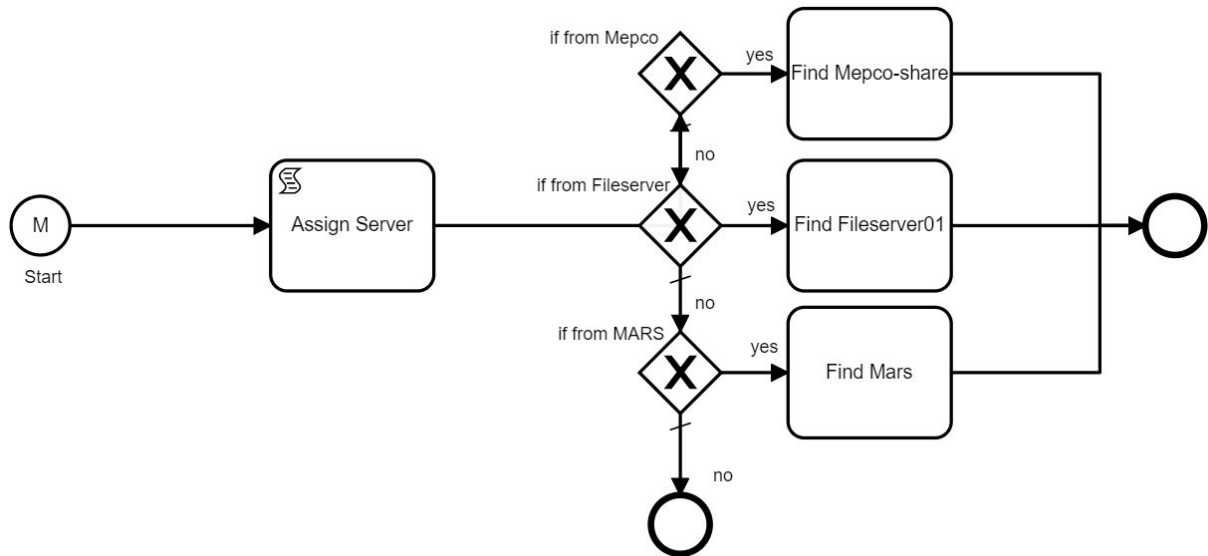


Prosessin toisessa vaiheessa (kuva 15) edellä esitetystä vaiheesta tiedostosta parsitut rivit luetaan jokainen erikseen for each -loopissa läpi. For each -silmukassa on Friendsin Scope-tehtävän sisällä sekä lähtö- että kohdekansiolle oma aliprosessi (kuva 16) oikeuksien tarkistamista sekä tehtävä IstRows-muuttujaan rivien kirjoittamista varten. Scopesta on erillinen Catch-tehtävä, joka poimii rivit, joissa oikeudet eivät ole kunnossa. Catch johtaa toiseen Scopeen, jossa epäonnistuneet rivit kirjoitetaan IstErrorRows-muuttujaan ja sen jälkeen lisätään IstErrorRows-tiedot emailMessage-muuttujaan.

Kuva 15. Friends-ylläpitointegraatio toisen vaiheen tehtävät.



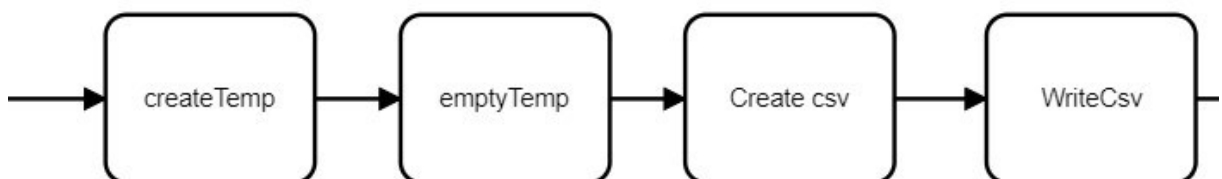
Kuva 16. Kansio-oikeuksien tarkistamisen aliprosessi.



Pääprosessi syöttää käynnistysparametrina aliprosessille kansiopulun, mikä halutaan tarkistaa. Prosessi hakee muuttujaan palvelimen, jolla kansio sijaitsee. Sen jälkeen tutkitaan millä palvelimella kansio sijaitsee ja tehdään sen mukainen tarkistus Find-tehtävällä. Friendsillä on omat tunnukset eri palvelimille, joten siksi tarkistus pitää tehdä jokaiseen palvelimeen omalla erillisellä Find-tehtävällä.

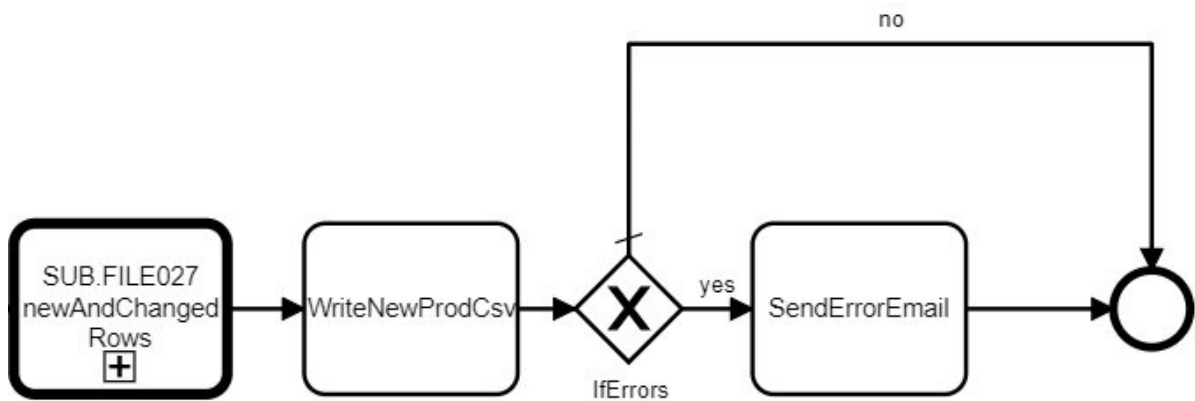
Kun väliaikaistiedoston kaikki rivit on käyty läpi, prosessin seuraavassa vaiheessa (kuva 17) luodaan Friendsin Create-tehtävällä väliaikaiskansio Friends-verkkolevyille, mikäli sitä ei löydy. Sen jälkeen Delete-tehtävällä tyhjennetään väliaikaiskansio ja luodaan Create-tehtävällä uusi tiedosto sekä kirjoitetaan for each -silmukan onnistuneista riveistä Write-tehtävällä IstRows- ja IstRowHeaders-muuttujista tiedostoon rivit.

Kuva 17. Friends-integraatio 1 kolmannen vaiheen tehtävät.

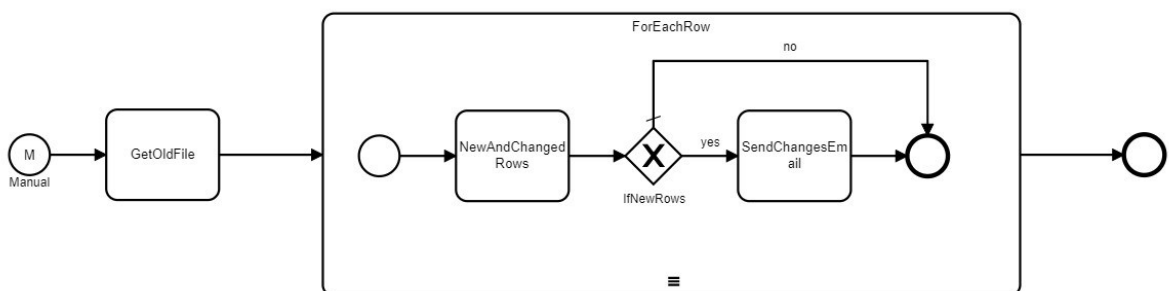


Friends-ylläpitointegraation viimeisessä vaiheessa (kuva 18) vertaillaan väliaikaistiedostoa ja tuotantotiedostoa erillisessä aliprosessissa, joka on luotu tätä integraatiota varten (kuva 19). Aliprosessissa haetaan kopio tuotantotiedostosta ja käydään läpi sen jälkeen jokainen rivi tuotannossa olevasta ja edellisessä vaiheessa luodusta väliaikaistiedostosta for each -loopissa. Vertailun jälkeen, jos löytyy uusia tai muuttuneita rivejä, lähetetään näistä tiedot sähköpostilla integraatiotiimille. Aliprosessikutsun jälkeen tallennetaan aikaisemmin kirjoitetuista onnistuneista riveistä muodostettu uusi tuotannon ohjaustiedosto vanhan päälle. Ennen prosessin päätöstä tutkitaan, löytyikö kansio-oikeuksien tutkimisessa virheellisiä rivejä. Jos löytyi, lähetetään sähköpostilla emailMessage-muuttujaan kerätyt tiedot virheellisistä riveistä.

Kuva 18. Friends-integraatio 1 prosessin viimeiset vaiheet.



Kuva 19. Tiedostojen vertailun aliprosessi.



Ylläpitoprosessi on ensimmäisessä vaiheessa manuaalinen, joten sille ei tehdä erillistä ajastusta. Se ajetaan tarpeen mukaan tai esimerkiksi kerran vuorokaudessa. Kun automaatio Sharepointin ja Friendsin välille saadaan rakennettua, voidaan ylläpitoprosessi ajastaa käynnistymään automaattisesti esimerkiksi kerran vuorokaudessa tai välittömästi, kun API-rajapintaa pitkin on saatu tietoa.

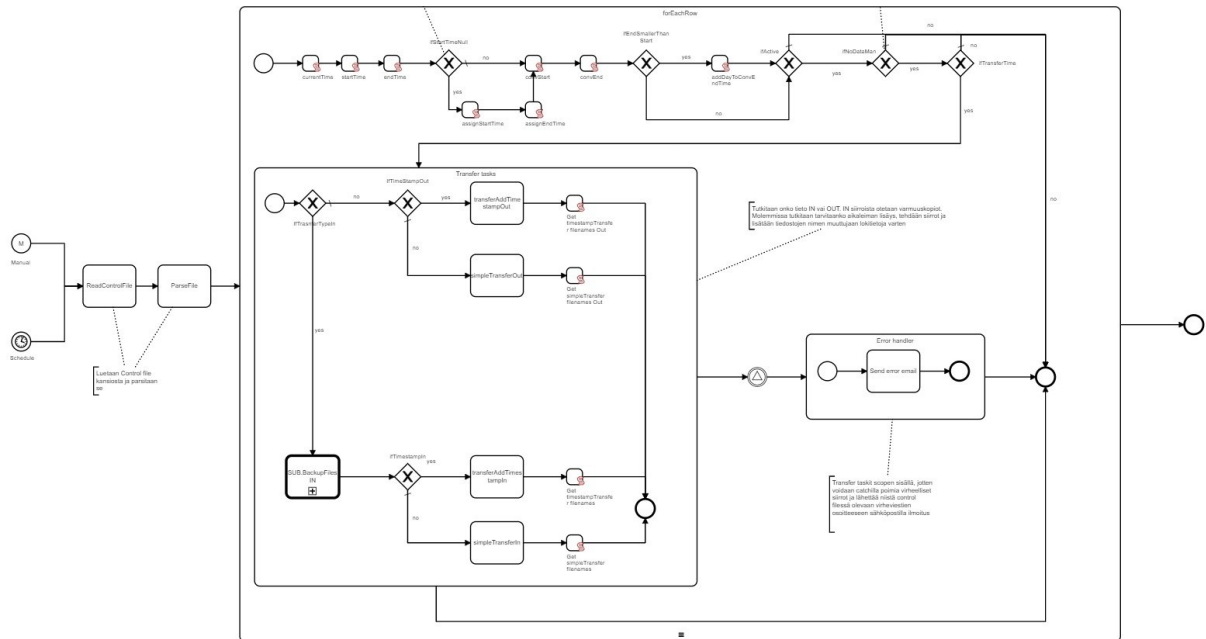
9.3 Friends-tiedostosiirtointegraatio

Friends-tiedostosiirtointegraatio suorittaa tiedostojen siirrot lähtökansioista kohdekansioon. Kokonaissuunnitelmassa prosessi noutaisi tiedot Friends-tietokannasta. Noudossa huomioidaan, että siirto on yleissiirroksi sopiva (ei sisällä laajoja datamuunnoksia), aktiivinen ja siirto on ajankohtainen (siirron viikonpäivä, siirron alkamisaika- ja päättymisaikaehdot täyttyvät). Näin saadaan siirtoon mukaan vain ne rivit, joita kulloinkin on tarkoitus ajaa.

Prosessissa tarkistettaisiin vielä ennen siirtoja, että tarvitseeko tiedostonimeen lisätä aikaleima sekä otettaisiin sisäänpäin tuotavista tiedostoista varmuuskopio. Ulospäin menevissä varmuuskopiointi saadaan hoidettua muilla tavoin. Siirron jälkeen Friends raportoisi kustakin siirrosta tiedot takaisin tietokantaan, jotta voidaan seurata mitkä siirrot oikeasti tekevät jotain. Lisäksi siirtojen yhteydessä on virnehallinta, jolloin epäonnistuneista siirroista saadaan lähetettyä virheviesti sekä integraatiotiimille että palvelulle.

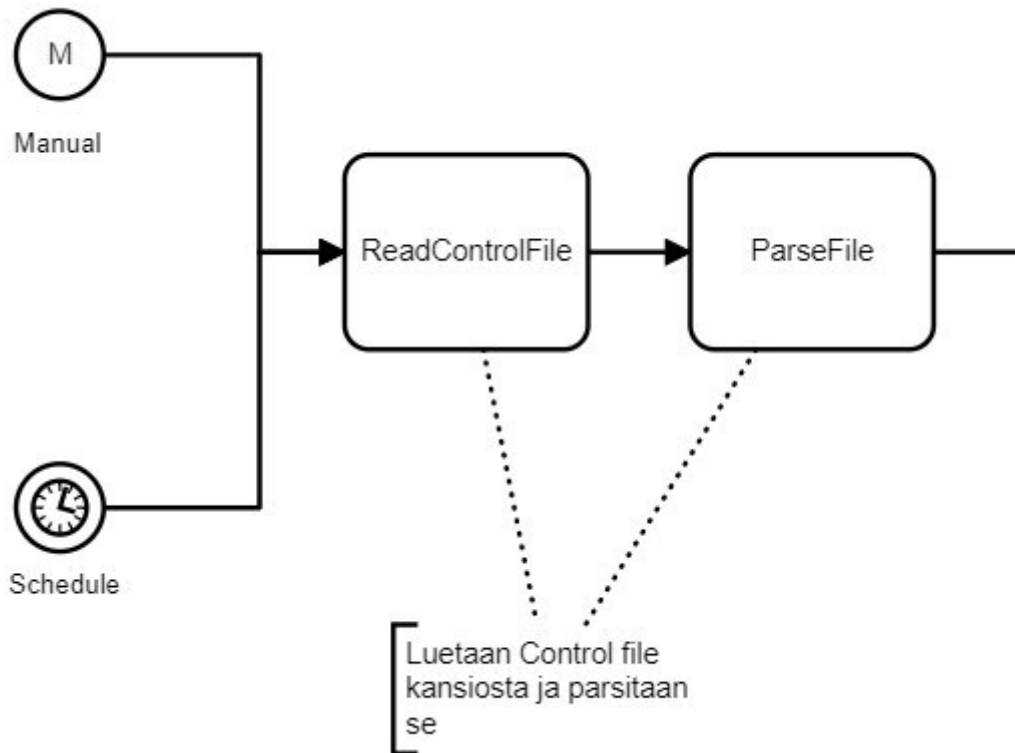
Tämän työn toteutuksessa tietokannan pois jättämisen myötä, yleissiirtointegraatio (kuva 20) eroaa joiltain osin suunnitelmasta. Siirron alussa luetaan CSV-ohjaustiedosto Friends-verkkolevyltä ja parsitaan sen jokainen rivi Friendsille. Parsittuja rivejä tutkitaan for each -silmukassa. Jos tiedostosiirto on aktiivinen, ei vaadi laajoja datamuunnoksia ja on ajankohtainen siirron alkamisajan ja loppuajan mukaan, prosessi etenee siirtovaiheeseen. Siirtovaiheessa tarkistetaan ennen siirtoja, onko siirron tyyppi sisään- vai ulospäin. Jos siirto tehdään ulospäin, ei tarvita varmuuskopioita, joten voidaan suorittaa siirrot. Mikäli siirto tehdään sisäänpäin, otetaan ensin tiedostoista varmuuskopiot. Molemmissa vaihtoehdoissa tarkistetaan vielä ennen varsinaista siirtoa, tarvitseeko tiedostoon lisätä aikaleima. Virneenkäsittelynä epäonnistuneista siirroista lähetetään viesti integraatiotiimille ja palvelulle.

Kuva 20. Toteutettu Friends-yleissiirtointegraatio.



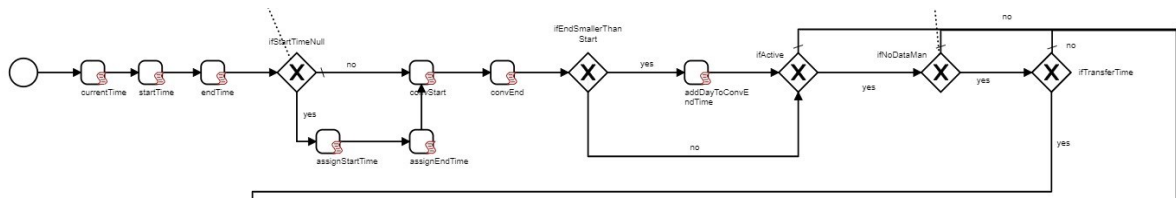
Prosessin ensimmäisessä vaiheessa (kuva 21) luetaan tiedosto Read-taskilla ja parsitaan sen jokainen rivi Parse-taskilla, samaan tapaan kuin tehdään ylläpidon prosessissa.

Kuva 21. Yleissiirtointegraation aloitus.



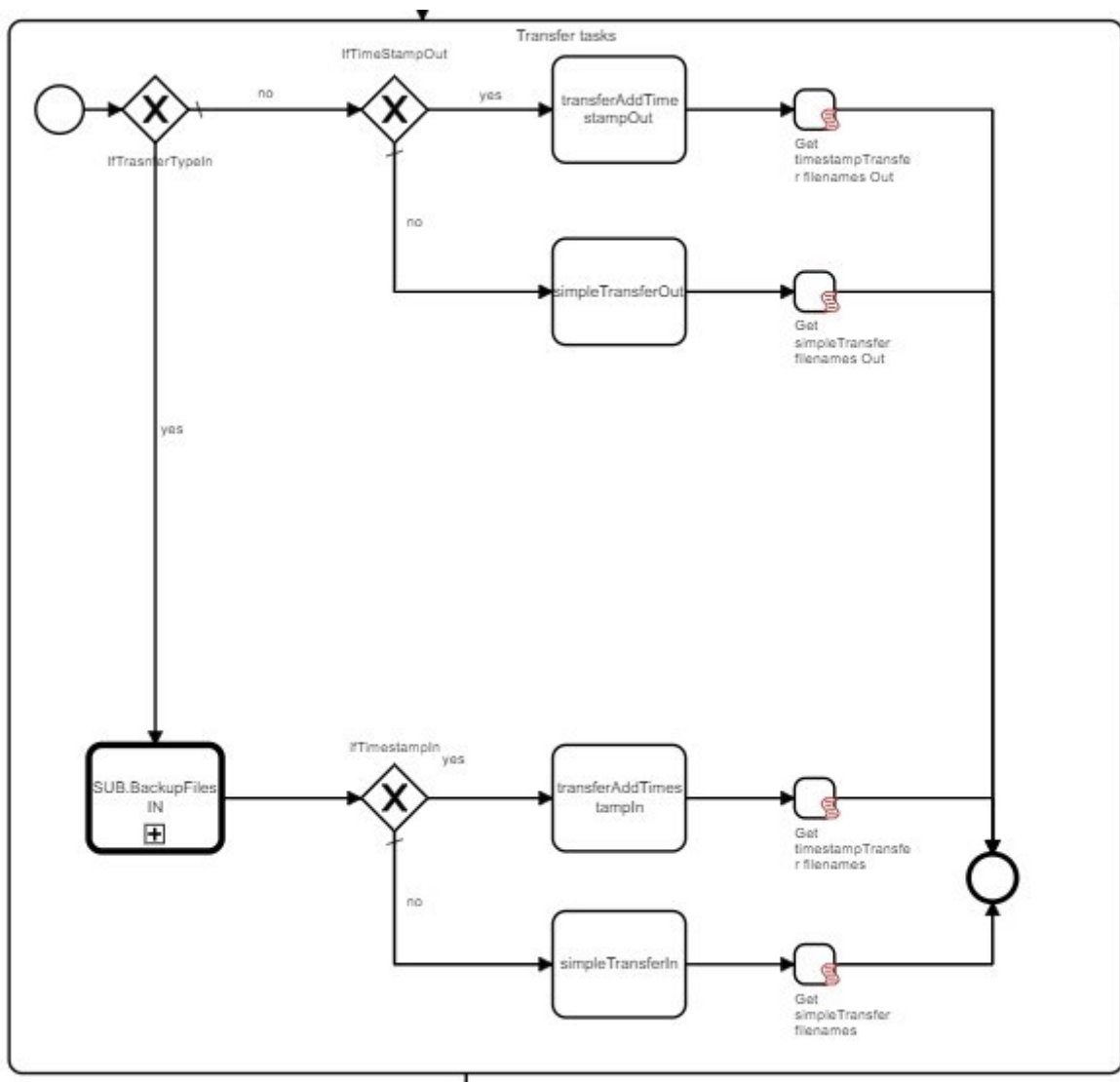
Toinen vaihe (kuva 22) on kokonaisuudessaan for each -loopin sisällä. For eachin sisällä ensimmäisenä haetaan muuttujiin ajohetki sekä tiedostolta siirron alku- ja loppuaika. Mikäli alkuajaa tai loppuaikaa ei ole annettu, annetaan Friendsissä alkuajaksi ajohetki -1 tunti ja loppuajaksi ajohetki +1 tunti. Vuorokauden vaihteen käsittelyä varten, täytyy alkuajaa ja loppuajaa vielä lisätä päivämäärä. Tämä tehdään C#-kielen Convert.ToDateTime-metodilla. Vuorokauden vaihteen käsittelyyn liittyen tutkitaan vielä, onko tiedostolla päättymisajan lukuarvo pienempi kuin alkuajan. Mikäli on, voidaan olettaa, että siirron halutaan alkavan eri vuorokauden puolella kuin päättymisen on (esimerkiksi 22:00 – 03:00). Tätä käsittelyä varten loppuajan muuttujaan lisätään +1 pv. Kun prosessissa on saatu muuttujat asetettua oikein, voidaan tutkia mitä siirtoja kuuluu tehdä. Mikäli tiedostolta luettu rivi on aktiivinen, ei sisällä laajoja datamuunnoksia ja ajohetki osuu siirron ajankohtaan, ryhdytään suorittamaan tiedostosiirtoja.

Kuva 22. For each -loopin muuttujien sijoittaminen ja siirron ehtojen tutkiminen.



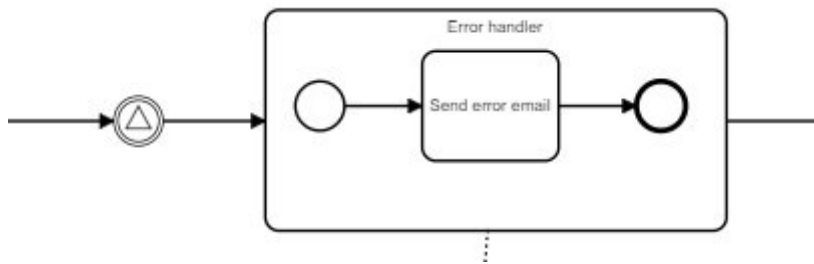
Itse tiedostosiirrot löytyvät for each -loopissa Scopen sisällä (kuva 23), jotta virnehallinta saadaan helposti toteutettua. Ennen tiedostosiirtoja, tutkitaan, onko siirron tyyppi sisään- (In) vai ulospäin (Out). Mikäli siirto tehdään sisäänpäin, otetaan ensin alkuperäisistä tiedostoista varmuuskopiot. Varmuuskopio on oma aliprosessinsa, johon Accountor Servicellä on määritelty tietyt säännöt muun muassa sen suhteen mihin varmuuskopio tallennetaan ja kuinka kauan sitä säilytetään. Varmuuskopion jälkeen tutkitaan, onko tarve lisätä tiedostonimeen aikaleima. Out-siirroissa ei tehdä varmuuskopioita, vaan siirrytään suoraan tutkimaan, onko tarvetta aikaleiman lisäykselle. Out-siirroissa varmuuskopiointi hoidetaan lähtöjärjestelmässä. Siirto tehdään Transfer-tehtävällä ja niitä on tässä prosessissa neljä, koska samaisessa tehtävässä voidaan tehdä myös aikaleiman lisäys. Kaksi prosessin Transfer-tehtävää siis sisältää aikaleiman lisäyksen (yksi per siirron tyyppi) ja kaksi ei. Kaikki Transfer-tehtävät on sijoitettu omalle polulleen prosessissa. Vielä ennen kuin Scopen sisällä oleva prosessi menee loppuun, otetaan siirrettyjen tiedostojen nimet talteen muuttujiin, jotta ne saadaan Friends-hallintamoduulin lokitietoihin.

Kuva 23. For each -silmukan sisällä oleva Scope, jossa tiedostosiirrot tapahtuvat.



Siirtoprosessin viimeisenä osana (kuva 24) on ylläpitoprosessin tapaan Catch-tehtävällä luotu virneenhallinta. Catch-tehtävä poimii Scopesta, jonka sisällä siirtotehtävät ovat, kaikki virheeseen jääneet siirrot ja niistä luodaan sähköpostiviestit, jotka lähetetään integraatiotiimille ja palvelulle. Virneenhallinta on omassa kokonaisuudessaan omassa Scopessaan. Tiedostosiirtoprosessi ajastetaan käynnistymään 30 minuutin välien 24 tuntia vuorokaudessa seitsemänä päivänä viikossa.

Kuva 24. Catch-tehtävä ja Scope virnehallintaa varten.



9.4 Tuotoksen tietoturva ja käyttäjäystävällisyys

Yleisesti ottaen tiedonsiirtojen ja integraatioiden tietoturva paranee, kun käytetään integraatioalustaa. Tämän työn tuotoksena tehty tiedostosiirto käsittelee tiedostoja, jotka sisältävät henkilötietoja, mutta koska itse tiedostojen sisältöä ei missään tilanteissa lueta, tässä työssä päähuomio tietoturvassa on ylläpidon puolella. Käyttöliittymää suunnitelman mukaisesti käyttävät vain Mepco-sovelluskonsultit sekä -pääkäyttäjät. Näin ollen saadaan käyttäjäjoukko rajattua samalla tavoin kuin se on rajattu itse Mepco-ohjelmassa. Sovelluskonsulteilla ja pääkäyttäjillä on sielläkin oikeus kaikkiin asiakkaisiin ja heidän tietoihinsa, joten samaa käytäntöä voidaan noudattaa käyttöliittymässä.

Integraatioissa käyttöoikeudet on rajattu integraatioita kehittäviin ja ylläpitäviin henkilöihin. Näin saadaan tehtyä rajaus muutoksiin integraatioissa ja tarkempaan lokitiedon tarkisteluun, vain niille henkilöille, joiden on tarve nämä tiedot nähdä. Tietokannan käyttö tiedostosiirtojen hallinnointiin lisää tietoturvaa ylläpidossa. Tietokantaan voidaan antaa oikeudet vain niille henkilöille, jotka käsittelevät tietokannassa säilytettävää tietoa. Myös mahdollinen API:n tai muun automaation käyttö vähentää tietoturvariskejä, kun tietoa ei esimerkiksi välitetä sähköpostin välityksellä.

Friends-integraatioalustalla on omat käyttöoikeudet, jotka sisältävät muun muassa pääsyoikeudet tietokantoihin ja verkkolevyille. Nämä eivät koskaan ole yksittäisen henkilön käyttöoikeuksia. Näitä hallinnoidaan Accountor Services Oy:n käyttäjäoikeushallintaprotokollien mukaisesti. Friendsissä käyttäjätunnukset ja salasanat

tallennetaan erillisiin ympäristömuuttujiin, joita voidaan käyttää eri prosesseissa. Näin saadaan käyttäjätunnusten ja salasanojen hallinta keskitettyä mahdollisimman pienelle joukolla henkilöitä. Rajaus voidaan tehdä esimerkiksi niin, että peruskäyttäjä ei näe ympäristömuuttujiin tallennettuja salasanoja, mutta pystyy käyttämään niitä omista prosesseistaan.

Käyttöliittymällä ja kattavalla virnehallinnalla lisätään käyttäjäystävällisyyttä verrattuna vanhaan tapaan toimia. Käyttöliittymän kautta Mepco-sovelluskonsulteilla ja -pääkäyttäjillä on mahdollisuus tarkistella perustietoja aktiivisena olevista siirroista ja tarpeen mukaan muokata niitä. Virnehallinta integraatioissa on toteutettu siten, että mahdollisissa virhetilanteissa virheviesti lähtee integraatiotiimin lisäksi palvelulle, mikä lisää integraatioiden läpinäkyvyyttä. Lisäksi kaksi tasoinen integraatioiden hallinta (ylläpito ja suorittava prosessi erikseen), lisää käyttäjäystävällisyyttä, kun yhteyden uusien tiedostosiirtojen kohdalla päästään testaamaan, ennen kuin tiedostosiirto varsinaisesti siirtyy tuotantoon.

10 Jatkokehitys, johtopäätökset ja pohdinta

Työn laajuudesta johtuen toteutusta rajattiin melko paljon. Edellisessä luvussa on esitelty tehty kokonaissuunnitelma ja siitä toteutetut osat. Jatkokehitykseen jäi suunnitelman loput osat, jotka esitellään seuraavissa kappaleissa.

Käyttöliittymästä toteutettiin työssä vain uusien tiedostosiirtojen tilaamisen käytettävä lomake. Jatkokehitykseen jäi käyttöliittymän osalta olemassa olevien tiedostosiirtojen listaaminen ja muokkaus. Nämä molemmat osat vaativat Power Apps -sovelluksen laajentamista. Listauksesta voisi valittua siirtoa klikkaamalla päästä näkemään tarkemmat tiedot ja tietojen tarkisteluun voisi samalla lisätä muokkaustoiminnallisuutta varten oman painikkeen, joka avaa muokkauslomakkeen. Listaukseen olisi hyvä lisätä myös haku, jotta siirtoja on helpompi etsiä.

Toinen kehityskohde on Sharepointin ja Friendsin välinen yhteys. Suunnitelmassa on ajateltu rakennettavan Sharepointin ja Friendsin välille API-rajapinta. Sharepoint kutsuisi APIa pitkin Friendsiä aina, kun käyttöliittymällä on tehty muutos tai lisätty uusi siirto. APIlle voisi olla vaihtoehtona myös Microsoft Sharepoint -työkaluista Power Automate, jolla voitaisiin suunnitella työnkulku, jossa esimerkiksi Sharepoint-luettelossa olevat tiedot siirrettäisiin CSV-tiedostona Friends-verkkolevyille. Työnkulku voitaisiin ajastaa tekemään siirto päivittäin, jolla taattaisiin, että Friendsillä on koko ajan ajantasainen tieto käytettävissä.

Friendsin puolella suurin kehityskohde on tietokannan käyttöönotto. Tällöin ensimmäinen Friends-prosesseista vertailisi Sharepointista tulevaa tietoa tietokannan tietoihin ja tekisi muutoksia sekä lisäyksiä APIsta tulevien tietojen perusteella. Jos käytössä olisi Power Automate -ratkaisu, vertailu tehtäisiin siirretyn tiedoston ja tietokannan välillä. Friends-integraatio ylläpitoprosessia tulisi muuttaa soveltuvin osin ja säilyttää siellä edelleen nykyiset toiminnallisuudet yhteyksien testaamisen ja virreehallinnan suhteen.

Tietokannan käyttöönoton myötä myös varsinaisia tiedostosiirtoja tehtävään prosessiin voisi tehdä yksinkertaistuksia. Sinällään nykyinen suunniteltu prosessi toimii ilman suurempia muutoksia, mutta yksinkertaistaminen keventäisi prosessia hiukan. Tässäkin prosessissa siirtojen tiedot luettaisiin tietokannasta nykyisen ohjaustiedoston sijaan. Yksinkertaistukset

koskisivat, miten tietoa luetaan tietokannasta. Nykyisessä prosessissa Friendsissä tehdään päätöksiä luetun tiedon perusteella, tarvitseeko kyseistä siirtoa tässä ajossa tehdä vai ei. Siirrot voitaisiin suodattaa suoraan tietokantahaussa siten, että Friendsissä ei ole tarvetta tehdä tätä tulkintaa. Tietokantakyselyssä tuotaisiin Friendsin työjonoon ainoastaan sellaiset siirrot, jotka ovat aktiivisia, yleissirroksi sopivia ja nykyhetki osuu pyydetyn siirron ajankohdan sisään.

Tiedostosiirtoprosessiin Friendsissä on suunnitelmassa määritelty myös tietokantaan palauttava osa niistä siirroista, joissa oikeasti jotain tapahtui. Tämä on osa, jota ei ole nykyiseen toteutukseen tehty ollenkaan. Friends voisi palauttaa esimerkiksi aikaleiman, jonka perusteella voidaan aika ajoin tarkistaa kaikki aktiiviseksi merkittyjen tiedostosiirtojen viimeisin tapahtuma ja tarpeen mukaan tarkistaa esimerkiksi onko kyseistä siirtoa tarve tehdä jatkossa.

Tämän opinnäytetyön tuotoksessa ei myöskään toteutettu aivan kaikkia toiminnallisuuksia, joita suunnitelmaan on kirjattu. Muun muassa siirtojen aikatauluttaminen viikonpäivän mukaan ja testitiedoston toimittaminen käyttöliittymällä jätettiin pois. Näiden pois jätettyjen toimintojen lisääminen ja jatkokehittäminen jää integraatiotiimin oman harkinnan varaan. Toteutuksessa rakennettiin ne toiminnallisuudet, joilla integraatio toimii tehokkaasti ja on laajuudeltaan riittävän laaja suorittamaan kaikki yleissirrot.

Friends-tietokannan käyttöönotto mahdollistaisi myös suuremman luokan kehityksen tiedostosiirtojen ja niihin liittyvien muidenkin integraatioiden käsittelyyn, joka ei suoranaisesti liity tähän opinnäytetyöhön, mutta esitellään tässä kuitenkin. Friendsiin voisi rakentaa yhden niin sanotun reader-prosessin, joka hallinnoisi kaikkia muita Friends-prosesseja. Tämä prosessi lukisi tietokannasta tietoa siirroista ja niiden suoritusajankohdista sekä -tavoista. Prosessi sen jälkeen ohjaisi muita Friends-prosesseja, kuten esimerkiksi tässä työssä toteutettua yleissirtoa ja kertoisi kullekin erilliselle prosessille mitä ja koska tehdään. Nyt Friends-integraatiot ovat omia irrallisia prosessejaan. Tällä tavoin kaikista saataisiin yhden kokonaisuuden piiriin kuuluvia osia. Tietokantaa lukeva ja muita ohjaileva prosessi olisi master-prosessi ja muut sen aliprosesseja.

Tuotoksen käyttöönottoa ei ole vielä tehty. Tuotos luovutetaan Accountor Services Oy:n integraatiotiimille, joka luovutuksen jälkeen tekee päätöksen, miten käyttöönotto suoritetaan. On mahdollista, että tuotos otetaan sellaisenaan käyttöön syksyn 2021 aikana ja jatkokehitetään tuotannon rinnalla erillisenä projektina. Tai sitten tuotosta kehitetään eteenpäin ennen käyttöönottoa. Esimerkiksi tietokannan lisääminen prosessiin on katsottu olevan merkittävässä roolissa myös tulevaisuuden prosesseissa. Tilaajan kanssa on keskusteltu myös mahdollisuudesta ottaa tämän työn tuotoksena toteutettu käyttöliittymä laajempaan käyttöön koko Accountor Services Oy:n tasolla, eikä ainoastaan palkkapalveluiden integraatioiden tilaus- ja hallintakanavana. Tällöin käyttöliittymän jatkokehityksessä pitää huomioida myös muiden yksiköiden erityistarpeet.

10.1 Johtopäätökset

Johtopäätöksenä tämän työn perusteella voidaan todeta, että automaatio ja sen osana integraatiot, ovat merkittävässä osassa, kun palkkaproessia kehitetään. Palkanlaskenta sisältää huomattavan määrän toistuvia ja helposti monistettavia tehtäviä, jotka ovat otollisia automatisoitavaksi. Liiketoimintaprosesseja voidaan automatisoida monella tapaa (integraatiot, RPA, tekoäly ja hyperautomaatio). Kaikkia näistä voidaan hyödyntää palkkaproessia automatisoitaessa, kuten työssä on edellä esitetty.

Palkanlaskenta linkittyy osana yrityksen talous- ja henkilöstöhallintoon, jonka johdosta sillä on paljon erilaisia sidosryhmiä. Sidosryhmien välinen tiedonsiirto ja sen sujuvuus on tärkeää. Tiedonsiirtoa voidaan hoitaa muun muassa rajapinnoilla ja integraatioilla. Integraatioista saadaan suurimmat hyödyt käyttämällä integraatioalustaa ja sen tarjoamia mahdollisuuksia. Integraatioalustojen avulla saadaan myös rajapinnat helpommin osaksi järjestelmäintegraatiota ja mahdollisesta esimerkiksi hyperautomaation käyttö.

Työn tuotoksena toteutettu tiedostosiirtointegraation modernisointi nykyaikaiselle integraatioalustalle, tuo tehokkuutta sekä Accountor Services Oy:n palkkapalveluihin, että integraatiotiimille. Tuotoksen käyttöönoton jälkeen palkkapalveluissa saadaan nopeallakin aikataululla uudet tiedostosiirrot käyttöön ja voidaan itse hoitaa näiden ylläpitoa. Integraatiotiimin ei jatkossa tarvitse tehdä jokaisesta yksinkertaisesta tiedostosiirrosta omaa määrittelydokumenttia ja konfiguraatiota integraatioalusta Friendsiin.

Työn tekijän mielestä laadittu kokonaissuunnitelma on toimiva ja askel oikeaan suuntaan ajatellen tiedostosiirtoja prosessina sekä automaation lisäämistä myös Accountorin sisäisissä palveluissa. Tekijä valitsi toteutukseen sellaiset osat, joista saatiin rakennettua toimiva kokonaisuus, joka voitaisiin ottaa käyttöön sellaisenaan. Tämä osuus työstä onnistui tekijän mielestä erityisen hyvin. Kaikki toteutetut osat tukevat toisiaan, muodostavat kokonaisprosessin ja toimivat myös sellaisenaan itsenäisinä työkaluina. Toisenlaisilla valinnoilla ei olisi saatu toteutettua haluttua kokonaisuutta.

Tekijän opinnäytetyöprosessi oli erityisesti tuotoksen osalta selkeä ja hallittu. Tuotokselle laadittu suunnitelma tehtiin yhteistyössä sekä palkkapalvelun että integraatiotiimin kanssa, jotta kaikille osapuolille saadaan tästä maksimaalinen hyöty. Suunnitelman pohjana olleet vanhat tiedostosiirrot ja niiden prosessi on tekijälle tuttu, joten sekin helpotti työn toteuttamisessa. Samoin palkkapalvelun tarpeet tiedostosiirroille olivat hyvin tekijän tiedossa. Tekijä pitää laadittua kokonaissuunnitelmaa hyvänä ja kattavana.

10.2 Pohdinta

Suurimmat haasteet tuotoksen osalta opinnäytetyöprosessissa olivat toteutukseen päätyneiden osien rajaaminen ja uusien järjestelmien opettelu. Rajaus onnistui hyvin ja suhteellisen nopeasti oli selvää mitkä osat kannattaa näin aluksi toteuttaa. Tekijä jäi pohtimaan sitä, että olisiko käyttöliittymä kannattanut jättää toteuttamatta kokonaan ja panostaa enemmän tietokannan käyttöönottoon, mutta työn tilaaja oli halukas sisällyttämään käyttöliittymän tähän tuotokseen.

Työn teoriaosa ja asetetut tutkimuskysymykset eivät suoraan kaikilta osin tue tuotoksena toteutettua työtä. Tämän vuoksi työn tekijä on valinnut teoriaan hiukan teknisemmän lähestymiskulman. Tiedostosiirrot ovat iso osa integraatiota, mutta eivät suoranaisesti liiketoimintaprosessin automatisointia palkanlaskennassa. Teoriaosassa suurimmat haasteet olivat palkanlaskennan automaatiosta kertovien hyvien lähteiden löytäminen.

Työn tilaaja pitää laadittua kokonaissuunnitelmaa hyvänä ja koko integraatioiden hallinnan osalta mahdollisuuksia lisäävänä työnä. Yleistiedostosiirtointegraatiot keventävät integraatiotiimille kohdistuvaa painetta ja työkuormaa, kun jokaisesta erillisestä siirrosta ei

tarvitse tehdä jatkossa omaa Friends-prosessia. Käyttöliittymä antaa mahdollisuuden laajentaa nykyistä sähköpostilla toimivaa tilauskanavaa sähköiseen muotoon sekä jossain määrin keventää tilausprosessia. Varsinkin hyvin yksinkertaisissa integraatiotarpeissa.

Pääkäyttäjille ja sovelluskonsulteille pidettiin katselmus käyttöliittymästä ja esiteltiin pääpiirteittäin kokonaissuunnitelma ja nyt toteutettu kokonaisuus. He pitivät käyttöliittymää hyvänä uudistuksena nykyisen word-pohjaisen määrittelydokumentin tilalle. Käyttöliittymä oli heidän mielestään selkeä ja riittävän yksinkertainen täyttää varten. Jatkokehitystoiveina saatiin yleissirrossa tiedostonimen muokkausmahdollisuus sekä seuranta varten lokitietojen kerääminen käyttöliittymässä tehtävistä muutoksista.

11 Yhteenveto

Työn tutkimuskysymykset olivat laajat. Tästä johtuen teoriassa on esitelty vastaukset kaikkiin kysymyksiin hyvin kompaktissa muodossa ja mahdollisimman käytännönläheisesti sekä työssä toteutettua toiminnallista osaa tukien. Tässä on onnistuttu hyvin.

Opinnäytetyön tuotoksen toteuttamiseen käytetyt välineet olivat minulle aikaisemmin tuntemattomia. Sharepoint on ollut käytössä jo pidempään, mutta koskaan aikaisemmin en ollut sinne luonut mitään sisältöä. Tämän vuoksi ajallisesti katsottuna isoin osa toteutusajasta meni Sharepoint-mahdollisuuksien tutkimiseen ja Power Apps -opiskeluun. Se kuitenkin kannatti, koska käyttöliittymästä sain toteutettua sellaisen version, jota on helppo tarpeen mukaan laajentaa ja jatkokehittää. Se on myös toteutettu niin, että siihen voisi lisätä tukifunktioita, kuten hyväksyntäkierto tai automaattiviestit.

Friends-integraatioalusta oli myös uusi työväline, mutta sen käyttö osoittautui loogiseksi ja yllättävän helpoksi, jolloin minun oli mahdollista hieman laajentaa integraatioalustalla tapahtuvia toiminnallisuuksia alkuperäisestä tavoitellusta lopputuloksesta muun muassa ylläpitoon käytettävällä prosessilla. Tämä teki tästä työstä kokonaisuutena yhtenäisemmän ja automatisoidumman sekä lisää tämän työn jatkokehitysmahdollisuuksia.

Tuotos ei ole vielä täysin laaditun kokonaissuunnitelman mukainen, mutta näen tämän kuitenkin vastaavan hyvin opinnäytetyölle asetettuihin tutkimuskysymyksiin automaation ja integraation merkityksestä nykypäivänä. Kokonaissuunnitelma kasvoi niin suureksi kokonaisuudeksi, että sitä ei ollut mahdollista toteuttaa kokonaan yhden opinnäytetyön tuotoksena. Siksi katsottiin parhaaksi, että rakennetaan kuitenkin selkeä kokonaisuus, jota voidaan sitten jatkaa, ennemmin kuin, että olisi tehty jokin irrallinen osio kokonaan loppuun asti. Tuotos on itsellään valmis käyttöönotettavaksi, mutta yksittäisen vaiheen tekeminen ei olisi tuottanut tätä haluttua lopputulosta.

Tulevaisuudessa olisi mielenkiintoista tutkia RPA:n ja Friendsin yhdistävää hyperautomaatiota ja sen hyödyntämismahdollisuuksia esimerkiksi palkkapalveluliiketoiminnassa.

Lähteet

- avoinrajapinta.fi. (11. 10 2014). *Avoin rajapinta*. Haettu 13. 6 2021 osoitteesta Avoimen rajapinnan määritelmä: <http://avoinrajapinta.fi/>
- Beasley, C. (30. 6 2020). *Fitsmallbusiness*. Haettu 28. 6 2021 osoitteesta What Is Payroll Software Integration?: <https://fitsmallbusiness.com/payroll-software-integrations/>
- Bourgeois, D.;& Bourgeois, D. T. (28. 2 2014). *Pressbooks*. Haettu 13. 6 2021 osoitteesta Information systems for business and beyond: <https://bus206.pressbooks.com/chapter/chapter-9-info-systems-strategic-advantage/>
- Chakray. (n.d). Haettu 31. 5 2021 osoitteesta What are the differences between REST and SOAP?: <https://www.chakray.com/what-are-the-differences-between-rest-and-soap/>
- Choudhuri, S. (1. 4 2019). *Actian*. Haettu 23. 7 2021 osoitteesta 5 ways a hybrid integration platform can make your data more secure: <https://www.actian.com/company/blog/5-ways-a-hybrid-integration-platform-can-make-your-data-more-secure/>
- Cronquist, D. (16. 4 2020). *Bamboohr*. Haettu 27. 6 2021 osoitteesta 8 Must-Have Benefits of Automated Payroll: <https://www.bamboohr.com/blog/benefits-of-automated-payroll/>
- CyberSecurityExecutive. (n.d). Haettu 23. 7 2021 osoitteesta Case Frends: Mission critical integration platform security testing: <https://www.cybersecurityexe.com/single-post/2019/11/08/case-frends-mission-critical-integration-platform-security-testing>
- Dvorak, J. (24. 1 2020). *Processand*. Haettu 13. 6 2021 osoitteesta What is business process?: <https://www.processand.com/blog/what-is-business-process>
- Eisner, M. (20. 8 2020). *ProcessMaker*. Haettu 19. 6 2021 osoitteesta What is Hyper-automatio?: <https://www.processmaker.com/blog/what-is-hyper-automation/>
- Ferguson, G. (n.d -a). *Small Business*. Haettu 27. 6 2021 osoitteesta What are the benefits of automating a payroll system?: <https://smallbusiness.chron.com/benefits-automating-payroll-system-2126.html>
- Ferguson, G. (n.d -b). *Small business*. Haettu 30. 6 2021 osoitteesta What are the Benefits of Automating a Payroll System?: <https://smallbusiness.chron.com/benefits-automating-payroll-system-2126.html>
- Ferguson, J. (17. 9 2019). *TrustRadius*. Haettu 21. 7 2021 osoitteesta What are the Best iPaaS for Enterprises?: <https://www.trustradius.com/buyer-blog/ipaas-for-enterprises>

- Fiedler, M.-O. (n.d). *Payzaar*. Haettu 27. 6 2021 osoitteesta Process Automation in Payroll – GPMI Feature Article: <https://payzaar.com/robotic-process-automation-in-payroll/>
- Flashnode. (30. 5 2021). *Itewiki*. Noudettu osoitteesta Digitalisoinnin opas: <https://www.itewiki.fi/opas/integraatiot/>
- Frankenfield, J. (31. 8 2020). *Investopedia*. Haettu 19. 6 2021 osoitteesta What is machine learning?: <https://www.investopedia.com/terms/m/machine-learning.asp>
- Frankenfield, J. (8. 3 2021). *Investopedia*. Haettu 19. 6 2021 osoitteesta Artificial intelligence (AI): <https://www.investopedia.com/terms/a/artificial-intelligence-ai.asp>
- Frends. (n.d -a). Haettu 23. 7 2021 osoitteesta <https://frends.com/platform>
- Frends. (n.d -b). Haettu 23. 7 2021 osoitteesta Frends eiPaaS GDPR data processing description: <https://frends.com/gdpr/>
- Gartner. (n.d -a). Haettu 21. 7 2021 osoitteesta Enterprise low-code application platforms (LCAP) reviews and ratings: <https://www.gartner.com/reviews/market/enterprise-low-code-application-platform>
- Gartner. (n.d -b). Haettu 21. 7 2021 osoitteesta What is iPaaS (Enterprise Integration Platform as a Service): <https://www.gartner.com/reviews/market/enterprise-integration-platform-as-a-service>
- Goel, A. (30. 5 2021). *Enterprise Integration*. Noudettu osoitteesta EAI vs. SOA vs. ESB : <https://hosteddocs.ittoolbox.com/Enterprise%20Integration%20-%20SOA%20vs%20EAI%20vs%20ESB.pdf>
- Haglund, J. (30. 5 2021). *Alfame*. Noudettu osoitteesta Järjestelmäintegraatio: Mitä se on selkokielellä: <https://www.alfame.com/blog/jarjestelmaintegraatio-mita-se-on-selkokielella>
- Henderson, C. (26. 3 2021). *Anyconnector*. Haettu 15. 6 2021 osoitteesta Business Process Automation 101 - Types of BPA & When to Use Them: <https://anyconnector.com/business-process-automation.html>
- HiQ. (n.d -a). Haettu 23. 7 2021 osoitteesta Frends integraatioalusta: https://hiq.fi/palvelut/integroii/frends/?utm_term=integraatioalusta&utm_campaign=SEM%2BDisplay+%E2%80%93Suomi,+Integraatio&utm_source=adwords&utm_medium=ppc&hsa_acc=6423282974&hsa_cam=44306175&hsa_grp=85065617255&hsa_ad=401250273032&hsa_src=g&hsa_tgt=kwd

- HiQ. (n.d -b). Haettu 23. 7 2021 osoitteesta Mikä on Friends ja millaista sillä on toteuttaa integraatioita: <https://hiq.fi/blogi/mika-on-friends-ja-millaista-silla-on-toteuttaa-integraatioita/>
- Honkanen, R. (n.d). *Integrata*. Haettu 20. 7 2021 osoitteesta Ohjelmistorobotiikka palkanlaskennan apurina: <https://www.integrata.fi/ohjelmistorobotiikka-palkanlaskennassa/>
- ISA. (n.d). *Internatiol Society of Automation*. Haettu 30. 5 2021 osoitteesta What is automation: <https://www.isa.org/about-isa/what-is-automation>
- Kaarlejärvi, S.;& Salminen, T. (2018). *Älykäs taloushallinto - automaation aika*. Helsinki: Alma Talent Oy.
- Kehal, B. (3. 8 2019). *Medium*. Haettu 13. 6 2021 osoitteesta A look into automation its different types: <https://medium.com/aiautomation/a-look-into-automation-its-different-types-f4266049f54d>
- Keskiruokonen, M. (30. 6 2020). *Pinja*. Haettu 23. 7 2021 osoitteesta Integraatioiden piilevät tietoturvaasteet ratkaistaan modernilla integraatioalustalla: <https://blog.pinja.com/integraatioiden-piilevat-tietoturvaasteet-ratkaistaan-modernilla-integraatioalustalla>
- Korhonen, P. (10. 1 2018). *CGI*. Haettu 31. 5 2021 osoitteesta Pieni API-sanakirja: <https://www.cgi.com/fi/fi/blogi/pieni-api-sanakirja>
- Kosinski, J. (12. 5 2021). *Patriosoftware*. Haettu 30. 6 2021 osoitteesta What is Payroll Integration, and How Can it Help Your Business?: <https://www.patriosoftware.com/blog/payroll/payroll-integration/>
- Kouhia-Kuusisto, K.;Mikkonen, L.;Syvänperä, O.;& Turunen, L. (2017). *Palkkavuosi*. Edita.
- Kugasia, A. (10. 4 2019). *Azilen*. Haettu 20. 7 2021 osoitteesta Take a Leap in Payroll Management using Advanced AI Capabilities: <https://www.azilen.com/blog/payroll-management-using-advanced-ai-capabilities/>
- Kääriäinen, J.;Aihkisalo, T.;Halén, M.;Holmström, H.;Jurmu, P.;Matinmikko, T.;. . . Tirronen, J. (2018). *Ohjelmistorobotiikka ja tekoäly - soveltamisen askelmerkkejä*. Valtioneuvoston kanslia. Haettu 31. 5 2021 osoitteesta <https://julkaisut.valtioneuvosto.fi/bitstream/handle/10024/161123/65-2018-Ohjelmistorobotiikka%20ja%20tekoaly.pdf?sequence=1&isAllowed=y>
- Lahti, S.;& Salminen, T. (2014). *Digitaalinen taloushallinto*. Alma Talent Oy.

- Makadia, M. (28. 10 2020). *Business 2 community*. Haettu 20. 7 2021 osoitteesta HR Automatio - 12 ways RPA can streamline your HR and payroll: <https://www.business2community.com/human-resources/hr-automation-12-ways-rpa-can-streamline-your-hr-and-payroll-02357074>
- Mattinen, K.;Orlando, C.;& Parnila, K. (2014). *Palkanlaskenta käytännönläheisesti*. Helsingin seudun kauppakamari.
- McDermott, D. (12. 2 2021). *ProcessMaker*. Haettu 13. 6 2021 osoitteesta What is business process automation (BPA)?: <https://www.processmaker.com/blog/what-is-business-process-automation-bpa/>
- McHugh, B. (n.d). *IT Automation without boundaries*. Haettu 19. 6 2021 osoitteesta What is intelligent automation?: <https://www.advsyscon.com/blog/intelligent-automation/>
- Moore, B. (2. 4 2001). *TechRepublic*. Haettu 31. 5 2021 osoitteesta An introduction to the Simple Object Access Protocol (SOAP): <https://www.techrepublic.com/article/an-introduction-to-the-simple-object-access-protocol-soap/>
- Mustonen, M. (n.d). *HiQ*. Haettu 13. 6 2021 osoitteesta Älykäs prosessiautomaatio ja hyperautomaatio: <https://hiq.fi/blogi/alykas-prosessiautomaatio-ja-hyperautomaatio/>
- O'brien, R. (1. 3 2008). *Hubpages*. Haettu 30. 5 2021 osoitteesta Integration Architecture Explained: <https://discover.hubpages.com/business/Integration-Architecture-Explained>
- Pal, S. (18. 2 2021). *Zenesis*. Haettu 20. 7 2021 osoitteesta RPA Use Cases In Payroll Management: <https://www.zenesis.com/blog/rpa-use-cases-in-payroll-management>
- Pratt, M. (27. 2 2021). *Baeldung*. Haettu 31. 5 2021 osoitteesta Difference Between REST and HTTP: <https://www.baeldung.com/cs/rest-vs-http>
- Pratt, M. K. (11. 5 2021). *TechTarget*. Haettu 18. 6 2021 osoitteesta RPA vs. BPA vs. DPA: Compare process automation technologies: <https://searchcio.techtarget.com/tip/Process-automation-technologies-evolve-RPA-vs-BPA-vs-DPA>
- Pursiainen, T. (16. 9 2020). *CGI*. Haettu 18. 6 2021 osoitteesta Hyperautomaation lyhyt historia: <https://www.cgi.com/fi/fi/blogi/hyperautomaation-lyhyt-historia>
- Radin, B. (21. 9 2017). *Global payroll association*. Haettu 20. 7 2021 osoitteesta What will artificial intelligence mean for payroll?:

<https://globalpayrollassociation.com/blogs/latest-news/what-will-artificial-intelligence-mean-for-payroll>

Rock, G.; & Dwyer, T. (n.d). *BPMInstitute*. Haettu 13. 6 2021 osoitteesta What is BPM Anyway? Business Process Management Explained:

<https://www.bpminstitute.org/resources/articles/what-bpm-anyway-business-process-management-explained>

Ruokonen, R. (19. 4 2021). *Netvisor*. Haettu 27. 6 2021 osoitteesta Hyvä ohjelmisto + moderni tapa toimia = paras palkkaprosessi: <https://netvisor.fi/blog/paras-palkkaprosessi/>

Ryhänen, J. (8 2019). *Devisioona*. Haettu 21. 7 2021 osoitteesta Johdanto integraatioihin ja ipaasiin: <https://www.devisioona.fi/2019/08/johdanto-integraatioihin-ja-ipaasiin/>

Sen, S. (1. 4 2021). *Paylitehr*. Haettu 20. 7 2021 osoitteesta How will AI impact the future of payroll processing?: <https://www.paylitehr.com/blog/how-will-ai-impact-the-future-of-payroll-processing/>

Shankar, H. (2. 6 2020). *Zoho*. Haettu 21. 7 2021 osoitteesta What is an integration platform and how does it work?: <https://www.zoho.com/flow/articles/how-integration-platform-works.html>

Siegel, C. (28. 10 2020). *API Friends*. Haettu 13. 6 2021 osoitteesta What is an API: <https://apifriends.com/api-management/what-is-an-api/>

Silva, W. (30. 3 2020). *Benefits pro*. Haettu 20. 7 2021 osoitteesta Improving the payroll process with artificial intelligence: <https://www.benefitspro.com/2020/03/30/improving-the-payroll-process-with-artificial-intelligence/?slreturn=20210620031934>

Silven-Hoell, K. (n.d). *Smallbusiness*. Haettu 28. 6 2021 osoitteesta HR Payroll System Integration and Benefits: <https://smallbusiness.chron.com/hr-payroll-system-integration-benefits-19068.html>

Smartbear. (2. 1 2020). *SOAP vs REST. What's the Difference?* Haettu 31. 5 2021 osoitteesta <https://smartbear.com/blog/soap-vs-rest-whats-the-difference/>

Tietosuoja.fi. (n.d). *Usein kysyttyä EU:n tietosuoja-asetuksesta*. Haettu 8.9.2021 osoitteesta <https://tietosuoja.fi/gdpr>

Tirkkonen, J. (16. 5 2019). *Alfame*. Haettu 13. 6 2021 osoitteesta Liiketoimintaprosessien pullonkaulat auki - miten valita sopiva automaattioratkaisu?:

<https://www.alfame.com/blog/liiketoimintaprosessi-pullonkaulat-auki-miten-valita-sopiva-automaattioratkaisu>

Toivanen, A. (27. 12 2019). *Linkedin*. Haettu 21. 7 2021 osoitteesta Nykyaikaiset integraatioalustat: <https://www.linkedin.com/pulse/nykyaikaiset-integraatioalustat-antti-toivanen/>

Toivanen, A. (n.d -a). *HiQ*. Haettu 30. 5 2021 osoitteesta Integraatiot: <https://hiq.fi/ajankohtaista/integraatio/>

Toivanen, A. (n.d -b). *HiQ*. Haettu 19. 6 2021 osoitteesta Mitä ovat hyperautomaatio ja hyperautomaatioalusta?: <https://hiq.fi/ajankohtaista/mita-ovat-hyperautomaatio-ja-hyperautomaatioalusta/>

Tähtinen, S. (16. 8 2016). *Youredi*. Haettu 21. 7 2021 osoitteesta 3 reasons to use a data integration service: <https://www.youredi.com/blog/3-reasons-to-use-a-data-integration-service>

Varon, L. (18. 1 2021). *Sweet process*. Haettu 13. 6 2021 osoitteesta The Complete Guide To Business Process Automation (And How To Find The Right Software For Your Company): <https://www.sweetprocess.com/business-process-automation/>

Ventä, O.;Honkatukia, J.;Häkkinen, K.;Kettunen, O.;Niemelä, M.;Airaksinen, M.;& Vainio, T. (2018). *Robotisaation ja automatisaation vaikutukset Suomen kansantalouteen 2030*. Valtioneuvoston kanslia. Noudettu osoitteesta https://julkaisut.valtioneuvosto.fi/bitstream/handle/10024/161102/47-2018-ROBOFINN_raportti_.pdf

Verohallinto. (14. 12 2020). Haettu 3. 7 2021 osoitteesta Tieto rajapinnoista: <https://www.vero.fi/tietoa-verohallinnosta/yhteystiedot-ja-asiointi/asiointiverkossa/tietoa-rajapinnoista/>

Verohallinto. (1. 1 2021). Haettu 3. 7 2021 osoitteesta Tietoa tulorekisteristä: <https://www.vero.fi/tulorekisteri/tietoa-meist%C3%A4/>

White, D. (6. 1 2020). *Techfunnel*. Haettu 27. 6 2021 osoitteesta 9 Ways Robotic Process Automation Is Changing the Future of Payroll: <https://www.techfunnel.com/hr-tech/9-ways-robotic-process-automation-is-changing-the-future-of-payroll/>

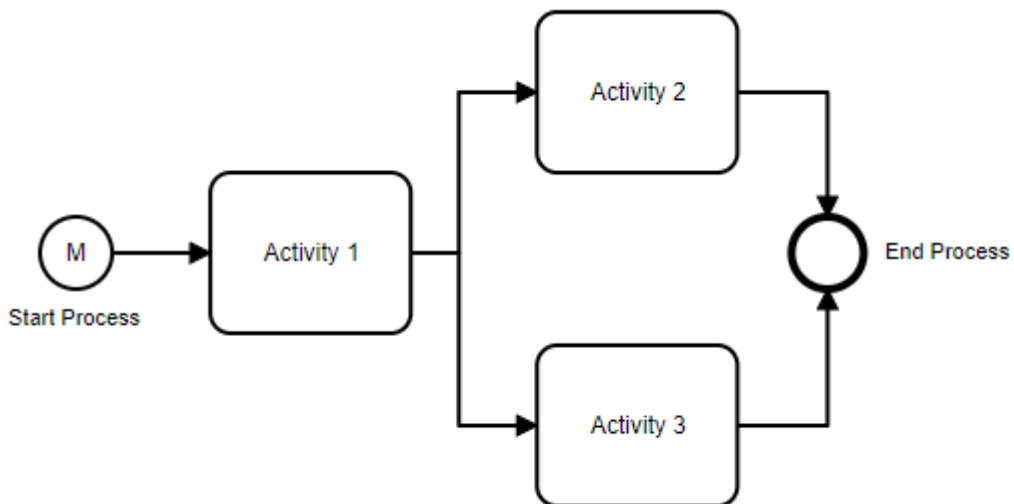
Xabeire, S. (n.d). *Insperity*. Haettu 28. 6 2021 osoitteesta HR technology: Integration with legacy systems done right: <https://www.insperity.com/blog/hr-technology-integration/>

Ylén, J.-P.;Ventä, O.;Tommila, T.;Lappalainen, J.;Hirvonen, J.;Karhela, T.;. . . Voho, P. (2010).

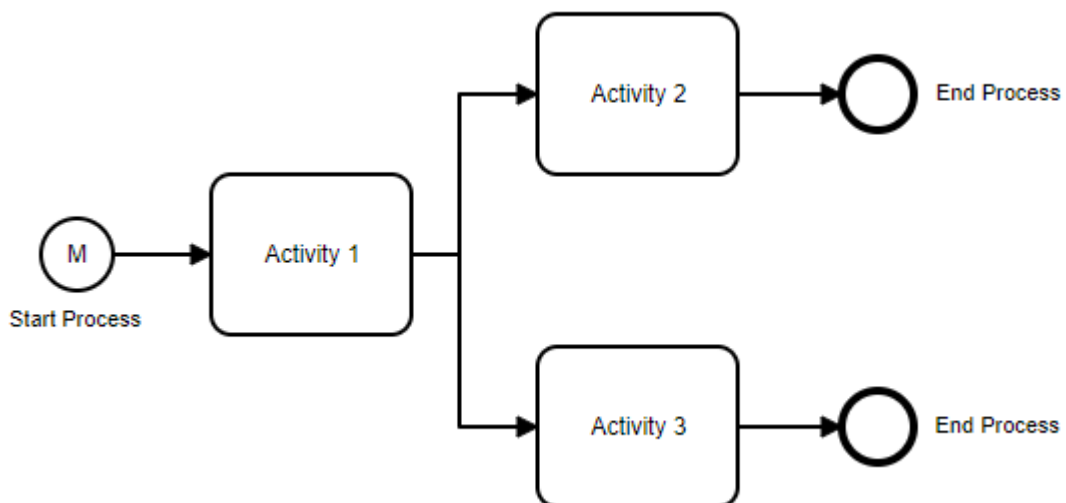
Automaatiot liiketoimintaprosessien tukena. Helsinki: Tekes.

Liite 1: Friends-integraatioiden mallikuvat

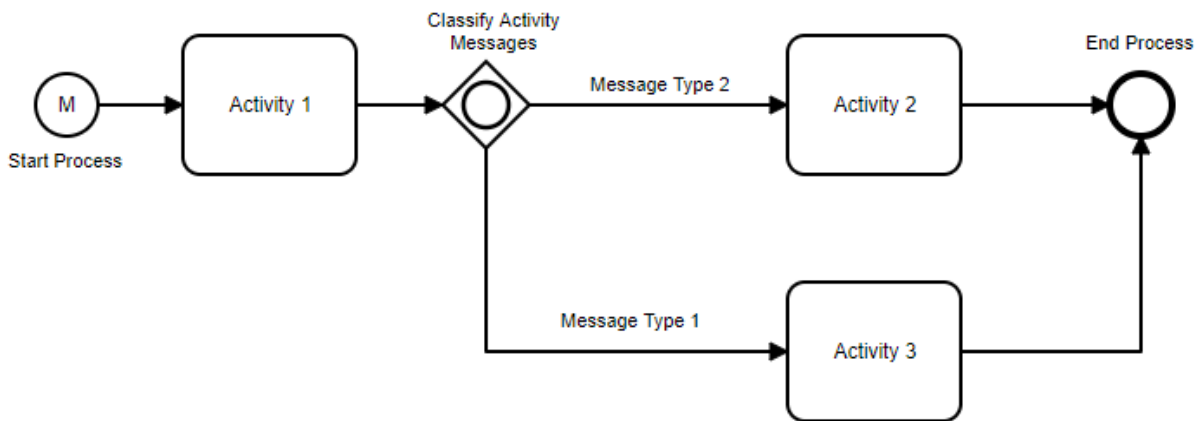
Malli rinnakkain ajettavista tehtävistä, jossa tehtävä 2 ja 3 suoritetaan samanaikaisesti.



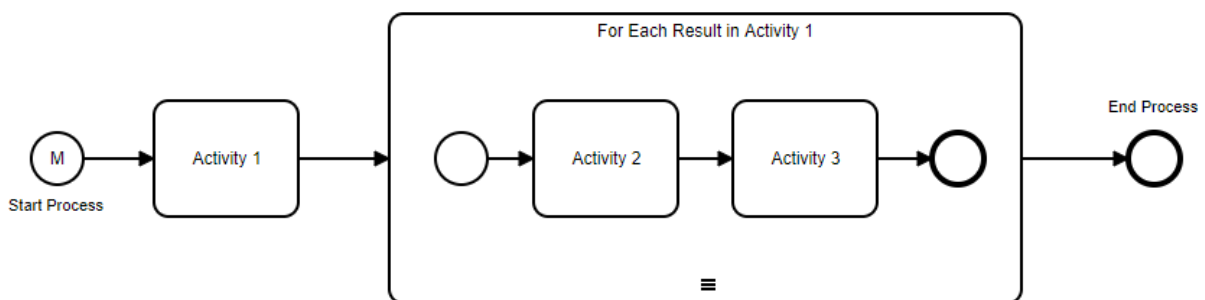
Malli jaetusta prosessista, jossa tehtävät 2 ja 3 sisältävät omat loppupisteensä.



Malli inklusiivisesta päätöksestä, jossa suoritetaan tehtävä 2 tai 3, riippuen tehtävän 1 tuloksesta.



Malli silmukasta, jossa tehtävän 1 tulos (esimerkiksi listassa tai taulukossa) suoritetaan For Each -silmukassa erikseen.



Liite 2: Aineistonhallintasuunnitelma

Kehitysprojekti:

Kehitysprojektin aineisto on toimeksiantajan omaisuutta. Aineistoa säilytetään toimeksiantajan aineistonhallintakäytänteiden mukaisesti ja toimeksiantajan ympäristössä. Kehitysprojektin aineistona on käytetty vanhojen SSIS-pakettien määrittelydokumentaatiota, tehtyä selvitystä SSIS-paketeista, sovelluskonsulttien ja pääkäyttäjien kanssa pidetyn workshopin muistiinpanot sekä kehitysprojektityöstä toimeksiantajalle tehty määrittelydokumentti. Aineistoja säilytetään toimeksiantajan säilytysaikakäytäntöjen mukaisesti.

Opinnäytetyöstä on säilytetty kahta eri kopiota tekijän tietokoneen C-asemalla ja kolmatta HAMKin OneDrive-pilvitalennustilassa.