



Päiväkirjaopinnäytetyö: Ohjelmistorobottien hallinnan siirto SaaS-palveluun

Hanna Rekola

Haaga-Helia ammattikorkeakoulu

Tradenomi tietojenkäsittely

AMK-opinnäytetyö

2024

Tiivistelmä

Tekijä(t) Hanna Rekola
Tutkinto Tradenomi, tietojenkäsittely
Raportin/Opinnäytetyön nimi Päiväkirjaopinnäytetyö: Ohjelmistorobottien hallinnan siirto SaaS-palveluun
Sivu- ja liitesivumäärä 43
<p>Tässä päiväkirjaopinnäytetyössä seurataan opiskelijan ammatillista kehitystä ohjelmistorobottien hallintatyökalun siirtoprojektin aikana. Seurantajakso on 12.08.2024-22.11.2024 jona aikana opiskelija työskentelee Accountor Solutions -nimisessä yhtiössä automaatiokehittäjän roolissa. Accountor Solutions tuottaa IT- ja kehityspalveluja Accountor Services yhtiöiden maaorganisaatioille Pohjoismaissa. Opinnäytetyö koostuu päiväkirjamerkinnöistä, joissa seurataan projektin etenemistä ja analyyseistä, joiden tietoperusta koostuu sovellusten ohjeista ja matalakoodia, integraatioita sekä business intelligence -ratkaisuja käsittelevästä kirjallisuudesta.</p> <p>Seurantajakson lähtötilanteessa ohjelmistorobottien hallinta oli toteutettu organisaation omassa laiteympäristössä olevasta UiPath Orchestrator -sovelluksesta. Projektin aikana Orchestrator siirrettiin toimittajan pilvipalveluun ja robotit kytkettiin ohjattavaksi sieltä. Pilvipalvelusta saatiin kustannushyötyjä verrattuna aiempaan tilanteeseen. UiPath robotit toimivat edelleen organisaation omassa laiteympäristössä ja niitä ohjataan pilvestä käsin. Projektin suunnitteluvaiheen aikana ilmeni, että roboteille on ennen Orchestratorin siirtoa tehtävä versiopäivitys. Ohjelmistorobottien ajamien automaatioiden tulosten raportoinnin tekninen toteutus oli lähtötilanteessa toteutettu robotiikan alihankkijan toimesta. Projektissa se suunniteltiin ja toteutettiin organisaation sisäisesti käyttämällä Friends integraatioalustaa.</p> <p>Opinnäytetyön tavoitteena oli kehittää opiskelijan ammatillista osaamista robottien hallinnassa eli UiPath Orchestrator -tuotteen osaamista. Toinen työn tavoite oli ymmärryksen lisääminen sisäisen raportointiratkaisun toteutuksesta. Seurantajaksolla raportoitiin vain siirtoprojektiin liittyvät tehtävät, ja työn ulkopuolelle rajattiin muut päivittäiset työtehtävät. Kehittymistavoitteet saavutettiin seurantajakson aikana osittain. Raportointiratkaisun suunnittelun ja toteutuksen osalta tavoitteet täyttyivät kokonaan, mutta Orchestratorin ominaisuuksien osalta tavoitteet tulevat täyttyiksi kokonaisuudessaan vasta kun uusien ominaisuuksien käyttöönotto etenee seurantajakson jälkeisenä aikana.</p>
Asiasanat Ohjelmistorobotiikka, projektit, automaatio, kehittäminen

Sisällys

1	Johdanto	1
1.1	Päiväkirjamerkintöjen aikaväli	1
1.2	Yritys, toimenkuva ja työympäristö	1
1.3	Työtehtävissä tarvittava osaaminen	1
1.4	Ammatillisen kehittymisen tavoitteet ja rajaukset.....	2
1.5	Keskeiset ammattikäsitteet ja lähteet	2
2	Lähtötilanteen kuvaus.....	4
2.1	Oman nykyisen työn analysointi	4
2.2	Sidosryhmien esittely	5
2.2.1	Sisäiset sidosryhmät	5
2.2.2	Ulkoiset sidosryhmät.....	6
2.3	Työpaikan vuorovaikutustilanteet	7
3	Seurantajakson raportointi viikkoanalyysineen	8
3.1	Seurantaviikko 1.....	8
3.2	Seurantaviikko 2.....	11
3.3	Seurantaviikko 3.....	16
3.4	Seurantaviikko 4.....	21
3.5	Seurantaviikko 5.....	25
3.6	Seurantaviikko 6.....	30
3.7	Seurantaviikko 7.....	34
3.8	Seurantaviikko 8.....	37
4	Pohdinta.....	41
	Lähteet.....	43

1 Johdanto

1.1 Päiväkirjamerkintöjen aikaväli

Päiväkirjaopinnäytetyössäni seuraan ohjelmistorobotiikan robottien ohjaamiseen käytettävän työkalun, nimeltään UiPath Orchestrator, siirtoa yrityksen omasta Azuren pilviympäristöstä ohjelmiston toimittajan hallinnoimaan SaaS-pilviympäristöön. Projektin tavoitteena on karsia robottien hallinnan alustan ja ylläpidon kustannuksia. Projekti toteutetaan elo-marraskuun aikana ja seurantajakso ajoittuu 12.08.2024-22.11.2024 väliseen aikaan. Viiden päivän pituisten seurantajaksojen jälkeen analysoin työtehtäviäni ammatillisen kasvun näkökulmasta. Seurantajaksoja on yhteensä kahdeksan.

1.2 Yritys, toimenkuva ja työympäristö

Accountor Solutions on yhtiö, joka tuottaa Accountorin tilitoimisto-, palkanlaskenta- ja HR-palveluiden maayhtiöille eli Accountor Servicesille yhteisiä palveluja, kuten IT-hallintoa, järjestelmien kehitystä ja liiketoimintatiedon hyödyntämisen palveluja. Meitä on noin 40 henkeä ja palvelemme sisäisesti yli 1000 henkeä viidessä eri maassa. Accountor Services toimii vuonna 2024 maayhtiöinä Suomessa, Ruotsissa, Tanskassa, Hollannissa, Norjassa ja Ukrainassa. Keväällä 2024 Accountor yhdistyi ruotsalaisen Aspian kanssa, yhdessä yhtiö on pohjoismaiden suurin tilitoimisto-, palkanlaskenta-, ja HR-palveluja tuottava yritys.

Robottien ohjaus on käytössä neljässä eri maassa; Ruotsissa, Norjassa, Suomessa ja Hollannissa. Tämän vuoksi projektissa tulee suunnitella ja testata tarkoin, että siirron jälkeen edelleen automaatiot toimivat ja saamme toimittajan pilvipalvelusta tarvittavat tiedon raportointia varten. Osallistun SaaS-siirto projektiin toteuttaen suunnittelua, testaamista, asennusta ja mahdollisesti kehittämällä tarvittavia uusia ohjelmiston osia nykyisiin automaatioprosesseihin. Tavoitteena on sujuva ohjelmiston siirto ja tuloksena toimiva ja kustannustehokas ratkaisu robottien ohjaamiseen.

1.3 Työtehtävissä tarvittava osaaminen

Työtehtäväni koostuvat automaatioiden kehittämisestä ja seurannasta sekä automaatioympäristön ylläpitoon liittyvistä tehtävistä. Tarvitsen osaamista UiPath ohjelmistorobotiikkasovelluksesta sekä perusymmärrystä Azuren pilvi-infrastruktuurista ja tietoverkoista. Lisäksi minun on osattava eri ohjelmointikieliä, kuten C#, VB, SQL ja XSLT, jotta osaan vähintään lukea koodia ja myös luoda sitä. Ongelmanratkaisu- ja tiedonhaku-aidot ovat kuitenkin tärkeimmät työssäni tarvittavat taidot, sillä ympäristö ja sovellukset, joissa ohjelmistorobotit toimivat on hyvin moninainen.

1.4 Ammatillisen kehittymisen tavoitteet ja rajaukset

Opinnäytetyöni seuranta-ajan tavoitteena on, että osaamiseni syvenisi UiPath Orchestrator tuotteen sekä sisäisen raportointiratkaisun suhteen. Odotan oppivani lisää robottien ohjaamisen hallinnoinnista koska tällä hetkellä ohjauksen ylläpito on ulkoistettu ja tämän projektin myötä se siirtyy meidän tiimiimme sisäisesti hoidettavaksi sekä SaaS-palveluksi UiPathilta.

Opinnäytetyöni ei käsittele vertailua siitä kumpi on parempi, pilvipalvelu vai on-premise-ratkaisu. En sisällytä yksityiskohtaista UiPath Orchestrator tuotteen toiminnallisuuden analyysia työhöni, vaan käsittelen toimintoja sen mukaan, mitkä niistä ovat käytössä organisaatiossamme. Työni tavoite ei ole esitellä Azuren pilven tai SaaS-ohjelmistojen ominaisuuksia. En tee yksityiskohtaista selvitystä Accountorin tarjoamiin palveluihin, automaatioprosesseihin tai automaatioiden rakentamisen toimintatapoihin enkä myöskään maakohtaisesti erittele käytössä olevia automaatioita sillä nämä ovat salassapitovelvoitteen alaista tietoa.

1.5 Keskeiset ammattikäsitteet ja lähteet

Ohjelmistorobotiikka: Sääntöpohjaisia ja toistuvia tietotyön tehtäviä voidaan automatisoida skriptien tai rajapintojen kautta ohjelmistorobotiikan, eli RPA:n (Robotic Process Automation), avulla. Ohjelmistorobotti on nopea ottaa käyttöön käyttöliittymässä tehtäviin automaatioihin koska se simuloi ihmisen työtehtäviä. (IBM, 2024.).

UiPath: Alusta, joka sisältää ohjelmistorobotiikan sovelluksen sekä automaatioiden ohjauksen, kartoittamisen ja analysoinnin työkaluja. (UiPath, 2024.).

Frends: Integraatioalusta, jolla voi matalakoodia käyttäen rakentaa järjestelmäintegraatioita ja automatisoida tiedonsiirtoa. (Frends, 2024).

PowerBi: Microsoftin sovellus, johon voi kerätä tietoa eri lähteistä kuten excelistä ja tietokannoista ja visualisoida tieto halutussa muodossa erilaisille raporteille. (Microsoft, 2024.)

On premise: Organisaation omassa laiteympäristössä asennettuna oleva sovellus, jota organisaation sisäinen taho ylläpitää ja valvoo. (AWS, 2024.)

SaaS: Pilvipohjainen ohjelmistomalli, jossa sovellus toimitetaan käyttäjille internetselaimesta. Ohjelmiston toimittaja huolehtii sovelluksen ylläpidosta, päivityksistä ja saatavuudesta. (AWS, 2024.)

Azure: Microsoftin pilvialustapalvelu, joka tarjoaa palvelinkapasiteettia, sovelluskehityksen työkaluja, tallennuspalveluja sekä lisäpalveluja kuten verkko- ja käyttäjätunnusten määrittämisen palveluja. (Sulava, 2023.)

Tietoperusta koostuu UiPathin verkkosisällöistä, matalakoodisovelluksia käsittelevistä julkaisuista, UiPath-kehittäjien kurssikirjoista, Friends integraatioalustan ohjeista sekä PowerBi raportointiratkaisujen oppaista ja BI-ratkaisuista kertovasta kirjallisuudesta. Ammatillisen kehittymisen keskeiset teemat ovat ohjelmistorobottien ohjaus sekä automaatioprosessien seurannan raportointitoteutus. Selvitän opinnäytetyössäni, miten nämä ovat nyt toteutettu organisaatiossani ja miten ne tullaan toteuttamaan jatkossa. Yhdistän tähän teoreettista tietoa ohjelmistorobotiikasta ja business intelligence ratkaisuista.

Tässä peittomatriisissa esittelen työn sisällöllisen kiinteyden eli tekstinsisäiset kytkökset (taulukko 1).

Taulukko 1. Peittomatriisi päiväkirjaopinnäytetyön tekstinsisäisistä kytköksistä

Oman ammatillisen kehittymisen tavoitteet	Tietoperustan luku raportissa	Seurantaviikko	Oman ammatillisen kehittymisen tulokset
Ohjelmistorobottien ohjaus	3.4, 3.5,	viikot 4, 5	
Automaatioiden raportointitoteutus	3.1, 3.2, 3.6,	viikot 1, 2, 6	

2 Lähtötilanteen kuvaus

Kuvaan tässä luvussa nykyistä työrooliani ja analysoin osaamistasoani lähtötilanteessa ja asetan kehittymistavoitteet itselleni. Esittelen myös työni sidosryhmät ja vuorovaikutustilanteet. Toimin ohjelmistorobotiikan kehittäjän tehtävissä Accountor Solutions Oy yhtiössä. Olen ollut tässä roolissa Accountorilla noin kaksi ja puoli vuotta. Aloitin Accountorilla ensin ohjelmistorobotiikan harjoitteleluohjelmassa ja tein aluksi analyytikon ja scrum masterin töitä, kunnes siirryin kehittäjäksi.

2.1 Oman nykyisen työn analysointi

Työtehtäviini kuuluu ohjelmistorobotiikkaprosessien kehittäminen eli muutokset olemassa oleviin prosesseihin ja uusien prosessien tekninen suunnittelu ja toteutus. Kehitystehtävissä minun tulee osata ohjelmoinnin peruslogiikka, käytössä oleva UiPath-tuote on .NET-pohjainen matalakoodisovellus. Koska robotti toimii käyttöliittymällä ja usein websovelluksissa, html-osaamisesta on hyötyä, jotta saa robotin löytämään juuri oikean elementin, jota sen tulee klikata sovelluksessa. Ohjelmistorobotin toimintaympäristö on hyvin laaja, se voi lukea ja lähettää sähköpostia, kirjoittaa exceliin ja toimia kuten käyttäjä sovelluksia klikkaamalla tai suoraan sovelluksen API-rajapinnan tai tietokannan kautta. API-osaamista olen tarvinnut useasti, kun robotin tulee joko lukea tietoja tai päivittää tai luoda niitä rajapinnan kautta sovelluksiin. SQL-kyselyjen ymmärtämistä tarvitaan haettaessa tietoa sovelluksista. Taustalla ajava ohjelmistorobotti toimii palvelimella joko konesalissa tai pilvessä, joten minun tulee hallita perusasiat palvelinympäristöistä.

Kehitystyön lisäksi vastaan Suomen organisaation osalta robottien teknisestä alustasta eli toimin RPA Technical Owner -roolissa. Tähän rooliin sisältyy robottikapasiteetista huolehtiminen eli uusien robottien tilaaminen, vanhojen eläköityminen, robottien ohjauksen ja robottikoneiden tarvittavat käyttäjätunnukset ja pääsy sovelluksiin sekä UiPath-tuotteen versiopäivitykset on-premise ympäristössä olevien robottien ja kehitystyökalun eli Studion osalta.

Olen hankkinut kehittäjän roolissa tarvittavia taitoja tietojenkäsittelyn opintojen ja päivittäisen prosessien koodaamisen kautta. Tiimimme Tech Lead sekä kehittäjäkollegat ovat tukeneet minua koodausongelmien ja sovellusten sekä palvelinympäristön haasteiden kanssa. Ammatillinen kehittymiseni koskien automaatiokehittäjän työtehtäviä on hyvällä tasolla. Pystyn itsenäisesti suoriutumaan kehitystehtävistä, osaan hakea tietoa ongelmanratkaisutilanteissa ja voin opettaa ja tukea muita kehittäjän tehtävässä. Olen loma-aikoina toiminut Tech Leadimme varahenkilönä.

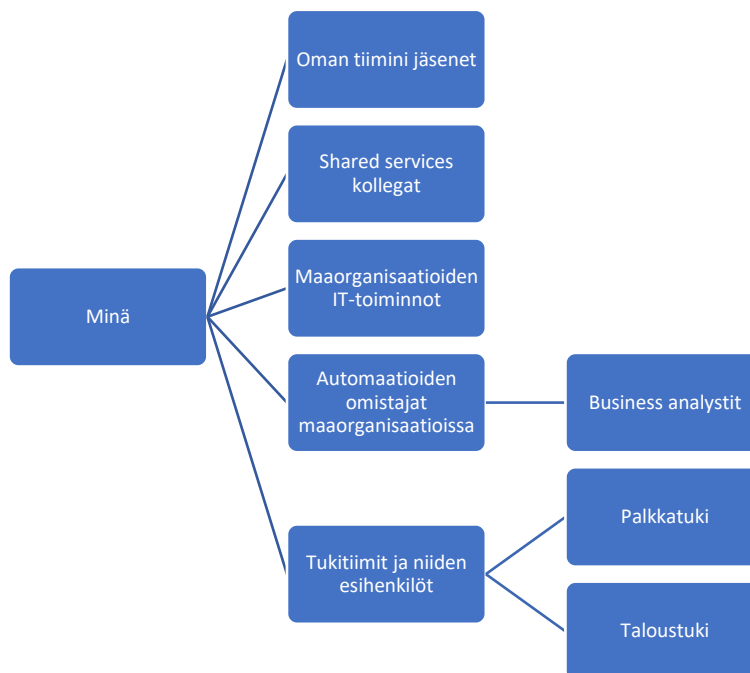
SaaS-siirtoprojektissa minulta vaadittava osaaminen liittyy ohjelmistojen tuntemiseen ja niiden toiminnan opetteluun. Olen valinnut kaksi ammatillisen kehittymisen teemaa, joiden toteutumista arvioin projektin aikana ja sen jälkeen. Nämä ovat:

- 1) Lisätä omaa ymmärrystäni UiPath Orchestratorin ominaisuuksista ja robottien ohjauksesta
- 2) Osallistua uuden raportointiratkaisun suunnitteluun ja oppia ottamaan sitä käyttöön

Projektin aikana ja jatkossa minun tulee panostaa syvälliseen ohjelmistorobotiikan ohjelmistojen tuntemiseen, jotta osaamiseni kokonaisarkkitehtuurista ja uusien teknologioiden, kuten GenAi, mahdollisuuksista kasvaa. Osaamisen laajentaminen nykyisestä roolistani kehittäjänä voisi kulkea kohti kokeneen asiantuntijan osaamistasoa ja jatkaa siitä kokonaisvaltaisen ohjelmistorobotiikka-arkkitehtuurin ymmärryksen ja osaamisen suuntaan. Tällaisia työtehtäviä voisivat olla RPA Solution Architect tai RPA Development Lead roolit. Valitsemani ammatillisen kehittymisen teemat auttavat minua kasvattamaan osaamistani, jotta siirtyminen seuraavalle tasolle mahdollistuisi noin 3–5 vuoden aikajänteellä.

2.2 Sidosryhmien esittely

2.2.1 Sisäiset sidosryhmät



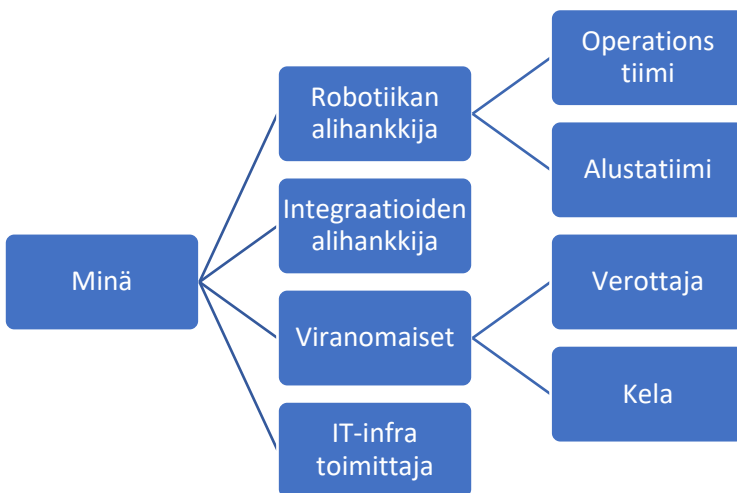
Kuva 1 - Sisäiset sidosryhmät työpaikalla

Sisäisiin sidosryhmiini kuuluvat oman “Processes and Automations”-tiimini jäsenet, joita on 7 henkeä mukaan lukien esihenkilömme. Oman tiimini sisältä saan tukea päivittäin. Olemme osa organisaatiomme seuraavan tason “Shared Services”-tiimiä, jossa on muun muassa it-infran asiantuntijat joiden apua ja tietoa tarvitsen verkkoyhteyksien kanssa. Eri maissa olevat IT-projektit

ja muutokset vaikuttavat minun työhöni automaatioihin tulevien muutosten kautta joten olen yhteydessä myös maiden IT-organisaation jäseniin ja johtoon.

Automaatioiden omistajat ovat maiden liiketoiminnassa ja teen yhteistyötä heidän tiimiensä kanssa, erityisesti business analyst -roolissa toimivien asiantuntijoiden kanssa, sillä he määrittelevät automatisoitavan kohteen lähtötilanteen ja tarvittavat muutokset. Päivittäinen automaatioiden valvonta on osaksi tukitiimien, palkkatuen ja taloustuen, vastuulla joten olen heidän kanssaan viikoittain yhteydessä erilaisten tuotannon ongelmien selvittelyn osalta. SaaS-siirtoprojekti koskettaa kaikkia sisäisiä sidosryhmiä ja tiedotamme projektin edetessä heitä.

2.2.2 Ulkoiset sidosryhmät



Kuva 2 - Ulkoiset sidosryhmät työpaikalla

Ulkoisiin sidosryhmiin lukeutuu ohjelmistorobotiikan alihankkija, joka tarjoaa jatkuvan palvelun eli toisen tason tuen tuotannon robotiikkaprosessien valvontaan sekä robottien nykyisen ohjauksen alustatuen, joka SaaS-siirto projektissa on tarkoitus korvata. Prosessien valvonta jatkuu alihankkijan toimesta edelleen projektin jälkeen. Sekä Operations-tiimi että Alustatiimi ovat mukana siirtoprojektissa. IT-infran toimittaja on projektissa tärkeä ulkoinen sidosryhmä koska he vastaavat tietoliikenneyhteyksistä robottipalvelimilta.

Muita sidosryhmiä työssäni ovat integraatioiden alihankkijan konsultti, joka tekee osana tiimiämme töitä. Lisäksi viranomaiset, tärkeimpänä verottaja ja Kela, joiden palveluita automaatioprosessit hyödyntävät, ovat ajoittain työni kannalta tärkeitä. Nämä sidosryhmät eivät osallistu SaaS-siirtoprojektiin, jota opinnäytetyöni kuvaa.

2.3 Työpaikan vuorovaikutustilanteet

Suurin osa vuorovaikutustilanteista, joita kohtaan työssäni on etänä tapahtuvia kontakteja. Puhumme oman tiimini kanssa päivittäin Scrum-tyyppisessä kehityspalaverissa Teamsin välityksellä. Tässä päivittäisessä kohtaamisessa on tarkoitus kertoa mitä kukin on tahoillaan saanut tehtyä, mitä on seuraavaksi tekemässä ja onko jotain ongelmia, joihin kaipaa apuja. Palaveriin on kutsuttu kaikki yhdeksän henkeä kehitystiimistämme. Näissä palaverissa on helppo pyytää apua tai tarkentaa jotain asiaa, joka on jäänyt epäselväksi. Tiimissä on aina avulias ilmapiiri ja matala kynnyks osoittaa tietämättömyytensä. Teemme kaikki niin laajan osaamisen tehtäviä, että kukaan ei yksin pysty hallitsemaan kokonaisuutta vaan tekniset ratkaisut muodostuvat aina parhaiten yhdessä niitä miettien.

Robottiikan toimittajan Operations-tiimin kanssa pidämme päivittäisen lyhyen tilannepalaverin etäyhteydellä. Käymme läpi tuotannon tilanteen ja onko jotain ongelmia virheiden ratkaisussa, joihin kaivataan apua, jaamme tehtäviä ja priorisoimme niitä. Haasteena tässä palaverissa on ohjata toimittajaa tekemään oikeita asioita ja tarttumaan kiireellisiin virheen selvityksiin ensin. Toisinaan tarvitsemme liiketoiminnalta vahvistuksen mihin ryhdytään ensin, jos tuotannossa on monta samanaikaista selvitettävää ongelmaa. Liiketoiminnan ihmisten kanssa vuorovaikutus voi välillä olla hidasta ja heidän voi olla vaikea käsittää miksi haluamme ohjeen tai kuvauksen ongelmasta mahdollisimman tarkalla tasolla ja DevOpsiin dokumentoituna. Jotta voisimme käyttää koodausaikaa tehokkaasti, niin tulee olla tarkasti selvillä mitä muutetaan tai korjataan koodiin. Tätä ei kannata tehdä moneen kertaan koska se tarkoittaa useaa eri testauskierrosta ja pahimmassa tapauksessa tuotannossa oleva automaatio lakkaa toimimasta.

Vuorovaikutustaidoissani on kehitettävää sen osalta, että muistaisin kertoa liiketoiminnan suuntaan, miksi ja miten kehitysprosessi etenee tai miksi joitain muutoksia kuten nyt Orchestratorin siirto SaaS:in tehdään ja mitä se tarkoittaa heille. Tämä ei ole heidän päivittäistä tekemistään, vaan heille ohjelmistorobotiikka näkyy lähinnä silloin kun jokin ei toimi, eivätkä he tiedä paljonko työtä ja millaisia vaiheita on uuden automaation tai Orchestratorin siirron takana. Ohjelmistorobotiikan osaajia on organisaatiossamme sisäisesti vain muutama ihminen, ja tämä osa-alue on aika tuntematon suurelle osalle ihmisiä, joten haluaisin kehittyä paremmaksi viestijäksi siitä, mitä oikeastaan teemme ja miksi se on tärkeää myös yrityksen markkinassa onnistumisen kannalta.

3 Seurantajakson raportointi viikkoanalyyseineen

Tästä alkaa SaaS-siirtoprojektin varsinainen seurantajakso, joka ajoittuu välille 12.08.2024-22.11.2024. Kerron projektin etenemisestä päivätasolla ja vertaan sitä tavoitteisiini. En sisällytä päivittäiseen seurantaan muita työtehtäviäni kuin projektin edistämiseen liittyviä tehtäviä. Tässä luvussa on mukana myös viikkoanalyysit jokaisen viiden päivän tarkastelujakson päätteeksi. Viikkoanalyysissa selvitän tarkemmin työvaiheita ja taustoja teorian kautta. Viikkoanalyyseja on yhteensä 8.

3.1 Seurantaviikko 1

Maanantai 12.08.2024

Päivätavoite: Muodostaa kokonaiskuva tällä hetkellä tuotettavista ohjelmistorobotiikan raporteista maittain ja automaatioprosessien tuottamasta datasta raportoinnin pohjana.

Päivätoteuma: Aloitin viikon selvittämällä, millaisia tietoja tällä hetkellä automaatioprosesseista raportoidaan ja aloin tehdä listaa kaikista Suomen, Ruotsin ja Norjan prosesseista. Oletuksena on, että löytyy erilaisia tapoja, miten raportointi on liitetty osaksi prosessia tai joissain prosesseissa sitä ei välttämättä ole lainkaan. Eri ikäisissä prosesseissa on käytetty erilaisia prosessikehyksiä, joten ne eivät ole kaikki samanlaisia myöskään raportointiosioden suhteen.

Tiistai 13.08.2024

Päivätavoite: Ymmärtää miten integraatioalustan raportointi toimii tällä hetkellä.

Päivätoteuma: Jatkoin selvitystä automaatioprosessien raportoinnista. Keskustelin Lead Integraatiokehittäjän kanssa, miten raportointi on toteutettu integraatioalusta Friendsissä. Prosessien ajot generoivat tiedot erilliseen tietokantaan, josta on tehty näkymät, joista PowerBi lukee tiedot ja visualisoi ne käyttäjille. UiPath Insights sovellus voisi tarjota osan nykyisistä raporteista suoraan pilvessä, niin ettei niitä tarvitsisi rakentaa erikseen maittain PowerBi:n vaan ne voisi tarjota pilvipalvelusta liiketoiminnan käyttöön. Tutkin millaisia raportteja Insights tuotteesta on saatavilla.

Keskiviikko 14.08.2024

Päivätavoite: Tutustua UiPath Insights tuotteen ominaisuuksiin ja verrata niitä organisaation tarpeisiin.

Päivätoteuma: UiPath Insights raportointisovelluksessa on rajoituksia. Insightsia ei tarjota ilmaisen pilvessä olevassa Community versiossa. Pilvestä tarjottavassa tuotteessa on Insightsin keräämälle datalle kahden vuoden säilytysaika. (Docs.Uipath, 2024) Rajoituksilla saattaa olla vaikutusta siihen, voidaanko Insights valita ohjelmistorobotiikan prosessien raportoinnin työkaluksi. Testausta ajatellen ilmaisversion puute aiheuttaa sen, että toimintojen kokeileminen on hankalaa ja täytyy luottaa dokumentaatioon ja käyttöohjeisiin sen sijaan että voisi testata tätä omassa ympäristössä. Kahden vuoden säilytysaikaa datalle ei voi ylittää, joten tulee arvioida, onko aika riittävä. Aloitin vertailemalla dokumentaatiosta mitä olemassa olevia raportteja voisimme korvata Insightsin ominaisuuksilla ja taulukoin näitä.

Torstai 15.08.2024

Päivätavoite: Syventää ymmärrystä Insightsin eri raporttien tiedoista ja niiden soveltuvuudesta organisaation käyttöön.

Päivätoteuma: Insightsissa on valmiiden muuttujien lisäksi annettu admin-käyttäjille mahdollisuus määrittää mukautettuja muuttujia, jotka on poimittu robottilokeista ja jonoista, joita käytetään tiettyjen KPI-arvojen laskemiseen tai kustomoitujen dashboardien rakentamiseen. (Docs.Uipath Insights, 2024) Näiden mukautettujen muuttujien avulla voidaan saada osa raporteista toteutettua, joita ei ole valmiissa raporttivalikoimassa.

Perjantai 16.08.2024

Päivätavoite: Edistää raportointitietojen keräämistä liiketoiminnalta.

Päivätoteuma: Esihenkilöni oli edellisellä viikolla lähettänyt kysymyksen liiketoiminnan edustajille eli Ruotsin, Suomen ja Norjan automaatioiden omistaja -roolissa toimiville henkilöille, että mitä nykyisistä PowerBi raporteista tarvitaan jatkossa. Vain Ruotsista on saatu vastaus tähän mennessä, joten muistutin muita vastaamaan kyselyyn, jotta pääsen omassa kartoituksessani eteenpäin.

Viikkoanalyysi 1 / viikko 33

Ensimmäisellä seurantaviikolla keskityin selvittämään raportoinnin nykytilaa ja käyttäjien vaatimuksia. Tämän lisäksi tutustuin UiPathin Insights tuotteeseen. Huomasin, etten ollut aiemmin ymmärtänyt kokonaiskuvaa eri maiden raportointitarpeista. Ruotsin osalta prosessien seuranta varten ei tarvita lainkaan prosessien transaktionäkymää, koska he seuraavat tätä suoraan Orchestratorin kautta. Tämä tarkoittaa, että Ruotsin osio voidaan jättää kokonaan pois uudesta raportoinnista.

Raportointi jakautuu nykyisellään kahteen eri osioon:

- 1) IA tracking eli analyysiraportit: automaatioiden säästämät tunnit, prosessien tehokkuus, prosessin säästämät uniikit minuutit, päivittäisten transaktioiden yhteenveto, automatisoidut tunnit vs. prosessin ajoaika, transaktioiden määrä ja tekninen onnistumisprosentti, prosessi-kohtainen käytetty kapasiteetti, ajojen kesto, ajojen tehokkuus.
- 2) RPA production tracking eli tuotannon seuranta. Prosessien transaktiokohtaiset suorituslokit sisältäen onnistuneet, liiketoimintavirheen takia epäonnistuneet ja teknisen vian takia epäonnistuneet transaktiot.

Molemmat osiot on nykyisin toteutettu PowerBi raporteina. Myös jatkossa tiedot prosessien transaktiokohtaisiin raportteihin pitää viedä PowerBi:n, jollei anneta kaikille pääsyä Orchestrator sovellukseen.

Business intelligence eli liiketoimintatiedon hallinta sisältää tiedon keräämisen, analysoinnin, jakamisen organisaatiossa ja tiedon hyödyntämisen eli sen perusteella tehtävät strategiset päätökset organisaatiossa. Liiketoimintatiedon hallinnan tavoite on auttaa organisaatiossa toimivia ihmisiä päätöksenteossa, jotta yritys menestyy markkinoilla. (Vitt & Luckevich & Misner, 2008, Luku 1). ETL eli extract/transform/load -prosessi kerää tiedot lähteestä, muuntaa ja muokkaa ne tietovaraston vaatimaan muotoon ja lopuksi vie tiedon tietovarastoon. Tiedon keräämisen suhteen tulee ensin päättää, mitä dataa tarvitaan ja miten se saadaan poimittua. Tietoa muokatessa saatetaan tarvita konversiota, datan siivoamista tai datan rikastamista. Kun tieto viedään tietovarastoon, täytyy määrittellä miten ja kuinka usein tietovarasto ottaa tiedon vastaan. (Loshin & Powell, 2003, Luku 10). Kun olemassa olevaan business intelligence -prosessiin tulee muutoksia järjestelmien osalta, tarjoutuu silloin myös mahdollisuus tarkastella uudelleen tiedon hyödyntämisen tarpeita.

Nykyiset Accountorilla käytössä olevat automaatioiden raportit on määritelty vuosina 2019–2021. Tuolloin on alettu keräämään ja tuottamaan tietoa automaatioista, samalla kun on rakennettu ensimmäisiä automaatiokohteita ja on tullut tarve seurata niiden tuomia hyötyjä. Automaatiokohteille on laskettu takaisinmaksuaika, eli kuinka paljon automaation rakentaminen maksaa ja miten paljon ammattilaisten työtunteja se säästää kuukaudessa. Pääsääntönä on ollut, että takaisinmaksuaika olisi alle vuoden. Automaation tekemisen ansiosta säästyy ihmisen tekemiä tunteja. Näitä säästettyjä tunteja on seurattu keräämällä dataa automaation ajokerroista, joihin on lisätty parametrina säästetyt minuutit ja viety tiedot visualisoitavaksi PowerBi:n, josta niitä on voitu raportoida liiketoiminnan päälliköille ja johdolle.

Siirryttäessä ohjaamaan ohjelmistorobotteja uudesta Automation Cloudista, automaatioiden raporttien tuottamiseen tarvittava tieto kerätään SaaS-palvelun rajapinnasta. Nykyisessä ratkaisussa tieto kerätään omassa palvelinympäristössä sijaitsevasta tietokannasta. Muutoksen myötä kä-

vimme ensin läpi, millaiset tarpeet liiketoiminnalla on jatkossa, jotta tietäisimme mitä tietoa tarvitsemme rajapinnasta. Liiketoiminnan tarpeet ohjaavat tiedon keräämistä ja varastointia ja asettavat rajat ja vaihtoehdot tekniselle ratkaisulle.

UiPath Insights on verkkosovellus, joka mahdollistaa tietojen mallintamisen ja analytiikan. Se käyttää alustalla olevaa dataa, josta tuotetaan valmiita raportteja, joiden lisäksi käyttäjä voi valita mittarit ja rakentaa omia raporttinäkymiään. (Docs.Uipath Insights, 2024). Ensimmäisen tarkastelun perusteella ainakin osa olemassa olevista PowerBi raporteista voisi olla toteutettavissa Insightsin avulla. Olin tyytyväinen oppimaani, koska olin alun perin oivaltanut, että kannattaisi tutkia jo olemassa olevan tuotteen mahdollisuudet, varsinkin kun se jo kuuluu lisenssiin, kuin että tekisimme saman asian uudestaan PowerBi raportille. Säästäisimme aikaa ja vaivaa.

3.2 Seurantaviikko 2

Maanantai 19.08.2024

Päivätavoite: Edistää SaaS-migraatioprojektin suunnitelmaa.

Päivätoteuma: Tiimin Tech Lead -roolissa toimiva kollegani palasi lomaltaan ja kävimme hänen kanssaan läpi tietoja mitä olen tähän asti saanut kartoitettua. Hänellä oli uutta tietoa lisättäväksi projektin suunnitteluun koskien SaaS-palvelun versiota.

UiPath Orchestrator tuotteen pilviversio tukee vain kolmea aiempaa robottien ja Studion versiota alaspäin. Tällä hetkellä pilvessä on uusin 2024.10 version, jonka tuki siis ulottuu 2023.4 versioon ja uudempiin. (Docs.UiPath Compatibility Matrix. 2024.) Uusi Orchestrator ei ole yhteensopiva meidän organisaatiomme nykyisten robottien version 2021.10. Tämä johtaa siihen, että meidän on lisättävä robottien päivittäminen osaksi projektia ja tämä lisää minun työmäärääni ja vaatii minulta suunnitelmaa, missä järjestyksessä robotit päivitetään.

Tiistai 20.08.2024

Päivätavoite: Raportointiratkaisun teknisen toteutuksen valinta.

Päivätoteuma: Keskustelimme Tech Leadin kanssa tarkemmin tavasta, jolla tuomme Orchestratorista tiedot raportointiamme varten. Päätimme että ensisijainen vaihtoehto on käyttää Orchestratorin API-rajapintaa ja hakea Friendsillä tiedot sieltä, tallentaa SQL-tietokantaan ja viedä sieltä visualisoitavaksi tarvittaville PowerBi-raporteille. Toissijainen vaihtoehto on rakentaa ohjelmistorobotiikka-prosesseihin koodikirjastoelementti, joka viedään jokaiseen prosessiin erikseen koodiin ja se vie tarvittavat tiedot Friendsille käsiteltäväksi Friendsin API-rajapintaa käyttäen.

Sovimme että teen suunnitelman robottien versiopäivityksen järjestykseksi, huomioiden tarpeen testata siirron aikana muutoksia robotiikkaprosesseihin. Nykyinen Orchestrator versio, jota robotiikkatoimittaja hallinnoi, on yhteensopiva uudempien robottiversioiden kanssa. Päätimme, että teemme robottien versionnostot ensin ja testaamme että prosessit toimivat uudella versiolla ennen kuin siirrämme Orchestratorin pilveen.

Keskiviikko 21.08.2024

Päivätavoite: Robottikoneiden päivityksen aikataulu ja osaprojektin suunnittelun saattaminen valmiiksi.

Päivätoteuma: Laadin suunnitelman ja aikataulun kaikkien neljän maan, Suomen, Ruotsin, Norjan ja Hollannin robottien sekä kehityskoneilla olevien UiPath Studio ohjelmien päivittämisestä 2023.10 versioon. Ensin päivittäisin Suomen kehitys- ja testiympäristöt sekä kolme tuotannon robottia. Suomessa on kaksi prosessia, joita ajetaan vain kuun vaihteessa eli pääsen jo elo- syyskuun vaihteessa testaamaan, että ne toimivat, joten tämä vie aikataulupainetta syys- lokakuun vaihteessa testata useampaa prosessia. Tämän jälkeen päivitän loput Suomen 9 robottikonetta tuotannossa. Kun nähdään että kaikki toimii, Hollannissa paikallinen automaatioista vastaava voi tehdä päivitykset ja minä jatkan Ruotsin Azuren testi- ja tuotantokoneiden päivityksen kanssa, jonka jälkeen päivitän Ruotsin on-premise-koneet. Norjan Azure-koneet päivitän viimeisenä ja tällä hetkellä on vielä avoinna mihin ajankohtaan Norjan on-premise-koneiden päivitys sijoittuu, koska heidän pitäisi siirtää nuo koneet vanhan palveluntarjoajan konesalista yhtiön uuden yhteistyökumppanin konesaliin. Siirto olisi hyvä tehdä ensin ja vasta sitten päivitys, jottei tarvitse tehdä asennuksia useaan kertaan.

Maanantai 26.08.2024

Päivätavoite: Aloittaa robottikoneiden päivitykset uuteen versioon.

Päivätoteuma: Aloitin päivittämällä yhden Suomen kehityskoneen Studion versioon 2023.10.8. Päivityksessä tuli eteen haasteita. Asennus vaatii admin-tason oikeudet, joten minun täytyi käyttää erillistä pääkäyttäjätunnusta. Minulla ei ole pääkäyttäjätunnusta Ruotsin tai Norjan ympäristöihin, joten seuraavaksi testasin, saanko normaalilla tunnuksella tehtyä päivityksen, se onnistui eli ympäristöissä on tämän osalta eroja. Avatessani automaatioprojektia Studioissa GIT-yhteys oli kadonnut. Koodirepositorio sijaitsee Azure DevOpsissa eikä uudesta Studio versiosta saanut enää yhteyttä sinne. Selvityksen jälkeen ilmeni, että uusi versio tarvitsee toimiakseen Microsoft Visual C++ 2015 Redistributable Update 3 RC paketin, jota koneella ei ole ollut. Asensin paketin ja sain GITin taas toimimaan ja uusimman koodin haettua repositoriosta. Huomasin että haettu projektipaketti kääntyi

uuteen Studio versioon ja siihen tuli automaattisesti päivityksiä, jotka olisi pitänyt lisätä koodiin. Tämä vaatii lisää tutkimista.

Tiistai 27.8.2024

Päivätavoite: Saada kaikki Suomen kehitys- ja testausrobotikoneet päivitettyä uuteen versioon ja aloittaa projektien testaaminen.

Päivätoteuma: Aloitin testaamalla kriittisiä automaatioprosesseja kehityskoneella. Ongelmaksi muodostui uuden Studio version pakottamat automaattiset muutokset koodiin. Jotta saan DevOpsin koodivarastossa olevan projektin version säilytettyä sen varalta, että siihen tarvitsee tehdä muutoksia, täytyy kaikki version 2023.10 muutokset tehdä uuteen koodihaaraan. Avatessani projektin Studiossa se pakotti kuitenkin tekemään muutoksia main-haaraan enkä pääsyt avaamaan repositorioon luomaani haaraa, johon olisin halunnut viedä muutokset. Yleensä muutokset voi perua, mutta tässä tapauksessa se ei toiminut.

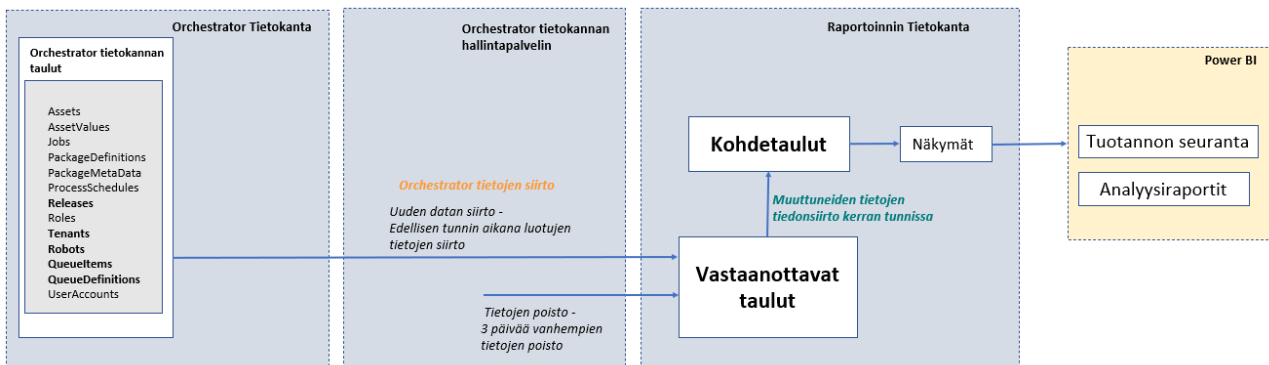
Hain koodin päivityshaaran erikseen remotesta Git Bashin avulla ja sain näin vietyä vain siihen muutokset ja main-haara jäi ennalleen. Tämä oli mukavaa vaihtelua koska sain muistella mieleen GIT Bashin käyttöä. Toinen tapa, joka myös oli toimiva, oli viedä vain paikallisesti main-haaraan muutokset ja vaihtaa päivityshaaraan, jossa sitten vein muutokset koodirepositorioon. Tässä on se vaara, että jossain kohtaa unohtaa main haarassa ollessaan tämän ja päivittää vahingossa muutoksen repon.

Sain kolme eri prosessia testattua onnistuneesti kehityskoneella ja jatkoin sen jälkeen päivittämällä testikoneen robotin ja minulla käytössä olevan kehityskoneeni Studion. Kävimme Suomen prosessiomistajien kanssa läpi projektin etenemisen ja sovimme että tiedotan tukitiimejä, kun tuotannon päivitykset roboteille tehdään. Tuotantokoneiden päivitys vaatii lyhyen noin puolesta tunnista tuntiin katkon. Tukitiimit valvovat robottien ajoja, joten heidän on oltava tietoisia katkosta ja päivityksestä, jos ongelmia ilmenee niin he tietävät mistä ne todennäköisesti johtuvat ja että meidän tiimissämme koitetaan niitä ratkoa. Teen tiketin DevOpsiin ja tiedotan sähköpostilla tiiminvetäjiä.

Viikkoanalyysi 2 / viikko 34 ja 35

Toisen raportointiviikon oivallukseni on, että automaatioiden raportointi kannattaa rakentaa käyttäen Orchestratorin rajapintaa, josta tiedot haetaan. Oli kuitenkin hyvä, että selvitin myös toisen vaihtoehdon rakentaa koodiin elementti, joka hoitaisi tätä ja totesin sen liian aikaa vieväksi. Koska en ole aiemmin tutustunut raportoinnin tekniseen toteutukseen niin selvitän, miten nykyinen raportointi toimii. Organisaatiomme Orchestrator-sovelluksen Azuressa olevasta tietokannasta haetaan hallintapalvelimella edellisen tunnin aikana luodut uudet tiedot määritellyistä tietokannan tauluista.

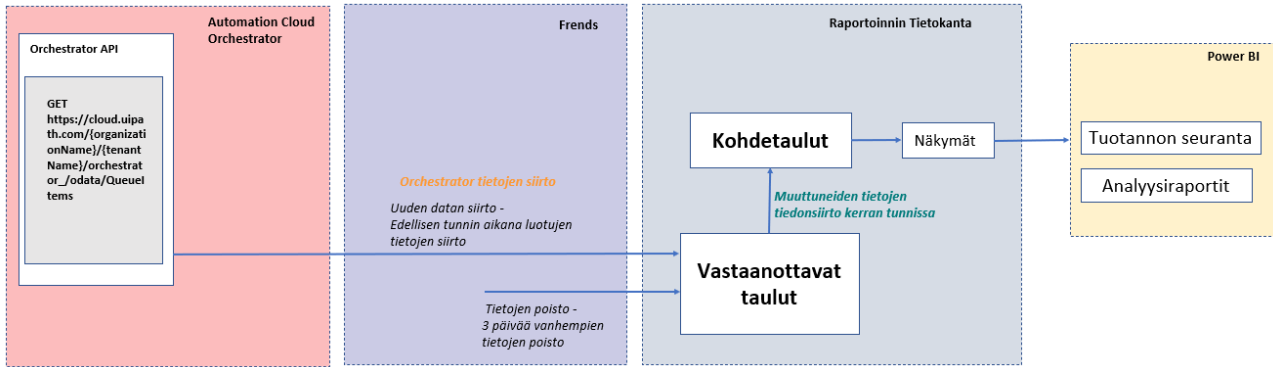
Tämä tieto viedään organisaation Azuressa olevaan raportoinnin tietokantaan, johon tulee tietoa muistakin sovelluksista. Tiedonsiirron on toteuttanut robotiikan alihankkijan alustatiimi ja se on heidän ylläpitämä ratkaisu. Tietokannassa on vastaanottavat taulut, joista uusi tieto siirretään kohde-tauluihin, joista on tehty näkymät, joiden tiedot viedään PowerBi:n kerran tunnissa.



Kuva 3 - Nykyisen raportoinnin toteutus

Kun otamme Orchestrator SaaS-palvelun käyttöön, niin UiPath omistaa ympäristön, josta palvelu tuotetaan. Ympäristö on usean asiakkaan kesken jaettu pilviympäristö, joten yrityksellämme ei ole pääsyä tietokantaan, josta voisimme hakea vanhaan tapaan tiedot tietokannan tauluista. Robotiikan toimittajan alustatiimi ei enää hallinnoi ja ylläpidä SaaS Automation Cloudia, vaan ylläpidon hoitaa UiPath ja hallinnoimme Orchestrator palvelua itse. Tämän vuoksi meidän on ratkaistava miten saamme korvaavan tiedonsiirron toimimaan jotta saamme edelleen tuotettua liiketoiminnan tarvitsemat analyysit ja raportit.

Tarkoituksena on hakea Orchestratorin ohjelmointirajapinnan eli API:n kautta tiedot. Tässä tapauksessa riittää, että saamme automaatioprosessien yksittäiset tapahtumat eli queueitemit haettua, joiden perusteella voimme rakentaa tuotannon seurannan ja osan analyysiraporteista PowerBi:n. Integraatiotyökalu Friends toimii hyvin tietokantakyselyiden suorittamiseen, joten rakennamme tietojen siirron prosessin sinne. Friends hakee rajapinnasta automaatiotapausten tulokset ja vie ne edelleen raportoinnin yhteiseen tietokantaan josta ne visualisoidaan PowerBi:n.



Kuva 4 - Raportoinnin toteutus Automation Cloudista

Liiketoimintatiedon hallinnan ratkaisun toteuttamiseen on monta eri vaihtoehtoa. Organisaatio voi ostaa valmiin pakettisovelluksen yhdeltä toimittajalta tai rakentaa kokonaan itse sovelluksen, jossa on räätälöity käyttöliittymä ja ETL-prosessi. Yleensä valitaan kuitenkin jotain näiden väliltä, koska on miltei mahdotonta löytää sellaista valmista BI-sovellusta, josta löytyisi kaikki tarvittavat ominaisuudet oletuksena, ilman että sitä täytyisi muokata organisaation omiin tarpeisiin sopivaksi. BI-ratkaisun teknisen toteutuksen osat ovat käyttäjille näkyvä käyttöliittymäkerros, tietokanta ja ETL-työkalut. Kaikki nämä voidaan hankkia erikseen. Käyttöliittymän tulee mahdollistaa tiedon sujuva saatavuus ja sen helppoon käytettävyyteen tulee kiinnittää erityistä huomiota. Tietokannan vaihtoehtoina voi olla OLAP tai relaatiotietokanta. Tietokanta on hyvin pitkäikäinen, jotta voidaan tehdä ajallisesti pitkiäkin analyysejä eikä tietokanta täten ole kovin helposti vaihdettavissa kuten käyttöliittymätyökalu olisi. ETL-työkalun täytyy pystyä lukemaan tietoa nopeasti useissa eri formaateissa, kuten excelistä, tiedostoista tai relaatiotietokannoista. (Vitt & Luckevich & Misner, 2008, Luku 10).

Ohjelmistorobotiikan raportoinnin lisäksi paneuduin toisella seurantaviikolla robottien versiopäivityksiin. Olen aiemmin vain asentanut uusiin robottikoneisiin UiPath-robottiohjelmiston ja päivitystoimenpiteet ovat minulle uuden oppimista osana SaaS-migraatioprojektia. UiPathin toteutus ja määritelmä robotille on agentti, joka mahdollistaa Studiossa rakennetun prosessin suorittamisen. UiPath Studion asennuksen mukana tulee aina mukana robotti, joka toimii paikallisesti koneella, johon asennus on tehty. Robotti avaa oletuksena Windowsin konsoli-istunnon ja ajaa Windows palvelua nimeltä Robot Service, joka suorittaa projektin eli automaatiotehtävät koneella, varmentaa kirjautuneen käyttäjän käyttöoikeudet koneella oleviin sovelluksiin ja verkko-osoitteisiin ja kommunikoi sovelluksen muiden osioiden kanssa suorituksen aikana. (Docs.UiPath Robot, 2024).

Jotta robotti voi ajaa automaatioprojektin, projekti on ensin luotava Studiossa. UiPath Studio on matalakoodinen sovellus, jolla voi toteuttaa yksinkertaisia sekä monimutkaisempia integraatioita ja valmisohjelmien automaatioita. Studiossa on valmiita koodiaktiviteetteja, joista rakentamalla muodostuu valmis työnkulku eli projekti, joka sitten tallennetaan Orchestratoriin ja jota ajetaan robotilla.

Valmis projekti voidaan julkaista Studiosta suoraan Orchestratoriin määriteltyä NuGet osoitetta käyttäen tai kirjoittaa projekti NUPKG tiedostoon ja ladata se manuaalisesti Orchestratoriin. (Docs.UiPath Studio, 2024)

Yleisemmin ohjelmistorobotiikassa erotellaan ”Attended RPA” ja ”Unattended RPA”. Attended on automaatio, jonka ohjaamiseen ihminen osallistuu erotuksena Unattended automaatiosta, jota ajetaan palvelimella taustalla ilman ihmisen toimia. RPA sovellukset eivät luo uutta ohjelmistoa vaan robotin ajaman koodin, joka ei tarvitse erillistä tietokantaa toimiakseen. Ihmisen ohjaama automaatiota kutsutaan myös työpöytäautomaatioksi ja taustalla toimivaa ohjelmistorobotiikaksi. Näistä seuraava askel on älykäs prosessiautomaatio eli ohjelmistorobotiikkaa ja muita teknologioita, kuten koneoppimista ja luonnollisen kielen ymmärtämistä, yhdistävä tietotyön automaatoratkaisu. (Langmann & Turi. 2022.) Keskityn työroolissani pääasiassa tekemään töitä ”Unattended RPA”:n kanssa, eli automaatioiden, joita ajetaan taustalla. Tarvitsen tietoa ja osaamista siitä, miten robotteja ohjataan ja ylläpidetään koska suurin osa automaatioista on taustalla toimivia ja niistä on isoin hyöty liiketoiminnalle. Robottien päivitys on yksi osa tätä osaamisen kerryttämistä.

3.3 Seurantaviikko 3

Keskiviikko 28.8.2024

Päivätavoite: Saattaa projektien testaaminen kehitys- ja testikoneilla valmiiksi.

Päivätoteuma: Testasin sekä päivitettyä ohjelmistopakettia että vanhaa pakettia testirobotilla ja molempien suoritus onnistui. Tämä vahvisti sen, että voin testata ensimmäisen tuotannon robotin päivitystä ja olettaa että vanha projektipaketti toimii siellä uudella robottiversiolla. En ajanut kaikille prosesseille erikseen testejä testirobotilla vaan ainoastaan valituille kriittisille prosesseille luottaen analogiaan, että jos yksi prosessi toimii niin vastaavat toisetkin ajavat onnistuneesti myös tuotannon roboteilla.

Sain kaikki suunnittelemani testit ajettua, joten päätin edetä tuotannon robottien päivittämiseen. Alkuperäisen aikataulun mukaan olin suunnitellut sitä tällä päivälle mutta uskoin että joudun siirtämään sen seuraavaan päivään, joka oli varalla. Lähetin tiedon tuotannon koneiden päivityksestä tukitiimeille ja heidän esihenkilöilleen. Päivitin ensimmäisen osuuden Suomen tuotannon roboteista onnistuneesti ja käynnistin prosessien ajot, jotta näin että ne toimivat.

Torstai 29.8.2024

Päivätavoite: Päivittää Ruotsin testi- ja kehitysrobotit.

Päivätoteuma: Olin jo aiemmin päivittänyt yhden robotin, joten jäljellä oli enää yksi testirobottikone, johon päivitin uuden version. Tämän lisäksi valmistelin Norjan osalta siirtoa heidän nykyisestä konesalistaan toiseen listaamalla prosesseissa käytössä olevat sähköpostiosoitteet, jotka vaihtuvat uuteen domainiin siirron myötä sekä käytetyt tiedostopalvelimet joihin pääsy täytyy ympäristön muutoksen jälkeen testata. Tämä liittyy robottien hallinnan pilveen siirtoon siten, että konesaliirto täytyy huomioida osana projektin aikataulua ja kytkeä uusi pilvessä oleva Orchestrator joko nykyiseen tai uusiin konesaliympäristön robottikoneisiin.

Perjantai 30.8.2024

Päivätavoite: Testata Ruotsin prosessit testi- ja kehitysroboteilla. Teen ohjeen muita kehittäjiä varten, siitä miten projekti avataan ja päivitetään uudessa versiossa Studiota.

Päivätoteuma: Ruotsin prosessien testaus vaati perehtymistä prosessien käsikirjoihin, joihin on koottu ohjeita, miten ajaa prosessia testiympäristössä. Valitsin prosessit, jotka on määritelty liiketoiminnan toimesta kriittiseksi ja joita ajetaan päivittäin sekä joitain prosesseja, jotka ovat medium tason kriittisyysluokituksella, asteikolla high-medium-low. On premise koneen testaamisen kohdalla jostain syystä robotti ei saanut yhteyttä orkestraattoriin, johtuen varmaan siitä, että käynnistin vahingossa päivityksen uudelleen ja se jäi jotenkin jumiin. Käynnistin päivityksen uudestaan koska tarkistin UiPath Assistantista version, joka oli 21 eikä 23. En tiedä miksi Assistant ei ollut päivittänyt, ajoin Change tyyppisen päivityksen, jonka jälkeen prosessin ajo toimi hyvin. Ruotsilla on varakoneena yksi kehitysrobotti, jota ei ole käytetty pitkään aikaan. Kun latusin päivityspakettia niin lataus kesti tunteja eikä minuutteja, joten päätin että koneelta puuttuvat päivitykset ja se on siksi niin hidas, joten tein tästä ongelmasta tiketin IT-infra toimittajalle.

Laadin ohjeen muille kehittäjille uuden Studion käytöstä mutten vielä jakanut sitä heille eteenpäin. Keskustelen heidän kanssaan maanantaina vielä asiasta ja varmistan että ovat ymmärtäneet mistä on kyse.

Maanantai 2.9.2024

Päivätavoite: Päivittää Ruotsin tuotantorobotit ja varmistaa että prosessit ajavat päivityksen jälkeen oikein. Jakaa ohje uuden Studio version käytöstä.

Päivätoteuma: Aamulla viestin Ruotsin automaatioiden omistajalle, että päivitän tuotannon koneet klo 10-15 välissä tänään ja siitä aiheutuu lyhyt katko robottiajoihin. Hän tiedotti liiketoimintaa asiasta. Päivityksen sujuivat odotetusti, ainoastaan yhdellä on-premise koneella UiPathin Chrome browser extension ei toiminut incognito tilassa, muutin asetuksen ja prosessi toimi sen jälkeen oi-

kein. Kävimme päivittäisessä ylläpitopalaverissa läpi ohjeen uuden Studion käytöstä. Sain vastauksen tikettiin koskien Ruotsin kehityskoneen verkko-ongelmia. Päivitykset olivat ajan tasalla eikä koneella ilmennyt mitään poikkeavaa, tällä kertaa myös paketin lataus onnistui normaalisti ja sain asennuksen tehtyä.

Torstai 5.9.2024

Päivätavoite: Päivittää Ruotsin kehitysrobotti ja Suomen tuotannon robotit.

Päivätoteuma: Suomen robottien toisen erän kanssa oli ongelmia johtuen hitaasta verkkoyhteydestä. Azuren virtuaalikoneilla verkkoyhteys oli todella hidas, joten minun piti löytää jokin toinen tapa saada asennuspaketti koneelle kuin lataaminen verkon yli. Sama ongelma oli aiemmin yhdellä Ruotsin kehityskoneella, mutta se korjaantui itsestään. Myös tuotannossa on erityisesti yöaikaan ollut verkkoyhteyden katkeamisia. Vein asennuspaketin verkkolevyille, josta sitten ensin robottitunnuksilla kirjautuen latusin sen koneen C-asemaan ja sitten admin-tunnuksilla kirjauduin tekemään päivityksen. Tämä oli todella paljon hitaampaa kuin verkon yli lataaminen. Verkosta lataamalla paketti oli koneella alle minuutissa, nyt paketin kopiointi ja kirjautumisen vaihtaminen kesti 10-20 minuuttia. Pienestä harmista voi aiheutua paljon odottelua ja lisätyötä ja minulla kului odotettua enemmän aikaa. Huomasin, että päivitys nollaa aina UiPath extension asetuksen Incognito modesta ja lisäsin sen kaikille koneille.

Viikkoanalyysi 3 / viikko 35 ja 36

Kolmannella projektin viikolla jatkoin edelleen pakettien testaamista päivitysten jälkeen. Selvitän tarkemmin, miten paketit muodostetaan ja miksi versionhallintaa tarvitaan ja miten se toimii. Lisäksi tarkennan, millaisia työkaluja käytämme organisaatiossamme versionhallinnan apuna, sillä ne kaikki liittyvät automaatioprosessien elinkaaren hallintaan.

NuGet on Microsoftin .NET pohjaisten pakettien luomiseen, säilyttämiseen ja käyttöön tehty työkalu. NuGet-paketti on nupkg.-päätteinen zip-tiedosto johon on pakattu projektin dynaamisesti linkitetty kirjastot, koodiin liittyvät muut tiedostot ja versionumero sekä kuvaus siitä mitä paketti sisältää. NuGet paketteja tarvitaan, jotta kehittäjät voivat luoda ja jakaa koodia helposti. Paketin voi julkaista joko julkiseen tai yksityiseen rajattuun käyttöön, josta sitten kehittäjä voi ladata paketin itselleen ja käyttää paketin ominaisuuksia osana omaa koodiaan. (Microsoft NuGet. 2024.).

UiPath Studioon voi ladata NuGet-paketteja käytettäväksi osana projektia useasta eri lähteestä. Julkisesti saatavilla ovat UiPathin viralliset "Official" kirjaston paketit, eli sellaiset koodipaketit, joita UiPath ylläpitää, kuten esimerkiksi Exceliin liittyviä tehtäviä automatisoiva koodi. UiPath ylläpitää

oman koodivaraston lisäksi käyttäjien omien pakettien jakamiseen tarkoitettua julkista "Marketplace"-nimistä online-kirjastoa, josta voi ladata muiden luomia koodipaketteja. Kukin UiPath ohjelmiston käyttäjä voi tallentaa yksityiseen käyttöön rajatut paketit paikallisesti omalle työasemalleen tai organisaation omaan rajattuun Orchestratoriin, josta niitä voi jakaa koodipaketteihin. (Docs.UiPath Studio. 2024. Managing activity packages.).

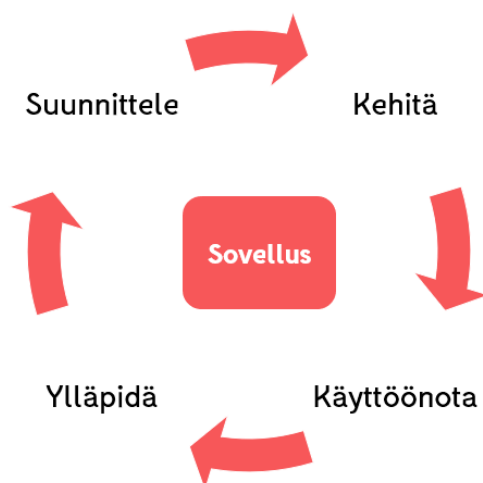
Versionhallinnan avulla kaikilla ohjelmistokehitystiimin jäsenillä on näkyvyys koodihistoriaan ja sama koodiversio pohjana omille muutoksilleen. Versionhallinnan työkalut pitävät huolta siitä, että uudet muutokset eivät yliaja toisia aiempia tai samanaikaisia muutoksia koodissa ja koodi pysyy eheänä ja käytettävänä. Versionhallinnan järjestelmät mahdollistavat tehokkaan yhteistyön ohjelmistokehitystiimissä. Koodi ei ole hukassa, vaan uusin versio kaikkien saatavilla yhteisestä koodikannasta, virheiden ilmaantuessa voidaan palauttaa edellinen versio koodista koodikannan kautta helposti ja kehitystiimin jäsenet näkevät kaikki toistensa tekemät pienimmätkin muutokset ja kommentit koodissa, joten tiedonjako tiimissä paranee. Versionhallinnan keskeisiä käsitteitä ovat:

- Repository eli varasto tai koodikanta johon tiedostot varastoidaan.
- Pull request eli pyyntö hyväksyä kehittäjän tekemät muutokset koodiin.
- Commit eli tilannekuva muutoksista, jotka viedään alkuperäiseen koodiin.
- Branch eli kehittäjän luoma koodihaara, jonka pohjana on alkuperäinen koodi ja tähän haaraan kehittäjä vie omat muutoksensa.
- Merge eli kehittäjän luoman koodihaaran muutosten lisääminen alkuperäiseen koodiin.

Versionhallinnan järjestelmä voi olla paikallisesti hallittava yhdelle käyttäjällä tarkoitettu tai useiden käyttäjien jakama mutta tietyllä palvelimella sijaitseva rajatun pääsyn keskitetty järjestelmä tai hajautettu versionhallinnan järjestelmä, joka mahdollistaa käytön mistä lokaatiosta tahansa. Yleisemmin käytössä ovat keskitetyt ja hajautetut järjestelmät. Git on avoimen lähdekoodin hajautettu versionhallinnan järjestelmä, jonka laaja suosio perustuu siihen, että kehittäjät voivat muokata samaa koodikantaa yhtäaikaisesti. SubVersion (SVN) ja Azure DevOps Server, joka aiemmin tunnettiin nimellä Microsoft Team Foundation Server (TFS), ovat keskitettyjä versionhallinnan järjestelmiä. (What is version control?. 2024.)

UiPath Studio tukee oletuksena sekä Git että SVN ja TFS versionhallinnan järjestelmiä. Git repositiorion sisällön voi hakea HTTPS tai SSH yhteydellä lokaalisti muokattavaksi ja tehdä haluttuun branchiin muutoksia ja viedä ne edelleen repositorioon. Studioissa on ominaisuus, jolla voi verrata lokaalin haaran ja repositoriossa olevan koodin eroja. (Docs.UiPath. 2024. Source control integration.) Azure DevOps on Microsoftin SaaS-alusta, jonka avulla voi hallita ohjelmistokehityksen ja ohjelmiston käyttöönoton tehtäviä. DevOps malli syntyi tarpeesta ratkaista perinteisen vesiputousmal-

lin muodossa tehtyjä ohjelmiston kehityksen, jakelun ja testaamisen välisiä ongelmia. Ohjelmistokehittäjät loivat ohjelmistoa, jonka he antoivat eteenpäin ohjelmistoa ylläpitävälle alustatiimille eikä kummallakaan osapuolella ollut tarpeeksi tietoa toisen tiimin toiminnasta tai vaatimuksista. Kun nämä yhdistetään yhdeksi tiimiksi ja lisätään mukaan laadunvarmistus, niin saadaan ketterä tiimi, joka voi toteuttaa asiakkaan vaatimusten mukaisia osia ohjelmistosta ja asiakas on lähempänä toteutusta ja voi vaikuttaa siihen jokaisessa vaiheessa. Tavoitteena on saada nopeammin ja tehokkaammin tehtyä paremmin asiakkaan tarpeisiin vastaava ohjelmisto.



Kuva 5 - Sovelluksen elinkaaren vaiheet (mukaillen Demiliani & Malik & Zaal 2020, Introducing Azure DevOps key concepts)

Suunnitteluvaiheessa tiimin käytössä Azure DevOps työkalussa ovat Kanban taulut ja backlogit joihin viedään työlistat toteutuksen eri osista. Kehitysvaiheen osalta voidaan käyttää koodityökaluja sekä tallentaa koodi koodivarastoon, josta se on tiimin jäsenten saatavilla. Käyttöönotto eli sovelluksen jakelu voidaan hoitaa Azure Pipelinen avulla julkaisemalla sovellus pilvi- tai on-premise ympäristöön. Ylläpidon avuksi Azure DevOps tarjoaa valvontaa sekä pilviympäristöön tietoturvaa. (Demiliani & Malik & Zaal, 2020. Azure DevOps overview). Organisaatiomme on ottanut tästä laajasta Azure DevOps tarjoamasta käyttöön suunnittelun osalta Kanban boardit sekä backlogit ja kehitysvaiheen osalta koodivaraston eli repon.

Azure DevOps tukee kahta versionhallinnan tapaa; hajautettua versionhallintaa Gitin avulla sekä keskitettyä versionhallintaa Team Foundation Version Control eli TFS, nykyisin nimeltään Azure DevOps Serverin kautta. (Demiliani & Malik & Zaal, 2020. Handling source control with Azure DevOps). Kuten päiväkirjamerkinnoistä käy ilmi, olemme valinneet hajautetun versionhallinnan eli Gitin käytön, jotta tiimin jäsenet saavat joustavasti käsiteltäviä koodihaaroja ja useita muutoksia automaatioprosesseihin.

Luomalla Gitin koodihaaran, ohjelmistokehittäjä voi eriyttää omaksi kehityslinjaksi juuri sillä hetkellä työstämänsä muutoksen koodiin. Haara perustuu juuri sen hetken tilannekuvaan koodista ja kehittäjät voivat jokainen luoda oman haaransa, joita on samanaikaisesti elossa useita ja jotka sitten myöhemmin liitetään yhteen. Uuden koodihaaran luomiseen voi olla useita eri syitä, jotka ovat organisaatiossa yhdessä sovittuja toimintatapoja enemmän kuin teknisiä rajoitteita. Koodihaarat voivat edustaa sovelluksen eri kehitysvaiheita, kuten testiversiota ja tuotannon versiota erikseen tai koodihaaraa voidaan käyttää eri tuotannon versioiden julkaisemiseen, siten että aiempi versio, joka on käytössä osalla asiakkaista, säilyy edelleen. Koodihaaraa käytetään usein eristämään korjaus ja sen testaus jo tuotannossa olevasta koodista tai koodihaara voi olla myös kehittäjäkohtainen, niin että jokaisella saman projektin kanssa työskentelevällä on oma haaransa. (Loeliger & Ponnuthurai, 2022, Luku 3.).

Kun koodihaaraan tehty muutokset on saatu valmiiksi, jonkun tulee kontrolloida, miten ne liitetään osaksi päähaaraa koodissa. Kehittäjä tekee Azure Devopsiin pull request tyyppisen pyynnön tarkastaa ja hyväksyä muutokset, jotta hän voi liittää ne osaksi koodia. (Chandrasekara & Herath, 2020, Luku 8.). Seuraamme päivittäisiä tehtäviä DevOpsin boardeilla. Tehtävän voi linkata pull requestiin jolloin Tech Lead, joka tarkastaa koodimuutoksia tai uutta koodia, näkee myös mihin automaatioon tai muutospyyntöön kyseessä oleva koodimuutos liittyy. Tähän mennessä automaatioprojekteihin ei ole tarvinnut tehdä muutoksia versionnoston tai Orchestratorin siirron takia, ainoastaan kun testasin että projektit toimivat myös päivitettyinä versioina. Näistä en kuitenkaan tehnyt lainkaan pull requestia enkä liittänyt olemassa olevaan koodihaaraan muutoksia.

3.4 Seurantaviikko 4

Perjantai 6.9.2024

Päivätavoite: Seurata että Suomen tuotannon roboteilla ajot onnistuvat.

Päivätoteuma: En havainnut ongelmia robottiajoissa. Olin saanut keskiviikkona kirjautumislinkin uuteen SaaS testipuolen Orchestrator palveluun. Ohjelmistorobotiikan toimittajan alustatiimi oli saanut uuden ympäristön luotua ja avattua meille tunnukset. Kirjauduin sisään ja ihmettelin miksi UiPath Automatio Cloud -palvelussa ei näkynyt lainkaan Insights-tuotetta. Selvisi että käyttäjätunnuksellani ei ole admin-tason oikeuksia ja että ne lisättäisiin, jolloin kaikki enterprise-tason lisenssiin kuuluvat tuotteet tulisivat näkyviin ja pääsisin tutkimaan tarkemmin. Sainkin oikeuden mutta en ehtinyt vielä perjantaina aloittaa tätä tutkintaa.

Tiistai 10.9.2024

Päivätavoite: Vahvistaa aikataulu Norjan robottien päivitykselle ja tutustua uuteen testi-Orchestratoriin.

Päivätoteuma: Tavoitteeni eivät toteutuneet koska Norjan kehityspuolen robottikoneiden kanssa oli ongelmia. Yhdelle koneelle ei päässyt lainkaan kirjautumaan ja toinen oli varattuna Operations-tiimillä tuotannon ongelmien selvityksien takia. Koska en tiennyt milloin nämä ongelmat ratkeavat, en voinut lyödä lukkoon päivitysaikataulua. Norjan automaatioiden omistaja vahvisti, että jos pilvessä olevat koneet voidaan päivittää tämän viikon aikana, niin päivitys voidaan tehdä, muutoin ei. En ehtinyt kirjautua uuteen testi-Orchestratoriin ja tutkia sitä. Pidimme statuspalaverin robotiikan alihankkijan alustatiimin kanssa ja sovimme että he siirtävät nykyisen testi-Orchestratorin datan ensi viikon maanantaina uuteen testipuolen SaaS Orchestratoriin. Keskustelimme mitä ominaisuuksia Automation Cloud lisenssiin kuuluu uutena, toimittaja lupasi selvittää asiaa ja palata.

Keskiviikko 11.9.2024

Päivätavoite: Suomen päivitysten loppuun saattaminen ja Norjan päivityksen aloittaminen.

Päivätoteuma: Vein jo tehtyjä ja suunniteltuja tehtäviä DevOpsin boardille, jotta saisin paremmin seurattua niitä ja delegoitua osan tiimissämme aloittaneelle harjoittelijalle. Sovimme että harjoittelija tekee listan nykyisistä käyttäjätunnuksista, rooleista ja ryhmistä Orchestratoreissa, jotta sitten saamme ne sujuvasti siirrettyä uuteen SaaS ympäristöön. Siirto täytyy tehdä käsin koska kaikki tiedot eivät kopioidu vanhasta ympäristöstä. Päivitin Suomen viimeisen kehitysrobotin sekä ensimmäisen Norjan kehitysrobotin ja testasin prosessia, joka ajoi onnistuneesti. Viestin Norjaan, että teen päivityksen ja jos se onnistuu, niin päivitän torstaina myös tuotannon koneet Azuressa. Norjalla on sekä Azure että on-premise robottikoneita, joista on-premise robottikoneet on tarkoitus siirtää Azureen syyskuun aikana. Edelleen on avoinna, päivitetäänkö versio ennen sitä vai ei.

Torstai 12.9.2024

Päivätavoite: Päivittää Norjan tuotannon Azure koneet.

Päivätoteuma: Sain Norjan koneet päivitettyä ja siivosin Norjan nykyisen Orchestratorin tietoja, jottei tarpeettomia arvoja vietäisi SaaS:iin kun siirto tehdään.

Perjantai 13.9.2024

Päivätavoite: Tutustua testipuolen uuteen Orchestratoriin.

Päivätoteuma: Aioin kytkeä yhden kehitysrobottilisenssin SaaS Orchestratorista omaan testikoneeseen, kunnes tulin miettineeksi, että ehkä onkin sovittu, että robotiikkatoimittaja lisää robottikoneet

Orchestratoriin ensi maanantaina. Tech Lead kertoi, että he tekevät tämän, joten en alkanut lisätä mitään tietoja, varsinkaan kun ei ole varmuutta siitä, säilyvätkö tiedot migraatiossa vai ylikirjoit- taako vanhan Orchestratorin tietojen tuonti nämä arvot.

Uudessa Automation Cloud Orchestratorissa on uusia ominaisuuksia. Prosessien triggereissä, eli ajastuksissa, jotka laukaisevat automaatioprosessin ajon robotilla käyntiin, on lisättyä aika- ja jo- notriggereiden lisäksi tapahtuma- ja API-triggerit. Monitoroinnista löytyy robottikoneiden utilisaation näkymä sekä jonoista uusi näkymä, jossa on eriteltyä robottien ja jonotapausten käsittelyä. Näistä uusista näkymistä saattaa olla hyötyä päivittäisessä valvonnassa.

Viikkoanalyysi 4 / viikko 36 ja 37

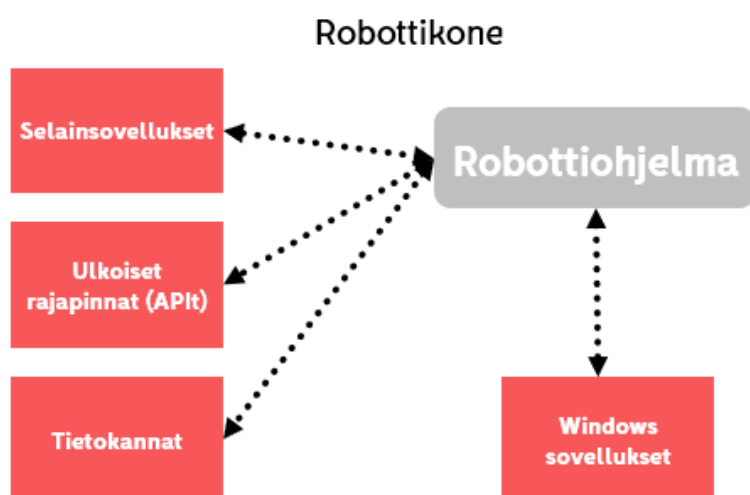
Olen tehnyt päivityksiä eli versionnostoja roboteilla, joten oman oppimiseni kannalta selvitän, miksi tämä on tarpeellista ja mitä toimenpiteitä ja millaista osaamista päivitys vaatii minulta. Koska kaikki ohjelmistorobotit ovat organisaation itse hallinnoimassa ympäristössä eikä tätä osuutta ole ostettu pilvipalveluna tai ylläpitopalveluna joltain toimittajalta, niin jonkun täytyy sisäisesti huolehtia, että robotit päivitetään säännöllisesti, jotta voimme hyödyntää UiPath ohjelmiston uusia ominaisuuksia eivätkä vanhat lakkaa toimimasta vaan ovat yhteensopivia muiden sovellusten kanssa.

RPA-ympäristön päivittäminen tulee suunnitella huolella, jotta automaatioiden päivittäinen toiminta ei katkea. Robotiikkaohjelmiston toimittajan versionjulkaisun tiedot tulee tarkistaa, jotta ymmärretään riippuvuudet siihen, miten uusi versio toimii nykyisessä ympäristössä. Käytännössä erot tulevat esiin, kun versiota testataan tuotantoa mahdollisimman hyvin simuloivassa testiympäristössä. Versiopäivityksellä voi olla vaikutuksia nykyisten sovellusten automaatioon, selainten toimintaan, tietokantojen tai käyttöjärjestelmän yhteensopivuuteen. Jos versiopäivitys tuotantoon ei onnistu- kaan, tulee suunnitella, miten aiempi versio palautetaan. Jotta päivityksellä olisi mahdollisimman vähän vaikutusta päivittäisiin automaatioajoihin, voidaan käyttökatko ajoittaa viikonlopulle tai jos se joudutaan tekemään arkena, tulee kaikkia sidosryhmiä tiedottaa katkon alusta ja sen loppumisesta. Versiopäivitys voidaan vaiheistaa esimerkiksi maittain tai tehdä pilottina päivitys ensin rajattuihin prosesseihin. Uuden version ominaisuudet on koulutettava tai tiedotettava kaikille RPA-ympäristöä käyttäville. (Sahgal, 2023, Luku 17).

UiPathin tuotteet kannattaa päivittää tuettuun versioon, sillä ohjelmistotoimittaja ei takaa, että vanhat versiot toimivat eikä toimita niihin korjauksia. Jos kuitenkin edelleen käyttää vanhentunutta versiota, on riskinä, että se ei enää toimikaan liiketoiminnan sovellusten kanssa, jolloin automaatiota ei voida ajaa. Ajantasaisen version myötä käyttäjä voi siis luottaa, että mahdolliset viat korjataan ja lisäksi tuotteen uudet ominaisuudet ovat saatavilla. UiPath tukee versioita aina kolme vuotta eteen- päin version julkaisuajankohdasta. Versioissa voi olla ominaisuuksia, jotka vanhenevat eli niitä ei

enää tueta. UiPath listaa näitä vanhentuvia ominaisuuksia tuotteittain. Vanhentuvien ominaisuuksien lisäksi tulee siis huomioida vanhentuva versio ja suunnitella on-premise tuotteiden päivitys tehtäväksi vähintään kolmen vuoden välein liiketoimintariskin poistamiseksi. Yksi mahdollisuus on siirtää päivitys UiPathin huoleksi ja siirtyä on-premise tuotteiden sijaan käyttämään pilvipalveluja, joissa on aina uusin versio. (Kannan, 2023.). Orchestrator SaaS-siirto projektissa on kyse tästä, siirrämme osan robotiikan ympäristöä pilveen, vaikka kehitys- ja tuotantorobotit edelleen ovatkin omassa ympäristössämme. Tämänhetkinen Orchestrator pilviversio on yhteensopiva kolmen edellisen Robot ja Studio julkaisun kanssa. Robot ja Studio julkaisut on viimeksi päivitetty vuonna 2021 ja ne ovat nyt versiota 2021.10. Valitsimme nostaa ne versioon 2023.10 uusimman on-premise version ollessa 2024.10. UiPath on poistanut Studiosta Windows – Legacy projektien luomisen mahdollisuuden sekä ”classic design experience” näkymän versiossa 2024.10 ja tuettuna on vain Windows projektit sekä ”modern design experience” (Docs UiPath. 2024. Deprecation.). Tämä aiheuttaisi kaikkien projektien testaamisen ja muuntamisen Windows projekteiksi sekä modernin kehitysnäkymän opettelun. Emme halunneet yhdistää näitä tähän projektiin, joten valitsimme aiemman version 2023.10 välttääksemme tämän työn, joka on edessä viimeistään vuonna 2026.

Unattended RPA eli tausta-automaatio tarvitsee toimiakseen palvelimen, jolla robottiohjelmaa suoritetaan. Organisaation palvelinympäristön hallinnointi on IT-infratiimin vastuulla, joten heidän apuaan tarvitaan robottipalvelimien pystyttämiseen, päivittämiseen ja mahdollisiin muutoksiin. IT-infratiimi voi olla ulkoistettu toimittajalle tai tuotettu itse sisäisenä palvelua. Robottiohjelman ominaisuudet asettavat vaatimukset robottipalvelimelle, joka voi olla pilvessä oleva virtuaalikone tai konessa oleva on-premise palvelin.



Kuva 6 - Unattended robottiarkkitehtuuri (mukaillen Sahgal, 2023, luku 4.)

Tausta-automaatiota ajavat unattended-robotit voivat ajaa läpi vuorokauden pysähtymättä. Koneelle asennettu robottiohjelma suorittaa tehtäviä koneelle asennetuissa tai koneen selaimen kautta toimivissa sovelluksissa käyttöliittymällä. Ohjelma voi ajaa automaatioita myös API-rajapintojen avulla tai suoraa tietokantoihin. (Sahgal, 2023, luku 4.).

Robot Service on tuote, joka yhdistää UiPath Studion, Assistantin ja lähettää käskyn Robot Executorille suorittaa prosessin sekä kommunikoi Orchestratorin kanssa. Service Mode Robot service on tarkoitettu tausta-automaatioiden ajoon ja se suorittaa Windows palveluja järjestelmänvalvojan oikeuksilla ja ottaa Orchestratorilta vastaan käskyn taustalla aloittaa prosessin ajaminen ilman että käyttäjä olisi kirjautuneena koneelle. Robot Executor ajaa Windows koneella joko Windows - Legacy -projektin .NET 4.6.1. ohjelmistokehyksessä tai Windows-projektin .NET 6 kehyksessä. (Docs.UiPath. 2024. Robot Service, Robot Executor.)

Samalla koneella olevat UiPath Studio ja Robot on oltava samassa versiossa. Päivitykseen tarvittava UiPathStudio.msi asennuspaketti voidaan ladata UiPath Resource Centeristä verkosta. Robotti päivitetään käynnistämällä asennuspaketin ohjattu asennusohjelma robottipalvelimella. Studion päivitys tehdään samalla msi-asennuspaketilla, jolloin valitaan oletuksena tulevien Robot ja Assistant ohjelmien lisäksi Studio ohjelma asennettavaksi. (Docs.UiPath. 2024. Updating the robot, Update Studio.)

3.5 Seurantaviikko 5

Tiistai 17.9.2024

Päivätavoite: Konfiguroida testipuolen uutta Orchestratoria Suomen osalta.

Päivätoteuma: Päätimme että lisäämme itse testirobotin ja kehityskoneet ja liitämme ne Orchestratoriin sen sijaan että alustatiimi tekisi tämän valmiiksi. Katsoimme asetukset läpi yhdessä Tech Leadin kanssa ja sain oman kehityskoneeni liitettyä SaaS Orchestratoriin ja ajettua yhden testitapauksen jonoihin. Pidimme statuspalaverin robotiikan toimittajan alustatiimin kanssa ja keskustelimme tuotannon Orchestratorin siirtämisestä SaaSiin ja tehtävien jaosta. Päätimme, että teemme myös tuotannossa itse robottien lisäämisen ja liittämisen Orchestratoriin. Koska siirtoon liittyy paljon testaamista ja prosessit pitää pysäyttää joksikin aikaa, on meidän helpompaa hallita koko aika-tilaa itse. Voin silloin päättää prosessi kerrallaan, milloin kytken sen vanhasta ympäristöstä pois ja siirrän ohjattavaksi uudesta ympäristöstä. Ajojen raportoinnin kannalta ajoituksella on myös merkitystä, ettei samasta prosessista tule molemmista ympäristöistä raportoitavaa vaan tällä on selkeä päivämääräraja, jolloin vanha raportointiratkaisu katkeaa ja uuden käyttäminen alkaa.

Keskiviikko 18.9.2024

Päivätavoite: Liittää Suomen kehitysrobotit uuteen Orchestratoriin ja käydä läpi raportointiratkaisun teknisen toteutuksen eteneminen.

Päivätoteuma: Pidimme robotiikkatoimittajan alustatiimin kanssa palaverin, jossa selvitimme käyttäjärooleja, joita uudessa Orchestratorissa on verrattuna vanhaan. Roolit ovat muuttuneet ja Tech Leadin kanssa halusimme ymmärtää miten oletusroolit pitäisi asettaa kehityskoneiden käyttäjille ja testirobotille, jotta ne toimivat parhaalla mahdollisella tavalla. Löysimme tähän ratkaisun mutta kun testasin testirobotilla tuotannon prosessin ajoa niin se epäonnistui. Prosessin sisällä luodaan jono, eikä robotille ollut annettu tällaista oikeutta, joten prosessin suorittaminen kaatui virheeseen. Huomasimme että meidän täytyy luoda ja lisätä erillinen rooli robotille tätä varten. Sain kehityskoneiden käyttäjät luotua ja liitettyä ne Orchestratoriin ja testasin kehityskoneilta prosessien ajoja onnistuneesti.

Keskustelimme aikataulusta ja päätimme Tech Leadin kanssa, että etenemme muiden maiden testipuolen Orchestratoreiden siirtojen kanssa koska Suomen osalta testausta voidaan vielä jatkaa kahden viikon ajan ennen tuotantoon siirtoa ja Ruotsin osalta aikaa on myös koko seuraava viikko. Olemme hyvin aikataulussa jollei tule mitään yllättävää. Aikataulun osalta raportointiratkaisun toteutus on hieman myöhässä. Pidimme palaverin integraatiokehittäjämme kanssa Friendsillä kehitettävän raportoinnin automaation tilanteesta ja siinä oli vielä avoimia asioita ennen kuin prosessi voitaisiin viedä nykyiseen tuotantoympäristöön. Näytti siltä, että UiPathin Orchestrator API-rajapinnasta ei tulisi lainkaan tietoa siitä, milloin jonotransaktiolla on tehty muutos eli LastUpdated arvoa ei voisi käyttää hakukriteerinä. Asia jäi vielä selvitykseen ja keskustelimme, että viimeinen aikaraja saada raportointi toimimaan on 30 päivää siitä, kun on siirrytty käyttämään SaaS Orchestratoria, koska sen verran jonotransaktiot säilyvät tietokannassa pilvessä. Pahimmassa tapauksessa voimme olla ilman raportointia 30 päivää koska tämä ei ole liiketoimintakriittinen osuus.

Torstai 19.9.2024

Päivätavoite: Konfiguroida Ruotsin ja Norjan testi- ja kehitysrobotit uuteen Orchestratoriin ja aloittaa näiden testaus.

Päivätoteuma: Huomaan että olisin kaivannut tarkempaa suunnitelmaa siitä, kuka tekee mitään projektissa ja tehtävien jaettua seurantataulukkoa. Olen aiemmalla urallani tehnyt projektiluontoista työtä ja tottunut siihen, että asiat suunnitellaan ennalta, jottei hukata asiakkaan aikaa. Nyt toimiesani organisaation sisäisessä roolissa käytännöt eroavat tästä. Lisäsin itselleni tehtävän laatia tarkka suunnitelma tehtävistä ja aikataulusta tuotannon Orchestratorin maakohtaisten siirtojen osalta. Voimme siten nähdä paljonko tehtävää on ja jakaa sitä tarvittaessa useamman tekijän kesken.

Norjan osalta ei vielä siirrettykään vanhan Orchestratorin kantoja uuteen, vaan tämä tehdään vasta seuraavalla viikolla koska oli epäselvyyttä siitä, kytkeemmekö myös on-premise testaus ja kehityskoneet uuteen ympäristöön. Sovimme että tätä ei tehdä, vaan kytkemme uudet Azureen lisättävät koneet SaaS-palveluun eikä vanhan on-premise ympäristön osalta lisätä lainkaan koneita SaaSiin. Epäselvyys suunnitelmien suhteen aiheuttaa epävarmuutta ja turhaa työtä. En päässyt myöskään vielä testaamaan Ruotsin Orchestratoria koska priorisoin muita tehtäviä.

Perjantai 20.9.2024

Päivätavoite: Konfiguroida Ruotsin testi- ja kehitysrobotit uuteen Orchestratoriin ja aloittaa näiden testaus.

Päivätoteuma: Lisäsin asetukset käyttäjien ja robottien osalta ja liitin testirobotit ja kehityskoneet SaaS Orchestratoriin. Jouduin jättämään yhden Ruotsin kehityskoneen liittämättä, koska Operations tiimissä tehtiin testausta tuotannon virheiden selvittämiseksi. Jatkan ensi viikolla tämän tehtävän loppuun.

Tiistai 24.9.2024

Päivätavoite: Liittää Norjan kehityskone SaaS Orchestratoriin ja aloittaa työsuunnitelman tekeminen tulevasta Ruotsin tuotannon siirron vaiheista.

Päivätoteuma: Norjan kehityskoneen liittämisessä oli useita ongelmia ja huomasin että robottitunnukset olivat menneet ristiin, joten yhdistäminen ei sen takia onnistunut. Lisäksi kun olin poistanut ja luonut käyttäjän uudestaan, oli tietokantaan jäänyt arvo kummittelemaan enkä saanut asetuksia tehtyä ja käyttäjätiliä toimimaan niin että robottikone olisi yhdistynyt Orchestratoriin. Lopulta onnistuin päivittämällä käyttäjänimeä ja ilmeisesti se päivitti jotain taustalla koska tämä korjasi ongelman.

Listasin Ruotsin osalta kaikki aktiiviset prosessit, niiden ajoitukset sekä prioriteetit ja virtuaalikoneet, joilla robotit ajavat. Tämän perusteella tein kolmiosaisen luokituksen siitä, mitkä koneet ja prosessit kannattaa ensin siirtää SaaS Orchestratorin hallittavaksi ja mitkä viimeisenä. Mietin myös mitä tehtäviä ja missä vaiheessa ne toteutetaan ja listasin näitä ylös, jotta voimme sopia kuka tekee mitäkin, jos aikataulu menee tiukaksi enkä ehdi tehdä kaikkea itse.

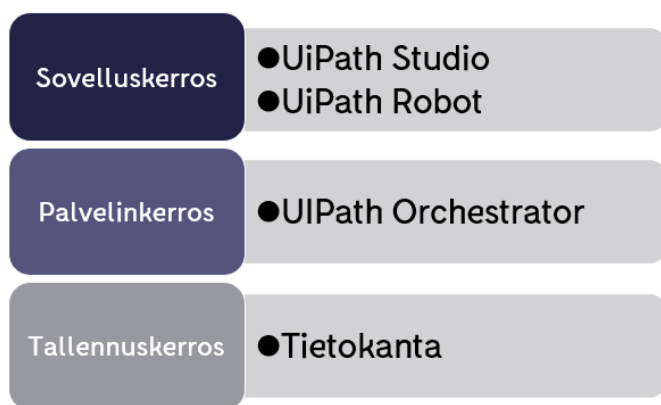
Viikkoanalyysi 5 / viikko 38 ja 39

RPA arkkitehtuuri on mahdollista toteuttaa monella eri tapaa. Accountorilla nykyisessä organisaatiossani on valittu rakentaa RPA toimimaan organisaation omistamaan laiteympäristöön. Selvitän

tässä analyysiosuudessa, millainen on tyypillinen ”enterprise”-tason RPA arkkitehtuuri ja miten tuleva robottien hallinta pilvipalvelusta käsin sopii tähän.

Kerrosarkkitehtuuri on sovellusarkkitehtuurityyleistä yleisin ja se jaetaan päällekkäisiin kerroksiin, joilla on jokaisella oma tehtävänsä sovelluksen hallinnoimisessa. Alempi kerros välittää tietoa ylempiin kerroksiin. Ylimpänä on käyttöliittymäkerros, seuraavana sovelluslogiikasta vastaava kerros ja alimpana tiedon tallentamiseen tarvittavat kerrokset. Jokainen kerros on eristetty ja sovelluksen komponentit toimivat vain siinä kerroksessa eikä yksi komponentti toimi kerrosten välillä. Tämä logiikka mahdollistaa sovelluksen eri tasojen komponentteihin ketterät muutokset. (Richards, 2015. Luku 1.)

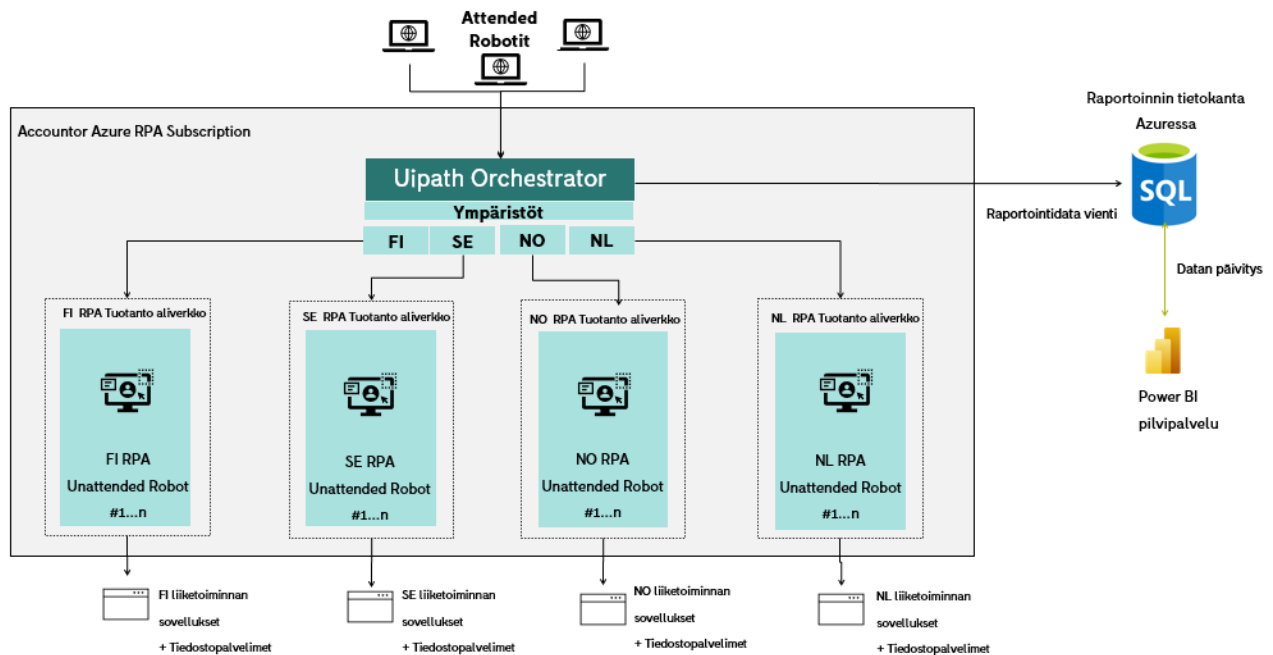
UiPathin arkkitehtuuri koostuu kolmesta tuotteesta, Studiosta, Robotista ja Orchestratorista. Arkkitehtuuria voidaan kuvata kolmen eri kerroksen tai tason kautta, jossa nämä tuotteet toimivat. Ensimmäinen näkyvä kerros on sovellustaso, jossa Studion avulla kehitetään projektit ja Robotit suorittavat projektin sisältämän automaatiokoodin. Toinen kerros on palvelintaso, jolla Orchestrator toimii huolehtien, että palvelimet suorittavat ajastetut tehtävät. Kolmas pysyvä kerros, joka huolehtii tiedon tallentamisesta, on tietokantataso, johon tallennetaan muun muassa tiedot Orchestratorin jonotehtävistä ja automaatioajojen tuottamat lokit. (Pedamkar, 2023.)



Kuva 7 - UiPath arkkitehtuurikerrokset (mukaiillen Kappagantula, 2023.)

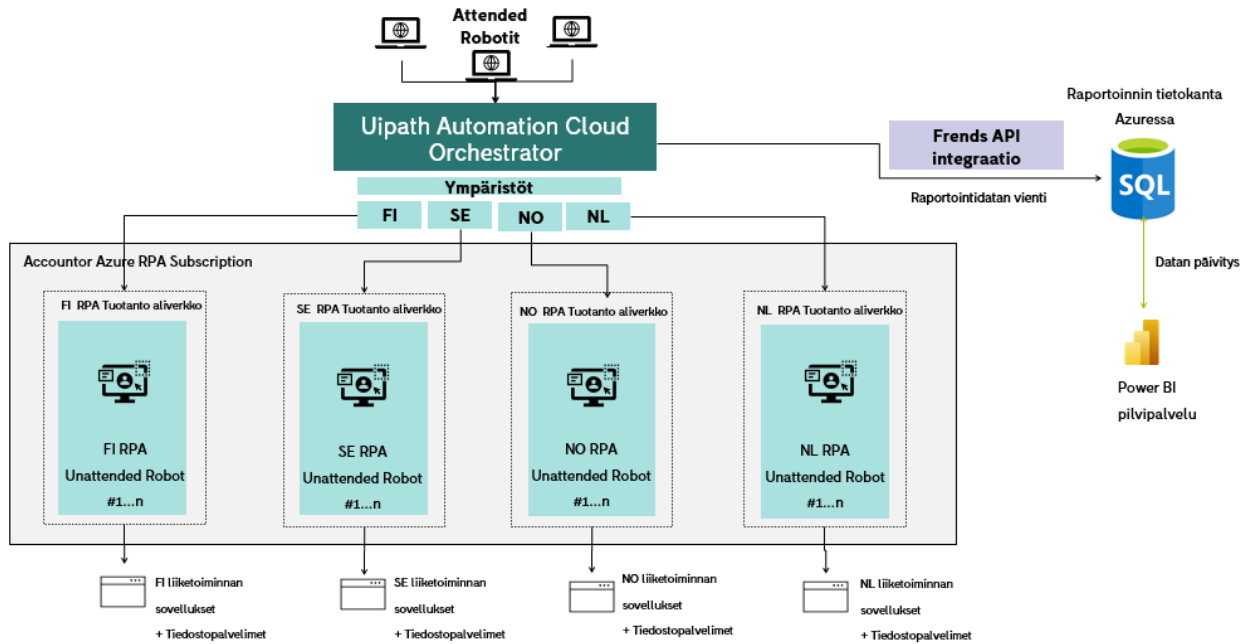
Jotta Studiossa luotu automaatioprojekti voidaan suorittaa robotilla, se julkaistaan NuGet pakettina Orchestratoriin. Jokaisella robotilla eli tietokoneella, joka on luotu Orchestratoriin, on oma ”machine key”-avain. Tämä on identiteetti, joka liittää tietokoneen ja sille asennetun UiPath Robot ohjelman käynnistettäväksi Orchestratorista. Koska projekti on tallessa Orchestratorissa, se voidaan suorittaa millä tahansa valitulla tietokoneella, joka on liitetty Orchestratoriin. (Kappagantula, 2023.)

Accountorin nykyinen RPA-arkkitehtuuri on jaettu maittäin, siten että jokaisella maalla on oma eriytetty ympäristönsä eli oma UiPath Orchestrator asiakasympäristö sekä oma rajattu virtuaalikoneiden aliverkko Azuren yhteisen ympäristön alla. UiPath Orchestrator tilaus on yhteinen, joten Orchestrator tietokanta on jaettu, josta viedään edelleen tiedot maiden yhteiseen raportointin tietokantaan, josta automaatioajojen tiedot viedään PowerBi:n visualisoitavaksi. Jokaisella maalla on omat liiketoiminnan sovelluksensa, joissa Unattended-robotit suorittavat automaatiotehtäviä sekä tallentavat ja hakevat tietoa maakohtaisilta tiedostopalvelimilta. Tämän lisäksi maissa on käytössä työasemakohtaisia Attended-robotteja jotka on kytketty Orchestrator-palveluun.



Kuva 8 - Accountorin nykyinen RPA-ympäristö

SaaS-siirto projektissa UiPath Orchestrator siirretään pois maiden yhteisestä Azuren pilvipalvelusta UiPathin Automation Cloud -palveluun. Lisäksi rakennetaan tiedonsiirtoratkaisu Friends integraatiotyökalulla jotta automaatioajojen tiedot saadaan vietyä maiden yhteiseen raportointin tietokantaan UiPathin pilvipalvelusta.



Kuva 9 - Accountorin tuleva RPA-ympäristö

3.6 Seurantaviikko 6

Keskiviikko 25.9.2024

Päivätavoite: Liittää Ruotsin kehityskone SaaS Orchestratoriin ja tarkistaa salasanasavarastot Azuresa.

Päivätoteuma: Kaikki Azuresa olevat testi- ja kehitysrobottikoneet on nyt siirretty käyttämään SaaS Orchestratoria. Salasanat eivät siirry tietojen migraatiossa, joten tarkistin ovatko kaikki tiedot tallessa Azure portaalissa olevassa salasanasäilössä. Joitain edelleen puuttuu ja nämä saattavat myös olla sellaisia, jotka eivät ole käytössä, mutta kaikki käytössä olevat arvot täytyy löytää jostain, koska salasanat ovat kryptattuna vanhassa Orchestratorissa eikä niitä voi sieltä kopioida edes manuaalisesti uuteen.

Torstai 26.9.2024

Päivätavoite: Viimeistellä Ruotsin SaaS-siirtosuunnitelma ja aloittaa työsuunnitelman tekeminen Suomen tuotannon siirrosta.

Päivätoteuma: Tarkistin Operations-tiimiltä, onko heillä tietoa muutamasta tunnuksesta, joille ei löytynyt tietoja salasanasavarastoista ja he täydensivät tietoja, jotka vein varastoon. Kävin läpi Ruotsin suunnitelman ylläpidon päivittäisessä palaverissa osallistujien kesken ja säädimme vielä yhdessä prioriteetteja, joiden mukaan prosessit ja koneet siirretään SaaS Orchestratoriin. Tarkensimme

Tech Leadin kanssa suunnitelmassa olevat tehtävät, jotta samaa mallia voidaan käyttää myös muiden maiden siirrossa. Lisäsimme myös kommunikaatiosuunnitelman eri sidosryhmien suuntaan.

Raportointiratkaisua varten tarvitsemme jonotapahtumia nykyiseen on-premise Orchestratorin testiympäristöön. Olin ajastanut yhden testiprosessin eilen ja ihmettelin miksi jonotapahtumia ei ollut muodostunut. Syynä oli tietysti se, että olin ajastanut prosessin on-premise Orchestratorista ajamaan testirobotilla, mutta testirobotti on jo liitetty SaaS-Orchestratoriin, joten se ei ottanut ajastettua käynnistyskäskyä vastaan.

Aloitin Suomen tuotannon SaaS-siirron suunnittelun listaamalla aktiiviset prosessit ja niiden kriittisyysluokituksen liiketoiminnan kannalta sekä robotit, joilla niitä ajetaan.

Perjantai 27.9.2024

Päivätavoite: Suomen tuotannon siirron työsuunnitelman tarkentaminen ja Orchestrator jonojen ja tallennettujen tunnusten kartoittaminen.

Päivätoteuma: Tein listauksen kaikista Orchestratorissa olevista jonoista, joissa on asetettu käytettäväksi yksilöllinen jonotapauksen numerointi. Osa prosesseista ajaa kerran kuukaudessa tai vuodessa ja tämän tarkoitus on verrata, löytyykö Orchestratorista jo kyseessä oleva jonotapaus ja hylätä sen sisäänluku. SaaS Orchestratoriin ei siirry on-premise Orchestratorin jonotapauksista mitään tietoa, joten siirryttäessä pilveen ei ole historiatietoja, joihin verrata uusia jonotapauksia ja vaara kaksoiskappaleille on olemassa. Pohdimme Tech Leadin kanssa, että tämä voitaisiin ratkaista viemällä kaikki olemassa olevat jonotapaukset vanhasta Orchestratorista tiedostoon ja rakentaa uusi prosessi, jolla lukea tiedot uuteen Orchestratoriin ja merkata samalla nämä käsitellyiksi. Jatkoin salasanalistauksen Norjan ja Suomen tunnuksista ja kartoitin mitä puuttuu ja mistä saan ne kerättyä Azuren Keyvaultiin.

Tiistai 1.10.2024

Päivätavoite: Orchestratoriin tallennettujen tunnusten kerääminen ja tallentaminen.

Päivätoteuma: Ennalta määriteltyjen toimintatapojen noudattaminen helpottaa usein muiden työtä, mutta tässä tapauksessa se olisi helpottanut myös omaa työtäni. Huomasin että Suomen osalta löytyi useampi sellainen käyttäjä-salasana pari joita en itse aikanaan ollut tallentanut Azuren salasanasäilöön ja nyt jouduin metsästäämään niitä. Joukossa oli toki muidenkin kehittäjien unohduksia, mutta varmimmin tämä opetti itseäni kantapäähän kautta.

Perjantai 3.10.2024

Päivätavoite: SaaS-Orchestrator käyttöönoton kommunikointisuunnitelman ja materiaalien valmistelu Suomen organisaatiolle.

Päivätoteuma: Tein PowerPoint-esityksen SaaS-siirrosta Suomen tukitiimejä, business analysteja ja prosessiomistajia varten. Alkuperäisenä suunnitelmana oli tehdä siirto SaaSiin ensi viikolla, mutta tällä hetkellä näyttää siltä, että aikataulu siirtyy kahdella viikolla koska raportointiratkaisun testaaminen on vielä kesken. Laadin sähköpostiviestin pohjan, jossa kerrotaan huoltokatkon aikataulutusta alkaen edellisestä illasta siirtopäivän aikatauluun. SaaS Orchestrator ei eroa merkittävästi vanhasta versiosta, ainoastaan kirjautumisessa tulee uutena kertakirjautuminen omalla sähköpostiosoitteella aiempien tunnus-salasana parin sijaan.

Viikkoanalyysi 6 / viikko 39 ja 40

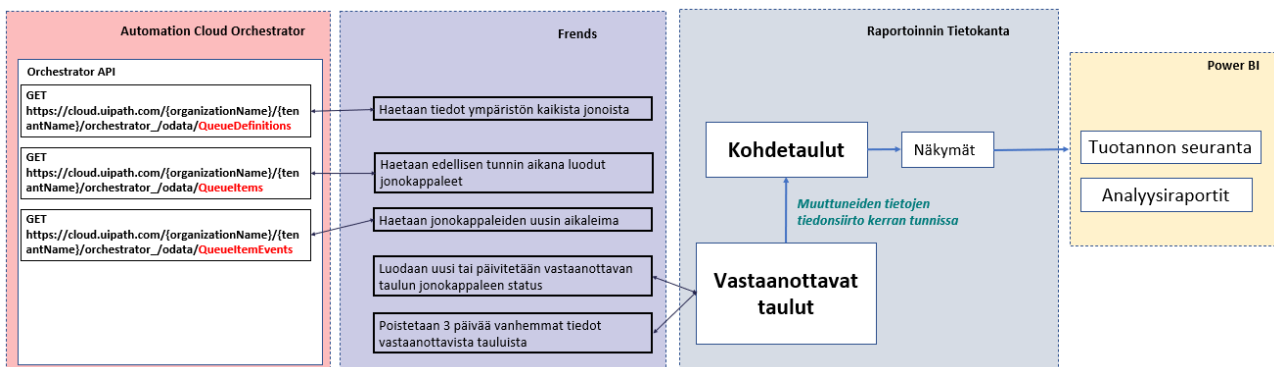
Seurantaviikolla kaksi selvitin, miten nykyinen ohjelmistorobotiikan automaatioiden raportointi on toteutettu ja miten suunnittelemme sen toimivan jatkossa, kun automaatioiden tiedot ovat saatavilla vain toimittajan pilvipalvelussa. Raportointiratkaisu on nyt tarkentunut ja sitä on testattu sekä nykyisessä testiympäristössä että tuotannossa. Selvitän tässä viikkoanalyysissä tarkemmin, miten ratkaisu toimii.

Ohjelmistorobotiikkaa voidaan käyttää API-rajapintoihin tehtäviin integraatioihin, mutta koska robotiikka on lähtökohtaisesti tarkoitettu toimimaan käyttöliittymällä, sitä tulisi käyttää vain, jos tehtävää ei pysty tekemään käyttöliittymän kautta kuten ihminen sen suorittaisi. Jos automaation voi rakentaa APIa käyttäen on robotiikan käyttäminen siihen tarkoitukseen turhaa. Joissakin tapauksissa organisaatio voi myös rajoittaa APIen hyödyntämistä koska ne ovat usein hintavia ja jokainen haku saattaa maksaa erikseen. (Sahgal, 2024, Luku 5.) Omassa tiimissäni on osaamista integraatioista ja rajapintojen hyödyntämisestä, joten valitsimme siksi toteuttaa raportoinnin tietojen haun Frendsillä emmekä käytä siihen ohjelmistorobotiikkaa, vaikka myös UiPath tarjoaa mahdollisuuden käyttää API-rajapintoja.

Frends integraatioalusta on matalakoodinen työkalu, jolla voi rakentaa yhteyksiä eri sovellusten välille, luoda ja ylläpitää API-rajapintoja ja tehdä datalle konversioita ja validointia. Frendsiä voi käyttää joko toimittajan julkisesta pilvipalvelusta tai asennettuna omaan laiteympäristöön. Sovellus on .NET pohjainen ja valmiit matalakooditoiminnot käsittelevät muuttujia sekä C# koodia Handlebars notaation avulla. Frendsissä on sisäänrakennettuna versionhallinta ja se mahdollistaa koodikirjastojen uudelleen käytön useissa eri projekteissa. (Frends, 2024.) Handlebars on koodikieli, jossa on tekstin lisäksi mallipohja, johon laitetaan objekti, joka halutaan ajaa tekstiä generoidessa (Handlebars, 2024). Frends kutsuu Handlebars notaatiossa olevaa objektia `.ToString()` -metodilla eli kaikki objektit, kuten taulukot, muunnetaan tekstijonoksi (Docs.Frends Handlebars, 2024).

UiPath Orchestrator tallentaa jonoihin tiedot yksittäisestä automaation ajamasta tapauksesta. Yhdellä automaatioprosessilla voi olla useita jonoja, ja jonoissa tieto voi olla vapaassa formaatissa tai aina samassa määrämuodossa. Robotti hakee jonosta käsiteltävän automaatiotapauksen ja käsittelyn aikana jonokappaleen status muuttuu uudesta prosessoitavan kautta joko epäonnistuneeksi tai onnistuneeksi. Robotti voi lähettää jonokappaleen prosessin seuraavaan jonoon, josta robotti taas hakee sen käsiteltäväksi ja merkkää statuksen. (Docs.UiPath, 2024, About Queues and Transactions.) Automaatiotapaukset ovat aina jonokappaleita, usein asiakaskohtaisia tai dokumentikohtaisia tapauksia. Automaatioiden raportoinnissa seurataan näitä yksittäisiä jonokappaleita, niiden onnistumisprosenttia ja kappaleiden määrää ja sitä kautta robotin säästämää ihmistyön aikaa.

Frends integraatioalusta hakee UiPathin Automation Cloud Orchestratorin rajapinnasta tiedot käsitellyistä jonokappaleista ja vie ne raportoinnin tietokantaan (Kuva 10). Friendsiin on toteutettu kaksi eri prosessia, joista toinen käynnistyy kerran tunnissa ja hakee tiedot ympäristössä olevista jonoista. Tähän tarkoitukseen käytetään API-rajapintaa QueueDefinitions. Toinen prosessi ajaa jatkuvasti ja käyttää tietoa olemassa olevista jonoista ja hakee niistä jonokappaleet eli QueueItem-tiedot ja jonokappaleiden uusimman aikaleiman perusteella, joka saadaan noudettua kullekin jonokappaleelle QueueItemEvents hakua käyttäen, päivittää tai luo uuden jonokappaleen tiedon raportoinnin tietokannan tauluun. PowerBi näkymä ei ole reaaliaikainen tieto siitä mikä on jonokappaleen status, vaan näkymä päivittyy kerran tunnissa ja sen tarkoitus on antaa tietoa päivän aikana ajetuista automaatioista sekä kerätä historiatietoa automaatioajojen onnistumisesta.



Kuva 10 - Jonokappaleiden haku ja päivitys Friendsiä käyttäen

Jos käy niin että tämä raportoinnin prosessi ei toimisi Friendsissä, eli integraatioalusta olisi alhaalla tai teknisen vian takia raportointi ei toimisi, on mahdollisuus ajaa prosessi manuaalisesti vikatilanteen mentyä ohi. Manuaalisesti ajettaessa voidaan prosessille asettaa ajanjakso, jolta jonokappaleet halutaan kerätä ja päivittää raportoinnin tietokantaan.

3.7 Seurantaviikko 7

Tiistai 29.10.2024

Päivätavoite: Ruotsin tuotannon Orchestratorin siirron ensimmäiset työvaiheet vanhassa ja uudessa ympäristössä.

Päivätoteuma: Robottiikan alustatiimi ajoi alkuiltapäivästä migraatiotyökalulla tiedot on-premise Orchestratorista uuteen Automation Cloud Orchestratoriin. Jatkoain asettamalla käsin kaikki tarvittavat salasanat paikalleen ja kopioin myös prosessien ajastukset eli triggerit uuteen ympäristöön. Suunnitelman mukaan pysäytin on-premisen ajastetut ajot, jätin vain muutaman kriittisen prosessin ajamaan illaksi, tavoitteena että aamulla jonoissa ei ole yhtään uusia jonokappaleita koska ne eivät siirry pilveen. Huomenna kytken Ruotsin tuotannon robotit Automation Cloudiin.

Keskiviikko 30.10.2024

Päivätavoite: Ruotsin tuotannon Orchestratorin käyttöönotto.

Päivätoteuma: Raportointiratkaisu siirrettiin onnistuneesti toimimaan Automation Cloudin tuotannon kanssa, jotta Ruotsin automaatioiden jonotehtävät saadaan siirrettyä alusta lähtien raportoinnin tietokantaan ja sieltä visualisoitua PowerBissä. Aloitin aamun tarkistamalla, että jonot ovat tyhjä ja vaihdoin UiPath Assistantin kautta vanhan Orchestrator yhteyden uuteen Orchestratoriin. Ensimmäiset koneet onnistuivat hyvin, mutta en päässyt lainkaan etäyhteydellä viimeiselle koneelle. Etäyhteys aukesi mutta edessä oli musta ruutu eikä kone ottanut mitään vastaan. Pyysin Ruotsin IA-omistajaa tekemään kiireellisen pyynnön lähitukeen, että he käynnistäisivät tämän robottikoneen uudelleen. Operations tiimistä oli yksi henkilö auttamassa minua automaatioprosessien testaamisessa Automation Cloudissa. Saimme kaikki prosessit ajettua päivän aikana ja totesimme että Ruotsin siirto saatiin tehtyä onnistuneesti.

Torstai 30.10.2024

Päivätavoite: Suomen Orchestratorin siirron kommunikointi liiketoiminnalle ja suunnitelman läpikäynti.

Päivätoteuma: Sovimme Suomen Orchestratorin siirron aikataulusta ja kävimme tehtävien jaon läpi yhdessä Tech Leadin ja kehittäjäkollegani kanssa. Suomella on eniten sellaisia automaatioita, jotka tarvitsevat ajoaikaa kuun vaihteen tehtäviin, joten robotteja on määrällisesti eniten verrattuna muihin maihin. Tämä aiheuttaa eniten testaustyötä projektissa, joten jaamme työtä, että se saa-

daan tehtyä mahdollisimman pian ja automaatioiden huoltokatko on lyhyt. Lähetin ensi viikon aika-
taulun tukitiimeille ja IA-omistajille tiedoksi sähköpostitse ja tein DevOpsiin erillisen palvelupyynn-
nön, jonka kautta voimme kommunikoida siirron etenemisestä näille tärkeimmille sidosryhmille.

Tiistai 5.11.2024

Päivätavoite: Suomen Orchestratorin siirron ensimmäinen vaihe, asetusten lisääminen datamigraa-
tion jälkeen.

Päivätoteuma: Ilmeni, että Ruotsin siirron jälkeen oli tullut ongelmia jonotapausten käsittelyssä. Jo-
notapaus merkitään nykyisessä koodissa aina ensin onnistuneeksi ja sen jälkeen vasta tarkiste-
taan, oliko se epäonnistunut ja muutetaan status Orchestratorissa. Tämä on toteutettu sen takia
että jonotapauksilla säilyy yksityiskohtaiset tiedot ja analytiikka, jonka perusteella lasketaan auto-
maation säästämää aikaa. Uusi Orchestrator versio pilvessä ei kuitenkaan ole enää yhteensopiva
tämän vanhan koodin kanssa eikä jonotapausten statusta voi muuttaa, kun se on kerran asetettu
onnistuneeksi. Aloitin siis päivän sillä että päivitin kaikki Suomen prosessit käyttämään uutta
UiPath system.activities pakettia jossa epäonnistuneellekin tapaukselle voi tallentaa analytiikka- ja
jonotapausten tarkemmat tiedot. Jatkoin sitten asetuksen konfiguroinnilla. Tiedotin tukitiimejä il-
lalla, että automaatioiden katko alkaa ja yön aikana ei ajeta uusia tapauksia jonoon.

Keskiviikko 6.11.2024

Päivätavoite: Huoltokatkon aikana Suomen robottien kytkeminen Automation Cloudiin ja automaa-
tioprosessien käynnistämisen testaaminen siellä.

Päivätoteuma: Liitin Suomen robotit ongelmitta Automation Cloudiin ja aloimme testata prosesseja.
Päivitetyn system.activities paketin kanssa tuli virhe sellaisen prosessin kanssa joka kutsuu raja-
pintaa ja ajaa kirjastokomponenttia. Kirjastokomponentin system.activities versio oli vanhempi kuin
projektin versio, joten päivitin kirjastokomponentin ja prosessin. Tässä kesti jonkin aikaa koska li-
säksi pakollisten muuttujien käsittely ei toiminut kuten ennen ja minun piti muokata kirjastoa odotet-
tua enemmän. Sain kuitenkin lopulta prosessin toimimaan ja päivän päätteeksi näytti siltä, että
suurin osa prosesseista oli toiminnassa ja viestin käyttäjille, että he voivat kirjautua Automation
Cloudiin.

Viikkoanalyysi 7 / viikko 44 ja 45

Viikolla 7 saimme jo suurimman osan projektin tavoitteista saavutettua. Aiemmin viikon 4 analy-
sissa käsittelin version nostoja ja versionhallintaa ja viikon 5 analyysissä robotiikka-arkkitehtuureja.

Olen oppinut tämän projektin aikana molemmista lisää ja viikolla 7 nämä opit konkretisoituivat pilvi-version ja koodiversion yhteensopivuusongelmaan. Jotta kokonaisuus toimii yhteen ja automaatiot pysyvät pystyssä, täytyy arkkitehtuurissa huomioida muutosten aiheuttamat riippuvuudet. Vaikka olimme siirtäneet ensin testiympäristön pilveen ja siellä kaikki vaikutti toimivan niin tuotannossa tulikin yllätys. Kaikkea ei aina voi ennakoida, joten on tärkeää tehdä suunnitelma, jossa on varauduttu siihen, että tapahtuu jotain yllättävää ja tarvitaan lisäkäsia auttamaan tai jotain työvaihetta voidaan siirtää. Jotta suunnitelmasta voisi poiketa, täytyy olla suunnitelma.

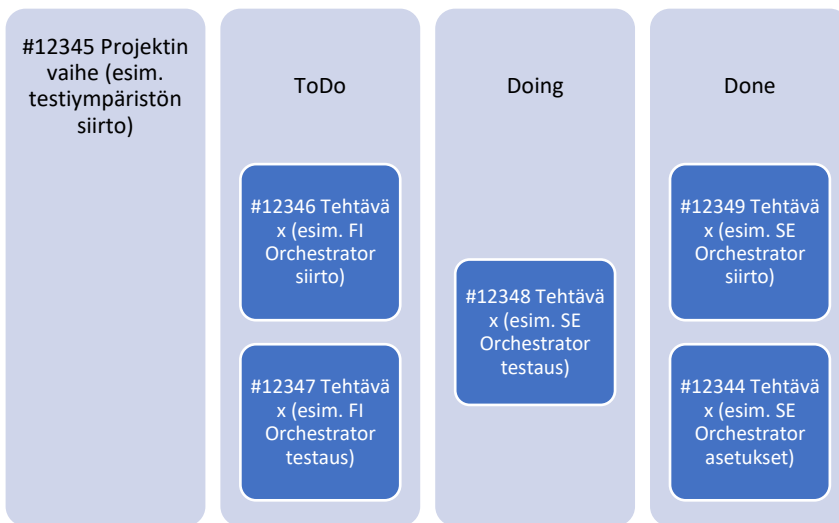
Teimme Orchestrator SaaS-siirron suunnitelmaa ennen projektin aloitusta yhdessä Tech Leadin kanssa, ja tarkensimme projektin edetessä sitä yksityiskohtaiselle tasolle ja katselmoimme yhdessä liiketoiminnan edustajien kanssa, jotka sitten hyväksyivät suunnitelman. Ylätason suunnittelussa käytettiin exceliä ja osa tehtävistä vietiin seurattavaksi Azure DevOpsin Kanban boardille ja testauksen ja asetusten osalta tuloksia merkattiin exceliin.

Mäntyneva jakaa projektin neljään eri vaiheeseen: valmistelu, suunnittelu, toteuttaminen ja päättäminen. Valmisteluvaihe kannattaa hoitaa huolella, sillä se auttaa varsinaista projektin suunnittelua, Suunnittelussa asetetaan tarkat tavoitteet ja kuvataan toimenpiteet, jotka pilkotaan tehtäviksi projektisuunnitelmaan. Toteutusvaiheessa toteutetaan suunnitelmassa kuvatut tehtävät ja projekti päätetään, jotta voidaan varmistua, että tavoitteet on saavutettu ja kaikki tehtävät on tehty. (Mäntyneva, 2016, s. 15.) Nyt kun SaaS-siirtoprojekti on toteutusvaiheessa, on jo mahdollista analysoida valmistelu- ja suunnitteluvaihetta. Valmisteluvaiheeseen käytettiin paljon kalenteriaikaa, koska tietoturvatietojen piti auditoida pilvipalvelu. Valmisteluun olisi ollut hyvä ottaa mukaan myös projektin osallistujia, jotta olisimme päässeet tehokkaammin eteenpäin suunnitteluvaiheessa. Suunnitteluvaihe tehtiin hyvin ylimalkaisella tasolla ja suunnitelmia tehtiin viikko viikolta tarkemmiksi, sen sijaan että olisi tehty yksityiskohtainen suunnitelma ennen projektin toteutuksen aloittamista. Projektiryhmän kokoamisen jälkeen on hyvä järjestää projektin aloituskokous ja sopia tavoitteista ja toimintatavoista (Mäntyneva, 2016, s. 26.). Projektiryhmää ei rajattu tai projektin rooleja ei sovittu missään kohtaa, vaan käytännössä on ollut vain joukko ihmisiä, joista jokainen on tehnyt tarvittaessa jotain ja siksi on ollut myös hyvin epäselvää mikä tehtävä on kenenkin vastuulla. Ammatillisen kehittymiseni kannalta tärkeä itsereflektio on tämän suhteen ollut, että vaikken ole teknisesti taitavin, minun pitäisi nostaa esiin projektinhallinta- ja johtamisosaamistani ja antaa sitä enemmän käyttöön organisaatiossa.

Suunnitteluvaiheeseen panostamalla olisi saatu projektissa läpinäkyvyyttä siihen, mitä kukakin tekee ja vietyä projektin tehtävät Kanban boardille DevOpsiin osaksi päivittäistä muuta tekemistä. Projektipäällikön nimeäminen olisi auttanut selkiyttämään kuka on vastuussa tehtävien etenemisen

seurannasta ja viestinnästä, nyt kun rooleja ei nimetty, tämä piti aina sopia tapauskohtaisesti eikä ollut lainkaan tehokasta.

Kanban konsepti on alunperin Toyotan kehittämä oikea-aikaisen valmistuksen metodi. Kanbanin kolme johtavaa periaatetta ovat tehtävien visualisointi, työn alla olevien tehtävien rajoittaminen ja tehtävien kulun seuranta. (Hammarberg & Sunden, 2014, luku 2.). Käytämme tiimissä kaiken työn visualisointiin DevOps boardeja ja projektin aikana listasimme myös projektin ylätason tehtäviä sinne.



Kuva 11 - Kanban board esimerkki

Työnkulku on määritelty siten, että kehitystehtävä tai tässä tapauksessa projekti jaetaan ylätason vaiheisiin, jotka taas pilkotaan alitehtäviin. Nämä tehtävät visualisoidaan boardilla ja niitä kuljete-taan vaiheesta toiseen. Tehtävät, jotka on suunniteltu tehtäväksi, näkyvät "ToDo"-osiossa ja kun tehtävä otetaan työn alle, se siirtyy "Doing"-osioon ja sieltä "Done" tilaan. Näin on helppo seurata kunkin tehtäviä samassa isossa näkymässä ja käydä ne päivittäin läpi mikä on edennyt ja onko jo-tain esteitä tekemiselle.

3.8 Seurantaviikko 8

Torstai 07.11.2024

Päivätavoite: Suomen Orchestrator siirron viimeistelytöiden jatkaminen.

Päivätoteuma: Jatkoin prosessien päivittämistä jonotapausten käsittelymuutoksella. Osa proses-seista ajaa vain kuun vaihteessa, joten niitä ei voida testata ennen seuraavaa kuun vaihdetta, siksi en kiirehtinyt näitä aiemmin. Yksi hybridiprosessi, jossa Friends tekee osan automaation töistä ja tuottaa sitten Orchestrator rajapinnan kautta jonotapausten robotille käsiteltäväksi, oli jäänytkin

osoittamaan vanhaan Orchestratoriin. Jonotapaukset menivät väärään paikkaan ja pyysin tukitii- mistä, että he tekevät automaation triggerin uudestaan, jotta saadaan testattua, että hybridipro- sessi toimii oikein. Toinen prosessi siirtää jonotapausten käsittelyä myöhemmäksi ja näitä tapauk- sia oli jäänyt vanhaan Orchestratoriin muutama. Sain ne siirrettyä uuteen muokkaamalla triggeriä, joka valitsee asiakkaat, joista jonotapaukset muodostetaan ja ajoin nämä uuteen Orchestratoriin.

Päätimme tehdä Norjan Orchestrator siirron seuraavalla viikolla ja tein sitä varten suunnitelman, jonka kävimme yhdessä kollegoiden kanssa läpi ja jaoimme tehtävät.

Perjantai 08.11.2024

Päivätavoite: Norjan Orchestrator siirron valmistelevat tehtävät.

Päivätoteuma: Oikeastaan minun piti perjantaina tehdä muita kehitystöitä, mutta koska kollega oli kiinni Suomen tuotannon ongelmien selvittelyssä niin otinkin häneltä tehtävän ajaa nykyisen Or- chestratorin Norjan prosessien jonotapaukset uuteen Automation Cloudiin. Joissain prosesseissa on toteutettu duplikaattitarkistus siten, että robotti ei lisää jonotapausta Orchestratoriin jos samalla viitteellä löytyy jo olemassa oleva jonotapaus. Näiden prosessien kohdalla on siis tuotava kaikki vanhat tapaukset uuteen Orchestratorin tietokantaan, jotta tarkistus toimii. Rakensin prosessin, joka lukee tapaukset excelistä ja ajaa ensin tapaukset jonoon ja sitten merkkää ne suoraan onnis- tuneiksi.

Maanantai 10.11.2024

Päivätavoite: Norjan Orchestrator siirron valmistelevien tehtävien loppuun saattaminen.

Päivätoteuma: Laitoin jonotapausten siirtoprosessin ajamaan heti aamulla. Ongelmana oli se, että tapauksia oli noin 13000 kappaletta ja niiden ajaminen kesti monta tuntia koska minulla oli käy- tössä vain yksi robottikone, kun muut robotit tekivät vielä vanhassa Orchestratorissa töitä. Lopulta illalla yhdeksän aikaan suurin osa oli siirretty ja päätin jatkaa seuraavana päivänä. Asetin puuttuvat salasanat ja triggerit pilvi-Orchestratoriin.

Tiistai 11.11.2024

Päivätavoite: Norjan Orchestratorin siirto Automation Cloudiin.

Päivätoteuma: Kytkein aamulla Norjan robotit pilvi-Orchestratoriin ja jatkoin jonotapauksen siirtä- mistä vanhasta uuteen Orchestratoriin. Käynnistin prosesseja ja kävi ilmi, että tuotannon palkan- laskennan järjestelmän tietokantaan kirjautumisen salasana oli väärin. Käänsin yhden roboteista

vielä osoittamaan vanhaan Orchestratoriin ja kollegani ajoi prosessin, joka hakee salasanan Orchestratorin kannasta. Päivän päätteeksi kaikki prosessit oli testattu ja ajoivat onnistuneesti. Nyt kaikkien maiden automaatiot on siirretty ohjattavaksi Automation Cloudin Orchestratorista.

Perjantai 22.11.2024

Päivätavoite: On-premise Orchestratorin sulkeminen ja palvelujen irtisanominen.

Päivätoteuma: Tech Leadimme sopi robotiikan alustatoimittajan kanssa vanhan Orchestrator palvelimen sulkemisesta ja he listasivat mitä Azure palvelimia ja palveluita ei enää tarvita. Pyyntö sulkea palvelimet laitetaan eteenpäin it-infran toimittajalle, joka sitten poistaa palvelimet Azure tilaukseltamme. Minun osaltani suljin projektin tehtävät DevOpsissa ja jatkan Orchestratoriin ja Insightsiin tutustumista tulevina viikkoina.

Viikkoanalyysi 8 / viikko 46 ja 47

Projekti on ollut pitkä ja sen tekeminen normaalin työn ohessa on ollut myös ajoittain väsyttävää mutta enemmän silti palkitsevaa uuden oppimisen kannalta. Alkuperäinen tavoitteemme oli saada kaikki neljä maata siirrettyä pilveen lokakuun loppuun mennessä. Tästä jouduttiin joustamaan kahdella viikolla sekä raportointiratkaisun testaamisen että muiden projektien aikataulujen vuoksi. Jälkikäteen analysoituna raportointiratkaisun suunnittelu ja testaus olisi voitu aloittaa jo kesällä ja rakentaa se valmiiksi ennen siirtoprojektia.

Projektin viimeisellä viikolla huomasin, että hybridiprosessin asetukset olivat jääneet osoittamaan väärin Friendsin puolella. Hybridiautomaatio tarkoittaa tällä hetkellä tiimissäni sellaista automaatioprosessia, jossa sekä integraatioalusta että robotti tekevät osan työstä. Esimerkkinä Friends voi hakea tietoja sovelluksen API-rajapinnasta, kuten mille asiakkaille on jo laskettu palkat, ja syöttää näiden perusteella suoraan jonotapauksen Orchestratorin jonoon, joka sitten antaa robotille käskyn jatkaa prosessia ja kirjautua sovelluksen käyttöliittymälle ja tulostaa palkkojen raportoinnin tulokset, joita ei ole saatavilla rajapinnasta hakemalla.

UiPath määrittelee hybridiautomaation prosessiksi, jossa käytetään sekä taustalla toimivia robotteja, että työasemassa ajavia robotteja eli yhdistelmäksi unattended ja attended automaatioita (UiPath, 2024, eGuide to the six automation scenarios.). Organisaationi tapauksessa oikeampi termi kuvaamaan sellaista automaatiota, jossa käytetään eri työkaluja, voisi olla hyperautomaatio. Gartnerin mukaan hyperautomaatio sisältää useita teknologioita, työkaluja tai alustoja, kuten ohjelmistorobotiikka, integraatioalusta, matalakoodityökaluja, sekä tekoälyä ja koneoppimista (Gartner, 2024, Hyperautomation.). Uskon, että rakennamme jatkossa lisää uusia hyperautomaatioita, jolloin

automaatioprosesseista tulee edelleen monimutkaisempia. Nämä riippuvuudet täytyy ottaa huomioon tehtäessä muutoksia järjestelmiin ja suunnittelun sekä testauksen määrä kasvaa. Hyperautomaatioissa tavoitellaan hyötyjä siitä, että tarkoitukseen parhaiten soveltuva työkalu tekee oman osansa prosessista eikä automaation kehittämiseen ja ylläpitoon tarvita niin paljon aikaa, kun voimme käyttää parhaat toiminnot matalakoodisovelluksista.

Hyperautomaation ongelmat liittyvät prosessien monimutkaisuuteen, monen eri työkalun yhdistelmä samassa prosessissa voi johtaa siihen, että prosessia on hyvin hankala ylläpitää. Jos teknologioita on useita, voi prosessin kehittämisestä tulla myös hyvin kallista, jolloin on syytä tehdä kustannusten ja prosessin tuomien hyötyjen analyysi huolella. (Sahgal, 2023. Luku 10.) Tähän mennessä tiimissämme on löytynyt osaamista sekä integraatioista- että robotiikasta, vaikka niiden ylläpito onkin toistaiseksi järjestetty erikseen. Tulevaisuudessa suunnittelemme, että saisimme jaettua osaamista siten, että voisimme yhdistää ylläpidon ja tukea näin myös hyperautomaatioita paremmin ja laajentaa samalla tiimin yksittäisten jäsenten ymmärrystä integraatioalustasta ja robotiikasta. Toivottavasti saamme tulevien vuosien aikana enemmän pilvipalveluja käyttöömmä ja sitä kautta koneoppimisen ja tekoälyn sovellutuksia osaksi hyperautomaatioita. Tässä projektissa tehty siirto UiPath Automation Cloudiin on askel kohti tuota tavoitetta.

4 Pohdinta

Opinnäytetyön seurantajakson alkaessa kuvasin nykytilaa ja sen hetkistä osaamistasoani roolisani automaatiokehittäjänä. Ammatillinen kehittymiseni oli jo hyvällä tasolla, mutta opinnäytetyön ja SaaS-siirtoprojektin kautta asetin tavoitteeksi, että oppisin lisää robottien hallinnasta ja automaatioiden raportoinnista. Syksyn aikana olen kehittänyt ammatillisesti laajentaen osaamistani robottien päivittämiseen, Norjan ja Ruotsin ympäristöjen hallinnointiin sekä vienyt yhdessä projektiryhmän kanssa Orchestrator sovelluksen onnistuneesti pilvipalveluun. Opinnäytetyö auttoi minua purkamaan aiempaa ja uutta automaatioiden raportoinnin teknistä ratkaisua osiin ja ymmärtämään vaihtoehtoja ja valitun ratkaisun toimintaa. Ilman opinnäytetyötä en ehkä olisi selvittänyt raportointiaihetta, vaan koska sen toteutus ei suoraan kuulu minun rooliini, olisin jättänyt sen kollegoilleni.

Orchestrator SaaS-siirto projektin suunnittelu, tietoturvakartoitukset ja liiketoimintahyödyn punnitseminen aloitettiin jo viime keväänä. Valitsin jo silloin, että kirjoittaisin projektista opinnäytetyöni ja onneksi saimme siirron tehtyä syksyn aikana, jotta pystyin syventämään opinnäytetyön avulla osaamistani etsimällä aiheesta tietoperustaa analyysien avuksi. Syvensin osaamistani robottien hallinnasta, robotiikka-arkkitehtuurista ja löysin kirjallisuudesta hyvin työtäni kuvaavan luonnehdinnan: Sahgal kuvaa RPA-ratkaisuarkkitehdin roolia ”jokapaikanhöyläksi” jonka tulee ohjelmoinnin ja teknisten sovellusten lisäksi tietää IT-infrastruktuurista, tietoverkoista, tietoturvasta ja tuntea alan standardit (Sahgal, 2023, Luku 1). Minusta tuntuu usein työssäni, että vaadittava osaamisalue on hyvin laaja. Tämän vuoksi yhteistyötaidot ja kyky hankkia tietoa sekä organisaation ulkopuolelta että omasta organisaatiosta, ovat hyvin tärkeitä työssä onnistumisen kannalta. Projektin aikana olen oppinut tuntemaan organisaatiossani toimivia henkilöitä paremmin ja hakemaan apua eteen tulleisiin ongelmiin eri tahoilta. Yhteistyökykyeni ovat kehittyneet.

Valmiuteni tehdä projektimuotoisesti töitä olivat hyvällä tasolla jo ennen tämän projektin alkamista, olinhan tehnyt yli 10 vuotta töitä projektitoimitusten kanssa. Huomasin projektin aikana, että nykyisen organisaationi projektikulttuurissa ja projektien läpiviennin osaamisessa on kehitettävää. Oivalsin, että jatkossa tämä on osa-alue, johon kannattaa kiinnittää huomiota ja hakea tarvittaessa tukea ja osaamista ulkopuolelta. Työtapojen osalta dokumentointia voisi myös jatkossa kehittää edelleen. Huomasin että automaatioiden raportoinnin osalta suunnitelma oli lähinnä henkilöiden päässä eikä teknistä ratkaisua ollut dokumentoitu ennen kuin sitä alettiin rakentaa.

Tärkein oppi, jonka sain SaaS siirtoprojektin ja opinnäytetyön kirjoittamisen aikana, on että IT-projektissa yllättävän suuri osa ajasta käytetään suunnitteluun ja valmisteluun ja huomattavan pieni aika itse tekemiseen. Koska ympäristö on niin monimutkainen ja täynnä riippuvuuksia, täytyy ensin varmistaa, että kaikki muuttujat on huomioitu ja tämän kartoittamiseen kuluu paljon aikaa. Toinen oppi, jonka koitan muistaa tulevaisuudessa, on kommunikaation ja tiedottamisen tärkeys.

Uskon, että tulevaisuudessa tulen edelleen kehittämään osaamistani robotiikan osalta. Koska hyperautomaatioita kehitetään yhä enemmän, tulee integraatio- ja tekoälyosaamisesta varmasti koko ajan tärkeämpää. Seuraavaksi tavoitteeksi asetan integraatioiden opettelun ja jään seuraamaan verkkaisesti etenevää tekoälymurrosta.

Lähteet

AWS. 2024. What's the difference between SaaS and On Premises? Luettavissa: <https://aws.amazon.com/compare/the-difference-between-saas-and-on-premises/>. Luettu 01.11.2024.

Chandrasekara, C. & Herath, P. Apress 2020. Hands-on Azure Repos: Understanding Centralized and Distributed Version Control in Azure DevOps Services.

Demiliani, S. & Malik, A. & Zaal, S. Packt Publishing, Limited 2020. Azure DevOps Explained: Get started with Azure DevOps and develop your DevOps practices.

Docs.Friends Handlebars. 2024. Luettavissa: <https://docs.friends.com/en/articles/8015990-introduction-to-handlebars>. Luettu 21.10.2024.

Docs.UiPath Compatibility Matrix. 2024. Compatibility Matrix. Luettavissa: <https://docs.uipath.com/overview/other/latest/overview/compatibility-matrix>. Luettu 14.09.2024.

Docs.UiPath Insights. 2024. Insights. Luettavissa: <https://docs.uipath.com/insights/automation-cloud/latest/user-guide/about-insights>. Luettu 09.09.2024.

Docs.UiPath. 2024. Luettavissa: <https://docs.uipath.com/robot/standalone/2023.10/user-guide>. Luettu 16.9.2024.

Docs UiPath Deprecation. 2024. Deprecation. Luettavissa: <https://docs.uipath.com/overview/other/latest/overview/deprecation-timeline>. Luettu 30.9.2024.

Friends. 2024. Luettavissa: <https://friends.com/ipaas>. Luettu 21.10.2024.

Gartner. 2024. Hyperautomation. Luettavissa: <https://www.gartner.com/en/information-technology/glossary/hyperautomation>. Luettu 25.11.2024.

GitHub Articles. 2024. What is version control? Luettavissa: <https://github.com/resources/articles/software-development/what-is-version-control>. Luettu 23.4.2024.

Hammarberg, M. & Sunden, J. Manning Publications. 2014. Kanban in Action.

Handlebars. 2024. Luettavissa: <https://handlebarsjs.com/guide/>. Luettu 21.10.2024.

IBM. 2024. What is robotic process automation (RPA)? Luettavissa: <https://www.ibm.com/topics/rpa>. Luettu 01.11.2024.

- Kannan, S. 2023. Never Upgrade Again: Transitioning UiPath Products Before End-of-Support Dates. Luettavissa: <https://www.linkedin.com/pulse/never-upgrade-again-transitioning-uipath-products-before-kannan/>. Luettu 20.9.2024.
- Kappagantula, S. 2023. UiPath RPA Architecture – A Deconstruction of the Components of UiPath. Luettavissa: <https://www.edureka.co/blog/uipath-rpa-architecture/>. Luettu 9.10.2024.
- Langmann, C. & Turi, D. 2022. Robotic Process Automation (RPA) -- digitization and automation of processes: prerequisites, functionality and implementation using accounting as an example.
- Loeliger, J. & Ponuthorai, P. M., O'Reilly Media, Inc, 2022. Version Control with Git, 3rd Edition.
- Loshi, D. & Powell, R. Morgan Kaufmann 2003. Business Intelligence.
- Microsoft. 2024. What is Power BI? Luettavissa: <https://learn.microsoft.com/en-us/power-bi/fundamentals/power-bi-overview>. Luettu 01.11.2024.
- Microsoft NuGet. 2024. Microsoft Learn. Luettavissa: <https://learn.microsoft.com/en-us/nuget/what-is-nuget>. Luettu 23.9.2024.
- Mäntyneva, M. Kauppakamari, 2016. Hallittu projekti: jäntevästä suunnittelusta menestykselliseen toteutukseen.
- Padamkar, P. 2023. UiPath Architecture. Luettavissa: <https://www.educba.com/uipath-architecture/>. Luettu 9.10.2024.
- Richards, M. O'Reilly Media, Inc. 2015. Software Architecture Patterns.
- Sahgal, S. Packt Publishing 2023. RPA Solution Architect's Handbook: Design Modern and Custom RPA Solutions for Digital Innovation.
- Sulava. 2023. Mikä se Azure oikein on? Luettavissa: <https://sulava.com/pilvi-infrastruktuuuri/mikase-azure-oikein/>. Luettu 1.11.2024.
- UiPath. 2024. Product. Luettavissa: <https://www.uipath.com/product>. Luettu 01.11.2024.
- UiPath. 2024. eGuide to the six automation scenarios. Luettavissa: <https://www.uipath.com/hubfs/Whitepapers/eGuide%20to%20the%20six%20automation%20scenarios.pdf>. Luettu 25.11.2024.
- Vitt, E. & Luckevich, M. & Misner S. Microsoft Press 2008. Business Intelligence.