

2020

Automatisering van de Equipment Control



Dylan van Vuren
Samskip Services B.V.
5/10/2020
Versie 1.0

Referaat

Auteur: Dylan van Vuren

Afstudeerverslag van Dylan van Vuren, studerende aan de Haagse Hogeschool, HBO-ICT.
Opleiding: Software Engineering.

Rotterdam, Samskip Services B.V., Oktober 2020.

Het afstudeerverslag beschrijft de procesgang van de automatisering van het Equipment Control van het bedrijfsproces binnen Samskip. Het project betreft de automatisering van het proces van het verwerken van container voorraad en -verplaatsingsrapporten met als doel de werklust te verlagen voor de equipment-control afdeling en de datakwaliteit en beschikbaarheid te verbeteren voor de gehele organisatie.

Het belang voor de organisatie van dit project is dat er meer inzichten verkregen worden uit de aangeleverde data. De daarop gebaseerde beslissingen kunnen uiteindelijk zorg gaan dragen voor een effectiever en efficiënter werkproces.

Deze afstudeeropdracht ondersteunt het doel om de datakwaliteit en -beschikbaarheid te verbeteren.

Descriptoren

- Afstudeerverslag
- Robotic Process Automation
- Kryon Systems
- Logistiek
- Tekstanalyse
- Unstructured Data
- Python
- Jupyter
- SQL

Voorwoord

Dit verslag beschrijft de afstudeeropdracht die ik heb uitgevoerd in het kader van mijn afstuderen aan De Haagse Hogeschool. De opdracht is uitgevoerd tussen februari 2020 en september 2020 in opdracht van Samskip Services B.V. De afstudeeropdracht betreft het automatiseren van het equipment control bedrijfsproces.

Het was een zeer uitdagende, maar vooral interessante opdracht om een bedrijfsproces van begin tot eind te automatiseren. De logistieke wereld was voor mij nog een compleet onbekend terrein en ik heb hier heel veel over kunnen leren. Ik kijk met veel plezier terug naar de afgelopen maanden waar ik hele leuke collega's heb mogen leren kennen, en veel plezier heb gehad in het werk zelf.

Graag wil ik Chris van Veelen in het bijzonder bedanken voor de hele fijne afstudeerplek, de uitdagende opdracht, en de continue ondersteuning en begeleiding.

Ook wil ik al mijn collega's in team digital solutions bedanken voor hun adviezen en hulp tijdens de afstudeerperiode.

Ik ben inmiddels in dienst getreden bij Samskip, waar ik verder werk aan de afstudeeropdracht, en ook andere robotische processen.

Dylan van Vuren

Rotterdam, 5 oktober 2020

Inhoudsopgave

1.	Inleiding	5
1.1	Organisatiebeschrijving	5
1.1.1	Team Digital Solutions	5
1.2	Probleemstelling	5
2.	Vorbereidingen	7
2.1	Onboarding	7
2.1.1	Kennismaking	7
2.1.2	Trainingen van Samskip	7
2.1.3	Kryon RPA Acadamy	8
2.2	Plan van Aanpak	11
2.2.1	Doelstellingen project	11
2.2.2	Cruciale succesfactoren	11
2.2.3	Aanpak	11
	Process Definition Document	11
	Solution Design Document	12
	Testplan & testrapport	12
	Robotscript en werkmethodek	12
2.2.4	Risicoanalyse	13
3.	Oplossingsmethode	14
3.1	Oplossingsmethode rapporten uitlezen	16
3.1.1	Artificial Intelligence	17
3.1.2	Tekstanalyse via scripts	18
3.1.3	Voor elk rapport een wizard maken	18
3.1.4	Keuze: Tekstanalyse via scripts	18
4.	Werkzaamheden	19
4.1	Meelopen met equipment control	19
4.2	De oplossing	20
4.2.1	Analyse huidige databronnen	20
4.2.2	Voorlopig ontwerp RPA/Python oplossing	22
4.3	Minimum Viable Product	25
4.3.1	Uitlezen van E-mails	25
4.3.2	Python tekstanalyse	27
4.3.2.1	Datum en tijd vinden	28
4.3.2.2	Containernummer vinden	30

4.3.2.3 Sleutelwoorden vinden	30
4.3.2.4 Statuscode bepalen	31
4.3.2.5 depot code bepalen	31
4.3.3 Verwerking van de resultaten	32
4.3.3.1 Verwerking van stock	32
4.3.3.2 Verwerking van moves	33
4.3.4 Rapportering	35
4.3.5 Testen	35
4.3.5.1 Unit tests	36
4.3.5.2 User acceptance test	37
4.3.6 Het uitrollen	38
5. Evaluatie	40
5.1 Evaluatie van de producten	40
5.1.1 Plan van Aanpak	40
5.1.2 Process Definition Document	40
5.1.3 Solution Design Document	41
5.1.4 Robotscript	41
5.1.5 Testplan	42
5.2 Evaluatie van de gekozen aanpak	43
5.3 Gedemonstreerde beroepstaken	44
5.3.1 Beroepstaak A1	45
5.3.2 Beroepstaak C6	45
5.3.3 Beroepstaak D14	45
5.3.4 Beroepstaak D15	45
5.3.5 Beroepstaak E18	45
5.3.6 Beroepstaak Gc	46
5.3.7 Beroepstaak Gf	46
6. Literatuur	47
7. Woorden en Afkortingenlijst	48

1. Inleiding

1.1 Organisatiebeschrijving

Samskip is een Nederlands bedrijf, waarvan het hoofdkwartier zich in Rotterdam bevindt. Het bedrijf is in 1990 begonnen in IJssland. Sindsdien is het bedrijf voortdurend gegroeid en hebben ze inmiddels het grootste multimodale transportnetwerk van Europa.

De voornaamste diensten die Samskip levert, is het vervoeren van containers van deur tot deur. Dit kan via de zee op een van hun 20 schepen, maar ook via de rivier, met de trein, of met vrachtwagens. Samskip vervoert ongeveer 850 duizend containers per jaar.

Samskip probeert zo duurzaam mogelijk te handelen. Dit wordt gedaan door zoveel mogelijk gebruik te maken van schepen en treinen. Wellicht is het niet altijd de goedkoopste oplossing, maar wel vaak de meest duurzame.

1.1.1 Team Digital Solutions

De afdeling ICT valt onder Samskip Services B.V. en ondersteunt de primaire activiteiten van de Samskip groep. Dit wordt gedaan door verschillende reeds werkzame software systemen binnen het bedrijf te onderhouden en nieuwe systemen te ontwikkelen.

Het team Digital Solutions richt zich met name op onderhoud en innovatie met data. Binnen dit team was mijn afstudeerstage.

Het afgelopen jaar richtte het team zich op een nieuw dataplatform. In de oude situatie bij Samskip waren er veel verschillende systemen met hun eigen databases. Dit bleek moeilijk onderhoudbaar en maakte nieuwe ontwikkelingen moeilijk. Opvallend aan de oude databases was dat het heel moeilijk te zien is, wat de relaties tussen tabellen zijn. Zo worden er veel nietszeggende attribuutnamen gebruikt, hanteerden de databases geen uniforme datastandaarden en is er geen, of zeer weinig, documentatie beschikbaar.

Inmiddels is overgestapt naar een Microsoft Azure cloud omgeving en er wordt gewerkt aan het opzetten van een centrale datawarehouse oplossing, die gebruikt kan worden voor alle systemen. Met deze nieuwe oplossing wordt dus alle data in een centrale plek opgeslagen. Vanuit deze plek worden alle rapportages gemaakt. Eveneens worden de systemen weer gevuld met (externe) data.

Er wordt binnen Samskip gebruik gemaakt van Robotic Process Automation (RPA) van Kryon Systems. Het onderhoud aan dit systeem valt ook onder het team Digital Solutions. Het doel van dit project is om verschillende inefficiënte processen binnen het bedrijf te automatiseren met software robots. Een aantal jaren geleden is dit project bij Samskip opgezet, echter door gebrek aan mankracht sinds een jaar niet voldoende onderhouden. Voortzetting van het RPA-project zou tot mijn takenpakket gaan horen.

1.2 Probleemstelling

De innovaties die het team maakt, worden allemaal gedaan op basis van data. Hiervoor is niet alleen een goede data infrastructuur nodig, maar eveneens het hebben van de juiste data. Helaas klopt niet alle data binnen de database.

De EquipmentTracking tabel van 'Doris', een Oracle database, wordt gebruikt om de lokatie en de status van de containers van Samskip op te slaan. Deze data wordt gebruikt in rapporten die planners kunnen gebruiken om bijvoorbeeld te weten waar lege containers staan die ze kunnen gebruiken. Elke keer als een container een 'move' maakt, komt er een entry in deze tabel. Een move betekent dat een container een depot, of terminal, binnenkomt of uitgaat. De move bevat belangrijke informatie, zoals datum, tijdstip, containernummer, richting (binnenkomend/uitgaand), of de container vol of leeg is en soms extra informatie, bijvoorbeeld of de container beschadigd is.

Deze data wordt door de meeste depots en terminals automatisch verzonden via een EDI-bericht. Dit komt meestal ook gelijk goed in de database te staan.

Waar het misgaat is wanneer het EDI bericht niet goed wordt doorgegeven, of er door depots helemaal geen EDI gebruikt wordt. Met deze depots is de afspraak gemaakt dat ze deze informatie via rapporten opsturen naar de equipment control afdeling. Deze afdeling houdt de statussen van containers bij en heeft ook als taak om deze rapporten handmatig in de database te zetten. Alhoewel er wel veel kennis van het bedrijf nodig is, is het proces van het handmatig invoeren van rapporten in de database erg repetitief en saai. Equipment control doet er ongeveer 3,5 uur per dag over om alle rapporten van die dag te verwerken. Het probleem bij deze manier van rapporteren is bovendien dat ze per depot verschillende structuren hebben. Ze bevatten wel altijd een tabel, maar de bestanden zijn in verschillende formaten (.pdf, .xls, .doc, .rtf, html). Verder zijn veel buitenlandse rapporten niet in het engels. Voorts staan in de meeste rapporten niet alle data die nodig zijn om een move in de database te zetten. Zo sturen de meeste rapporten bijvoorbeeld geen tijdstip van een move op, terwijl dit wel belangrijk is om de chronologische volgorde van moves te kunnen bepalen. Ook kan vaak niet worden gezien aan het rapport of een container via de zee of via het land is binnengekomen. Door ontbrekende data moet equipment control aannames doen, waardoor er regelmatig fouten in de database komen te staan.

De voornaamste reden voor de keuze om bovenbeschreven proces te automatiseren, is om te voorkomen dat er teveel fouten in de database komen.

Een ander probleem van deze manier van rapporteren en het handmatig verwerken is de data-beschikbaarheid. Doordat de handmatige verwerking van rapporten in de ochtend plaatsvindt, komt data uit rapporten die vanaf begin van de middag verstuurd worden pas de volgende dag in de database te staan, oftewel, er treedt dan vertraging op in de beschikbaarheid van data uit de centrale database.

Na automatisering zal deze vertraging weg zijn, omdat een robot 24 uur per dag beschikbaar is om rapporten te verwerken. Er zal nog wel enige vertraging zijn tussen het moment dat een move gebeurt en het moment dat het rapport verstuurd wordt, maar deze vertraging is in ieder geval al kleiner.

Er zijn dus genoeg redenen om dit proces te automatiseren met behulp van RPA. In het volgende hoofdstuk wordt er dieper ingegaan op de gekozen oplossingsmethode. In het Process Definition Document (Bijlage 2) staat een uitgebreide analyse van het equipment control bedrijfsproces.

2. Voorbereidingen

Er worden enkele voorbereidingen getroffen, voorafgaand er begonnen wordt met ontwikkelen. Ten eerste vindt een onboarding plaats, waar ik meer te leren krijg over Samskip en de logistieke wereld. Vervolgens wordt er ook een RPA developer cursus van Kryon Systems gevolgd, zodat duidelijk wordt wat de mogelijkheden zijn bij het ontwikkelen van een robotscript. Dit zal helpen bij het ontwerpen van een oplossing van de hiervoor omschreven probleemstelling. Tot slot is er ook een Plan van Aanpak opgesteld. Hierin wordt een aanpak gekozen, zodat er gestructureerd te werk kan worden gegaan.

2.1 Onboarding

In de eerste weken van de afstudeerstage wordt er vooral gefocust op het leren kennen van Samskip. Het doel is om zoveel mogelijk te leren over het bedrijf. Het is belangrijk dat ik het bedrijf, en vooral het equipment control proces, goed begrijp.

Het is van groot belang dat ik leer werken met Kryon RPA omdat dit de software is waarin de wizard ontwikkeld zal worden.

2.1.1 Kennismaking

Op de eerste dag wordt er kennis gemaakt met de collega's binnen de afdeling ICT van Samskip. Het team Digital Solutions bestond uit vijf collega's waaronder business Intelligence specialisten en een data scientist.

Ik zelf heb de rol van Robotic Process Automation Specialist. Dat betekent dat ik alle taken uitvoer die met Robotic Process Automation te maken hebben. Tijdens de afstudeerperiode zal ik mij echter alleen richten op het afstudeerproject.

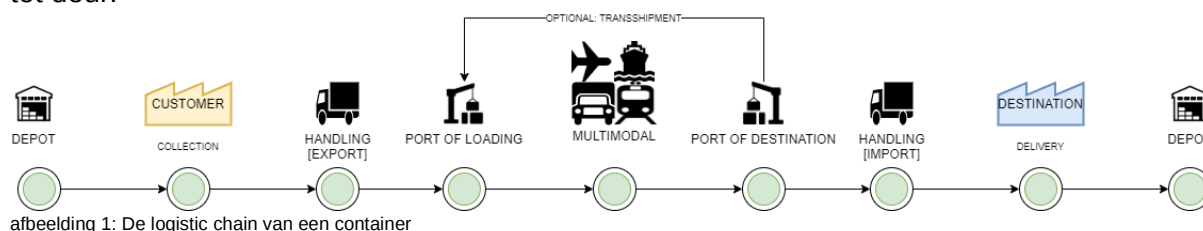
Nadat ik ben begonnen zijn er nog drie andere collega's bijgekomen met de rollen Data Steward, Business intelligence developer, en een Data engineer.

Voor mijn afstudeeropdracht werk ik vooral samen met Chris omdat hij ervaring heeft met RPA en de opdrachtgever en manager is. Voor vragen over de database kan ik bij de Business Intelligence Specialisten terecht. Voor vragen over python kan ik om hulp vragen aan de data scientist. Met de data steward kan ik om hulp vragen over datadefinities. Voor technische ondersteuning kan ik bij de samskip IT support terecht.

2.1.2 Trainingen van Samskip

Samskip biedt zelf trainingen aan nieuwe werknemers. Deze trainingen vertellen over de geschiedenis van het bedrijf, de waarden van het bedrijf, en over de business zelf. Deze trainingen bestaan uit verschillende filmpjes en quizjes, die ervoor zorgen dat nieuwe werknemers snel de business kunnen begrijpen.

Samskip biedt Deur tot Deur en kade tot kade transport aan. Dit betekent dat Samskip alle stappen van het transport regelt. Zie afbeelding 1 voor een voorbeeld van transport van deur tot deur.



De scope van dit project zijn de rapporten die via de mail geleverd worden door de depots (opslag voor lege containers), en de Ports (terminals).

Samskip maakt gebruik van schepen, binnenvaartschepen, treinen, en trucks. Samskip past zich aan op de benodigdheden van de klant. Zie hieronder het transportnetwerk van Samskip (afbeelding 2).

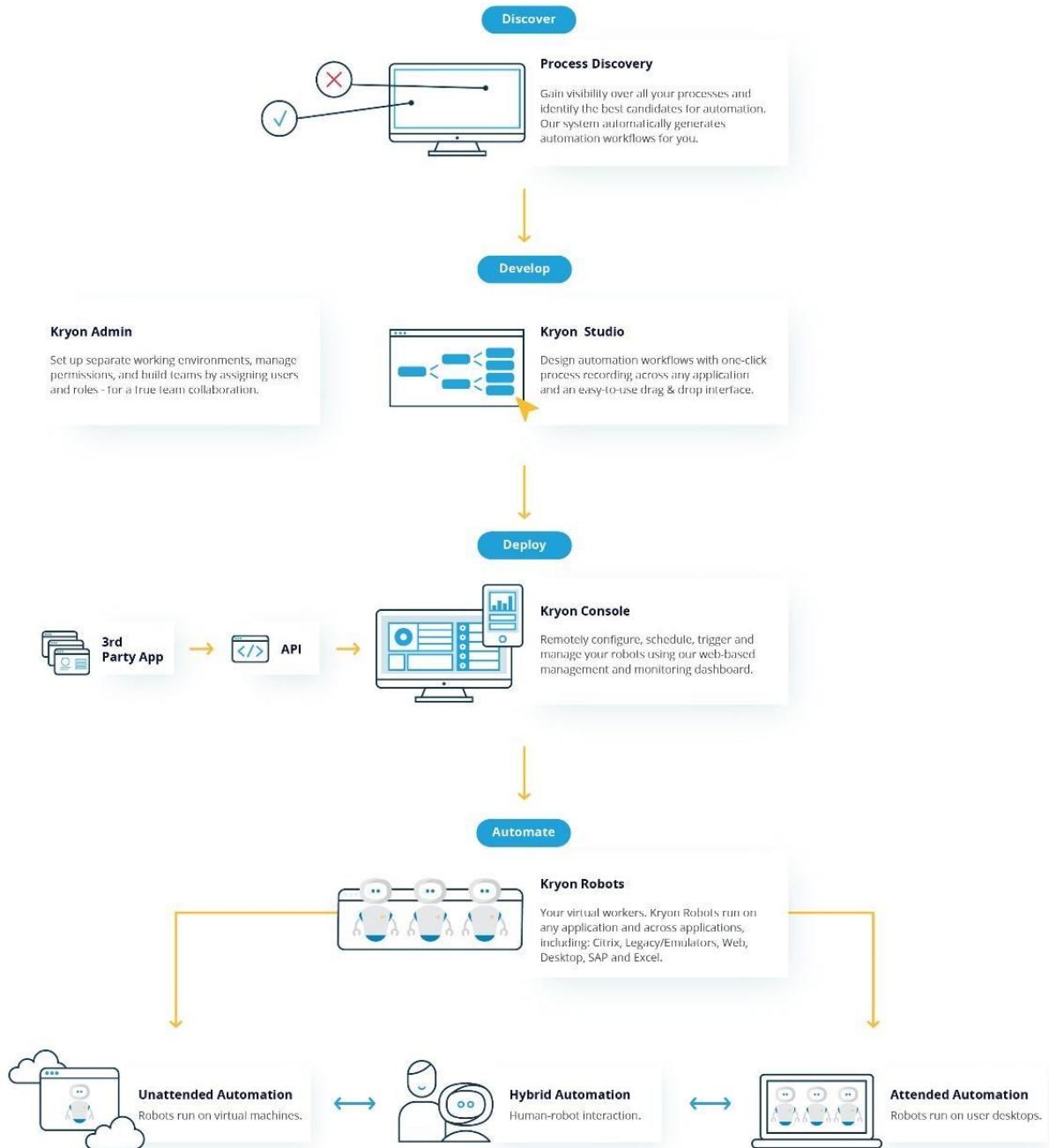


Afbeelding 2) Het transportnetwerk van Samskip.

2.1.3 Kryon RPA Academy

De softwareleverancier voor het ontwikkelen en gebruiken van robots heet Kryon Systems. Kryon biedt een cursus aan haar klanten om te leren omgaan met de software. In deze cursussen wordt bijna alles behandeld, maar de nadruk ligt bij het ontwikkelen van RPA-oplossingen. Ik heb voor deze afstudeerstage al eerder gewerkt met UiPath, een concurrerende RPA vendor. Deze ervaring heeft erg veel geholpen met het begrijpen van Kryon, waardoor ik binnen een week de gehele cursus kon afmaken.

In de cursus heb ik geleerd over de architectuur van Kryon. Deze informatie is vooral heel belangrijk voor het uitrollen van wizards. Zie de afbeelding hieronder (afbeelding 3) voor meer informatie over de Kryon architectuur.



Afbeelding 3) Kryon Architectuur

Kryon Studio is de development omgeving. In de studio ontwikkel je de wizards die de robot later kan uitvoeren. De studio heeft een drag-and-drop interface. Dat betekent dus dat de code niet zelf geschreven hoeft te worden. In plaats daarvan wordt er een blok gesleept en geconfigureerd. Om een variabele te zetten wordt er een “Set Value” blok versleept naar de juiste positie. Dit blok moet vervolgens geconfigureerd worden met de variabelnaam en de waarde.

Dit is voor beginners veel makkelijker dan het zelf schrijven van code, maar persoonlijk vind ik dit geen fijne manier van werken. Door te werken met vooraf vastgestelde blokken kan je vrij gemakkelijk geavanceerde functies gebruiken zoals database calls of e-mails uitlezen. Aan de andere kant ben je afhankelijk van de code die achter deze blokken zit, waardoor het

niet altijd even makkelijk is om te begrijpen wat er precies achter de schermen gebeurt. Het beperkt de aanpasbaarheid van code, en maakt het debuggen een stuk lastiger. Hier worden later in het verslag voorbeelden van gegeven.

Kryon Admin is een administratief programma waarin verschillende gebruikers en permissies kunnen worden ingesteld. Binnen dit programma worden ook applicaties en webapplicaties gewhitelist die Kryon robots mogen gebruiken (dit is een veiligheidsmaatregel).

Verder wordt Kryon Admin gebruikt om op een veilige manier inloggegevens op te slaan voor de applicaties die geautomatiseerd worden. Ook worden er binnen Kryon Admin de database connecties en query's gedefinieerd. Deze functionaliteiten kunnen ook binnen een wizard zelf worden gedefinieerd, maar het is veiliger om dit binnen Admin te doen. Ook is het op deze manier overdraagbaar.

Kryon Console wordt voornamelijk gebruikt om de productieomgeving te beheren. Het bevat een dashboard waarin belangrijke informatie te vinden is, zoals een run-historie en of er fouten zijn opgetreden tijdens het draaien van een proces.

Verder kunnen er via de console processen gestart worden via triggers. Een voorbeeld van zo'n trigger zou zijn: "Als er een email binnenkomt op mailbox <mailbox info> met subject "rapport", start wizard <wizardnaam> op robot <robotnaam>".

Er zijn veel verschillende manieren om een proces te starten. Dit kan door middel van:

- Handmatig
- Time trigger (bijvoorbeeld: elk uur, of op de 7^{de} van elke maand)
- Email trigger (zie bovenstaand voorbeeld)
- Database trigger (bijvoorbeeld: als er een insert wordt gemaakt in een bepaalde tabel)
- File trigger (bijvoorbeeld: Als er in een bepaalde map een bepaald bestand wordt geupdate)
- Folder trigger (bijvoorbeeld: als een bepaalde folder wordt gedelete)
- API (Om via een externe applicatie processen te starten)
- Andere processen (Binnen een wizard kunnen andere processen gestart worden)

Kryon Robot

De robot is het programma dat daadwerkelijk processen uitvoert. De robot krijgt een commando binnen van de console om een bepaalde wizard uit te voeren. Als de robot een fout tegenkomt, of klaar is, wordt dit gerapporteerd naar de console.

Samskip heeft op dit moment 5 robot licenties. Iedere robot heeft een eigen Virtual Machine waarop de applicaties zijn ingesteld die de robot nodig heeft. Hierdoor ontstaan er geen conflicten als er meerdere robots tegelijk werken.

Robots kunnen op drie verschillende manieren werken. Ze kunnen unattended werken, dat betekent dat de robot op de achtergrond automatisch processen uitvoert. Ze kunnen ook attended zijn, dat betekent dat ze geïnstalleerd zijn op de computer van een gebruiker. Deze robots worden door de gebruiker gestart en werken samen met de gebruiker. Er bestaat ook een hybride modus, waarbij de robot op de achtergrond werkt, maar bijvoorbeeld om input kan vragen bij de gebruiker. Op dit moment worden er alleen maar unattended robots gebruikt binnen Samskip.

2.2 Plan van Aanpak

In het Plan van Aanpak worden een aantal zaken behandeld die belangrijk zijn voordat er wordt begonnen met de ontwikkeling van de oplossing. Het volledige plan van aanpak is te vinden in bijlage 1. De probleemstelling van het plan van aanpak komt al voor in dit verslag. Deze paragraaf vat de onderwerpen samen die nog niet zijn benoemd in dit verslag.

2.2.1 Doelstellingen project

Het doel van de opdracht is om een softwarerobot te ontwikkelen die het data-entry proces in zijn geheel of gedeeltelijk kan overnemen. Ook kan de robot controleren of de database consistent is. Aan het einde van de afstudeerperiode zou deze softwarerobot 24/7 beschikbaar moeten zijn om email van verschillende depots te lezen en foutloos te verwerken.

2.2.2 Cruciale succesfactoren

We kunnen weten of het project succesvol is door cruciale succesfactoren op te stellen. Deze zijn gebaseerd op de bovengenoemde doelstellingen.

Beschikbaarheid

Om ervoor te zorgen dat de robot altijd beschikbaar is om rapporten te verwerken, zal de robot op een virtual machine draaien. Verder moeten alle excepties opgevangen worden op een manier dat het programma nooit crasht. Het is belangrijk dat het inventarisatierapport (Stock Report) zo te allen tijden zo recent mogelijk is, omdat deze informatie belangrijk is voor het plannen van verplaatsingen van containers.

Datakwaliteit

Om de datakwaliteit te verbeteren moet de robot zo geprogrammeerd worden dat foute invoer niet mogelijk is. Dit zal onder andere worden bereikt door een vertalingstabel te gebruiken om de data te normaliseren. Verder wordt er tijdens de ontwerpfase uitvoerig gekeken naar het as-is proces en worden zo veel mogelijk excepties en abnormaliteiten ontdekt, zodat deze gedekt kunnen worden in het solution design document. Als de robot er alsnog niet uitkomt (dit kan voorkomen als de data niet klopt), zal er een mail worden gestuurd naar de proceseigenaar met de data die de robot niet kan verwerken. De proceseigenaar kan vervolgens input leveren om de robot verder te trainen. Het product zal worden getoetst aan de hand van een testplan met acceptatietest.

2.2.3 Aanpak

In deze sub-paragraaf worden de activiteiten aangegeven die leiden tot het eindproduct. Ook wordt er ingegaan waarom er voor deze oplossing wordt gekozen. De reden voor een bepaalde oplossing kan voortkomen uit een cruciale succesfactor, de risicoanalyse, of een voorwaarde die is opgesteld door de opdrachtgever.

Process Definition Document

Het Process Definition Document (PDD) is een zeer gedetailleerde beschrijving van het proces dat wordt geautomatiseerd. Bij het automatiseren van een bedrijfsproces is het van belang dat alle mogelijke uitzonderingen zijn beschreven, zodat de robot minder fouten zal maken. In het document wordt de sequentie van stappen beschreven die deel uitmaken van het business proces. Het document beschrijft de relevante bedrijfsregels en de omgeving voor en na automatisering. Het document is noodzakelijk om het Solution Design Document te maken en om te zorgen dat de robot geen fouten zal maken.

Om het PDD te maken wordt er een tijd meegelopen met de SME (Subject Matter Expert). Hierdoor kan de afstudeerder vragen stellen en een beter inzicht krijgen van het proces. Deze stap is belangrijk om een goed PDD op te stellen, omdat er zo minder verwarring of

miscommunicatie zal plaatsvinden. Het PDD zal worden gecontroleerd door de opdrachtgever en de SME.

Solution Design Document

Het solution design document (SDD) is een ontwerpdocument van de softwareoplossing. Dit document bevat een gedetailleerd ontwerp van de softwareoplossing. Dit document wordt geüpdatet met elke verandering aan het geautomatiseerde proces. Het wordt zowel als ontwerpdocument en als documentatie gebruikt.

Als dit document wordt goedgekeurd door de opdrachtgever kan de development van het robotscript beginnen.

Testplan & testrapport

Er wordt een testplan opgesteld om de verschillende testmethodes te beschrijven. Deze tests zijn er om fouten in de oplossing te detecteren voordat het in productie gaat. Deze fouten kunnen een grote negatieve impact hebben op de beschikbaarheid en datakwaliteit, doordat het programma kan vastlopen als het een onbekende situatie tegenkomt.

De testresultaten zullen worden gedocumenteerd in het testrapport.

Robotscript en werkmethode

De daadwerkelijke oplossing zal in de vorm zijn van een robot script voor Kryon Robots. Dit script wordt ontwikkeld in Kryon Studio. Het robotscript zal ontwikkeld worden op basis van het Solution Design Document, en getest worden op basis van het testplan.

Als werkmethode is er gekozen voor een aantal elementen uit SCRUM en Waterfall. Doordat het huidige proces eerst volledig in kaart gebracht moet worden om een goede oplossingsmethode te kiezen, wordt de oplossingsmethode grotendeels van tevoren ontworpen. Alleen de exacte details van de implementatie worden niet van tevoren bepaald. Het vooraf maken van een ontwerp zijn typische kenmerken van Waterfall en levert het voordeel op dat er van tevoren gekeken wordt naar de uitdagingen in de oplossing, en hoe deze opgelost kunnen worden.

Er is gekozen om ook een aantal elementen van Scrum mee te nemen in de werkmethode. Zo vindt er een daily standup plaats, die ervoor zorgt dat de opdrachtgever nauw betrokken is met de ontwikkeling van de oplossing. Dit verlaagt het risico dat er iets wordt gemaakt dat niet goed is, of niet zal gaan werken.

Ook wordt er gebruik gemaakt van een digitaal scrumbord op JIRA om te helpen met het plannen en overzichtelijk maken van taken.

2.2.4 Risicoanalyse

Risico	Maatregel
Er is geen testomgeving beschikbaar	De database wordt tijdens de development niet geüpdatet met de outputdata van de robot. De entries worden in een .csv opgeslagen voor controle. Hierdoor zal de development niet de dagelijkse gang van zaken beïnvloeden, totdat we zeker weten dat alles goed werkt.
Het RPA-project doet niet wat het zou moeten doen	Het kan voorkomen dat het geautomatiseerde proces niet overeenkomt met wat het zou moeten doen. Het is hierdoor belangrijk dat het huidige proces goed wordt gedocumenteerd, en dat de process owner nauw betrokken blijft bij het ontwerp van de oplossing. Verder worden de ontwerpdocumenten goedgekeurd voordat de RPA-oplossing wordt ontwikkeld. Door met scrum in iteraties te werken, kunnen veranderingen vrij gemakkelijk plaatsvinden, en wordt er bij elke stap geverifieerd of het goed is.
Het proces blijkt toch niet RPA geschikt	Er is vooraf onderzoek gedaan, en tijdens het maken van het PDD wordt het proces tot in detail geanalyseerd. Een proces kan onmogelijk te automatiseren zijn als het veel ongestructureerde data bevat, of als het heel complex is en menselijke intelligentie nodig heeft. Als blijkt dat het proces niet mogelijk is, dan kan er gekeken worden hoe het proces veranderd zou kunnen worden om hetzelfde resultaat te bereiken. Daarnaast kan ervoor gekozen worden om een deel van het proces te robotiseren. Als zelfs dat niet mogelijk is dan wordt er snel naar een nieuw proces gezocht om alsnog te kunnen afstuderen.
Communicatieprobleem en met de Subject Matter Expert (SME) of Process Owner(PO)	De SME is de collega die daadwerkelijk het te automatiseren proces uitvoert. De process owner is de manager en verantwoordelijke voor dat proces. Samenwerking met deze collega's is zeer belangrijk, aangezien zij de informatie hebben die wij nodig hebben om het proces te automatiseren. Na de automatisering, zijn zij ook degene die met het veranderde proces moeten werken. De automatisering heeft dus een grote invloed op het dagelijks werk dat zij zullen doen. Hopelijk is dit een positieve invloed, aangezien het repetitieve, tijdrovende, simpele gedeelte geautomatiseerd wordt. Het kan echter zo zijn dat de oplossing niet gewaardeerd wordt. Het is belangrijk voor dit project om goed en duidelijk te communiceren met de SME en PO. Zodra er een plan is voor automatisering, zullen zij betrokken zijn bij de bespreking zodat eventuele bezwaren gehoord kunnen worden. Op deze manier kan er gewerkt worden aan een oplossing waar iedereen de vruchten van plukt.

3. Oplossingsmethode

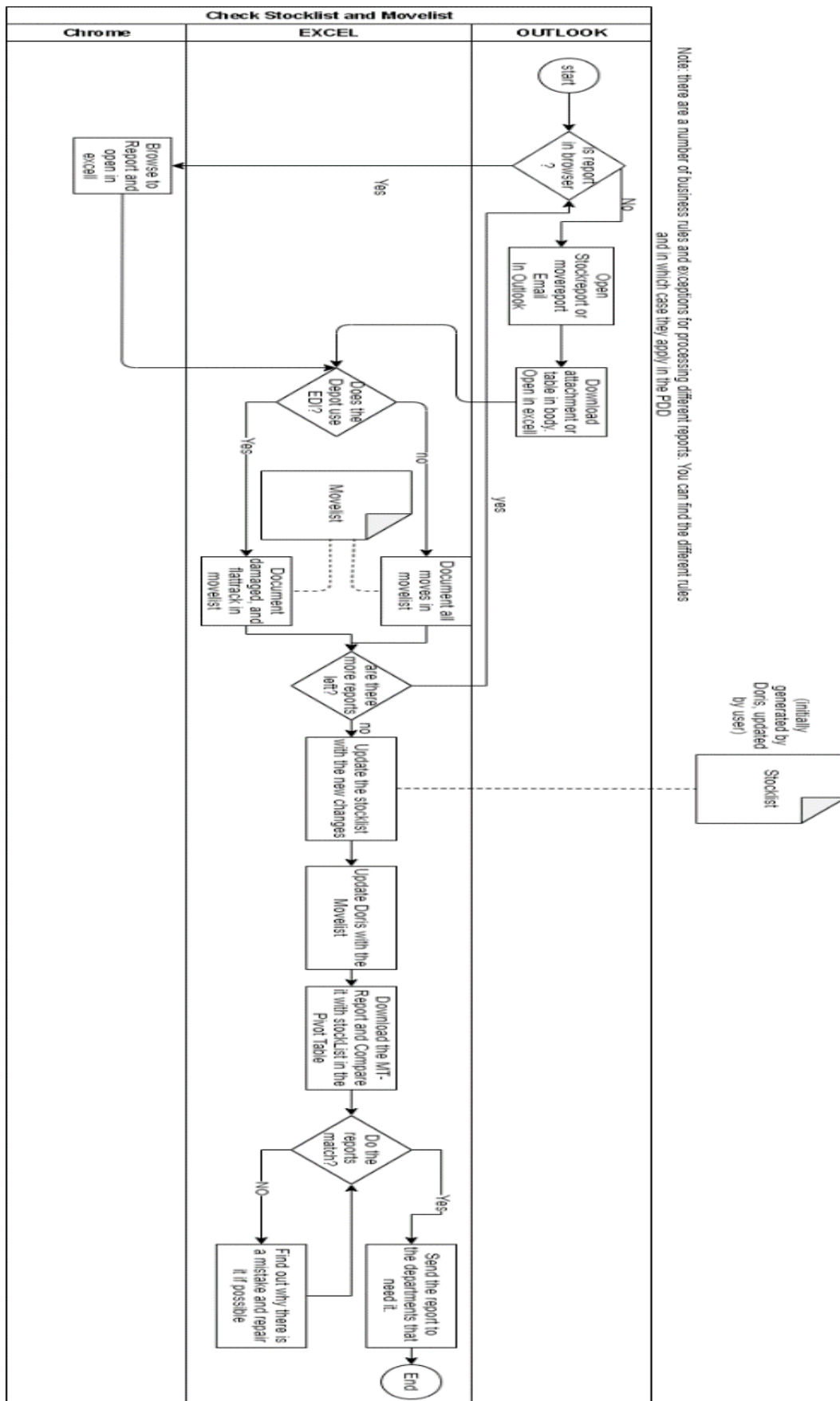
In dit hoofdstuk worden de mogelijke en gekozen oplossingsmethoden besproken. Eveneens wordt de gekozen aanpak verdedigd.

Er is al van tevoren besloten dat de oplossing met RPA van Kryon wordt gemaakt. De reden hiervoor is dat de andere procesautomatisering projecten van Samskip ook in Kryon worden gemaakt. Het voordeel van Kryon is dat er makkelijk scripts aangeroepen kunnen worden: API's en webservices, emails uitlezen, en user interfaces automatiseren. Hierdoor is het de beste oplossing wanneer het gaat om het kopiëren van menselijke handelingen in GUI's.

Nadelig is echter de performance van Kryon. Het is langzaam en crasht vaak. Sommige features werken niet helemaal intuïtief, waardoor debuggen langer duurt. Het veranderen van RPA vendor zou echter een hele grote geld en tijdsinvestering zijn. Kryon zal dus het RPA-platform zijn waarop de applicatie wordt ontwikkeld.

Om tot de juiste oplossingsmethode te komen, is het belangrijk om eerst het huidige bedrijfsproces goed in kaart te brengen. Omdat er binnen Samskip geen document bestaat waarin voornoemd proces precies uitgelegd wordt, is in samenwerking met de Subject Matter Expert (SME) van de afdeling Equipment Control een Proces Definition Document (PDD) opgesteld.

In afbeelding 4 op de volgende pagina staat een flowdiagram die het huidige equipment control proces visualiseert.



Atbeelding 4)) Visualisatie van het huidige equipment control proces. In bijlage 2 (PDD) staat een uitgebreide toelichting.

Er komen via de email rapporten binnen van verschillende depots en terminals in het netwerk van Samskip. In deze rapporten staan de stock (de containers die op dat moment worden opgeslagen op die locatie), en de moves (De events als een container een locatie binnenkomt of uitgaat).

Voor elk rapport waarin moves staan, worden deze moves in de database gezet. Hierbij wordt gebruik gemaakt van een excel upload sheet. Dit is een Excel bestand waarin alle benodigde data wordt opgenomen, die vervolgens wordt geüpload naar een applicatie die de moves in de database zet. De stock van depots en terminals worden ook bijgehouden. en zodra alle moves verwerkt zijn in de database wordt er gekeken of de data in de database overeenkomt met de data uit de stock rapporten. Deze controle stap is heel belangrijk omdat er altijd wel uitzonderingen zijn. Dit kan bijvoorbeeld komen omdat een container inmiddels is aangekomen bij een depot waar de moves automatisch worden verwerkt. Dit kan sneller zijn dan het handmatig invoeren, waardoor de data niet meer klopt. Voor elk geval waar de database niet overeenkomt met de stock rapporten, wordt er uitgezocht welke data correct is en wordt dit indien nodig bijgewerkt in de database.

De stock van depots en terminals worden tegelijk ook bijgehouden. Nadat de moves verwerkt zijn in de database, zou de stock in het rapport overeen moeten komen met de stock die de database heeft voor die locatie.

Zodra alle moves in de database verwerkt zijn, wordt er gecontroleerd of de data van de moves in de database overeenkomen met de data uit de stock rapporten. Deze controlestap is noodzakelijk om er voor zorg te dragen dat de data in de centrale database correct is.

Voor elk geval waar de database niet overeenkomt met de stock rapporten moet uitgezocht worden welke data van de twee systemen correct is en alsnog verwerkt worden in de database.

Als bijvoorbeeld een container vertrekt vanuit een depot waar handmatig gerapporteerd wordt en aankomt bij een depot waar moves automatisch worden verwerkt, kan er een verschil ontstaan in de snelheid waarin data wordt verwerkt. Zo kan het voorkomen dat door dit verschil data stock rapporten niet overeenkomen met gegevens uit de centrale database. In zulke gevallen dient dus elke keer uitgezocht te worden welke van de twee data correct is.

3.1 Oplossingsmethode rapporten uitlezen

Een groot probleem voor de automatisering van dit proces is dat de binnenkomende stock rapporten en move rapporten geen gestandaardiseerde structuur hebben. Ook is de data zelf niet gestandaardiseerd. Dit maakt het moeilijk om deze rapporten uit te lezen, omdat er een bepaald niveau van intelligentie nodig is om te weten wat er precies staat. Zo ontbreekt er vaak bepaalde data, waardoor deze data op een andere manier moet worden gegenereerd. In de afbeelding hieronder (afbeelding 5) is een voorbeeld van een bijzonder moeilijke rapportage te zien. In dit rapport ontbreekt er data over het tijdstip, of de container vol of leeg is, of de container schade heeft of in orde is, en of het om stock of moves gaat. Ook is de datum ingewikkeld voor een computer om uit te lezen omdat het de spaanse afkorting voor augustus gebruikt, en de header van de tabel in het spaans is.

From: rafael [REDACTED]
 Sent: Thursday, August 2, 2018 8:04 AM
 To: 'BILBAO GENERAL'; 'J'; 'MICHAEL'; RTM MCL EquipmentControl - rec; 'ROSANA'; 'YOLANDA'
 Subject: FAX



Fecha:		02-ago-18				
Movimientos del día:		01-ago-18				
Nº Contenedor	Tipo	F/Entr.	F/Sal.	Observaciones	Transportista	Matricula
STOCK.						
CNEU-451349-6	45	24-07-18		H/C		
SANU-796640-1	45	24-07-18		H/C		
SANU-752030-1	45	31-07-18		H/C		
SANU-770083-3	45	01-08-18		H/C		
SANU-799843-5	45	01-08-18		H/C		

Afbeelding 5) Voorbeeld ingewikkeld stock rapport.

Om te bepalen op welke manier deze rapporten het beste uitgelezen kunnen worden, is een voorbeeld rapport opgevraagd van elk depot waar Equipment Control rapporten van ontvangt. Op basis van deze rapporten zijn een paar potentiële oplossingsmethoden ontdekt voor het hierboven omschreven probleem, die ik hieronder verder zal toelichten.

3.1.1 Artificial Intelligence

Artificial Intelligence (AI) kan gebruikt worden om de context van ongestructureerde data te herkennen. Microsoft biedt met haar Azure Cognitive Services toegankelijke AI aan; een van de opties die interessant voor Samskip is, is de optie om tabellen uit te lezen. Met een paar rapporten is uitgetest hoe goed Azure Cognitive Services werkt. Hieruit blijkt dat er veel meer nadelen dan voordelen zijn.

Voordelen:

- Minder ontwikkelingstijd.

Nadelen:

- Extra kosten voor het gebruik van deze service.
- Werkt alleen als er trainingsdata voor elk type rapport wordt geleverd. Dit kost veel tijd omdat deze data gelabeld moet worden.
- Maakt regelmatig fouten doordat het volledig visueel werkt. Hierdoor leest het zo nu en dan een getal of letter verkeerd. Dit kan grote gevolgen hebben voor de datakwaliteit.

Uit de resultaten van de test blijkt dat deze service niet goed genoeg werkt voor dit probleem. De belangrijkste reden om het niet te gebruiken is de foutgevoeligheid. Het gebruikt visuele analyse om de structuur en tekst te extraheren van het bestand. Door tekst visueel uit te lezen, komen er in elk rapport wel een paar fouten voor. Een van de doelstellingen van dit project is om de datakwaliteit te verhogen. Als er verkeerde input data verwerkt wordt, zal de datakwaliteit alleen maar lager zijn met als gevolg dat er onjuiste gegevens worden verstrekt aan de eindgebruiker. Alhoewel het een tabel goed kan herkennen en uitlezen, is het niet slim genoeg om afkortingen uit verschillende Europese talen te herkennen, en al zeker niet als het een afkorting is uit de logistieke vaktaal.

3.1.2 Tekstanalyse via scripts

De tekst kan ook geanalyseerd worden door zelfgemaakte scripts in Python. Deze scripts kunnen de relevante data uit de tabellen halen. Deze data wordt op een uniforme manier uitgelezen uit de rapporten. Als de structuur van een rapport zodanig verschilt dat de uniforme aanpak niet werkt, dan kan er een uitzondering geprogrammeerd worden, zodat dit rapport anders wordt uitgelezen.

Omdat hier de context van de data ontbreekt, moet er voor elke variabele gedefinieerd worden hoe deze data er precies uit zou moeten zien. Dit om te voorkomen dat het niet verward kan worden met andere variabelen.

Enkele voordelen hiervan zijn:

- Een oplossing voor ieder rapport.
- Makkelijke uitbreidingen.
- Excepties kunnen makkelijk worden toegevoegd indien nodig.

Enkele nadelen zijn:

- Dit is een complexe oplossing.
- Er bestaat het risico dat de rapporten te verschillend zijn van elkaar, waardoor een generieke aanpak zoals deze niet mogelijk blijkt.

Om te bezien of deze oplossing een optie is om het betreffende probleem te ondervangen, is eerst onderzocht of de verschillende binnenkomende rapporten genoeg gemeen hebben. Er is aan de equipment control afdeling gevraagd om een voorbeeldrapport van alle verschillende depots op te sturen. Deze rapporten zijn vergeleken met elkaar.

3.1.3 Voor elk rapport een wizard maken

Een Wizard is een script dat uitgevoerd wordt door een RPA-software robot. RPA is een goede tool voor het uitlezen van pdf's, Excel, Word, etc. RPA kan worden gebruikt om via de User Interface(UI) van de rapporten een tekstanalyse uit te voeren. Dit kan door bijvoorbeeld door te zoeken naar de header 'containernummer' binnen een tabel. Vervolgens is dan bekend dat de data die eronder staat containernummers zijn.

Een voordeel van deze methode is:

- Unieke wizard voor ieder depot. Hierdoor is het makkelijker om bepaalde rapporten te automatiseren, aangezien er geen rekening gehouden hoeft te worden met formaten van andere rapporten.

Nadelen zijn:

- Hoge ontwikkeltijd. Als er 100 verschillende wizards gemaakt moeten worden, zal dit een enorme klus worden.
- Minder schaalbaar, omdat een nieuw rapporttype een nieuwe wizard nodig heeft.
- Minder robuust. Als het formaat van een rapport verandert, zal de wizard weer aangepast moeten worden. Dit zorgt voor meer onderhoud in de toekomst.

3.1.4 Keuze: Tekstanalyse via scripts

Uiteindelijk is besloten om de tekstanalyse via een aantal scripts te doen. Door goed naar de verschillende rapporten te kijken, zijn verbanden ontdekt waardoor het mogelijk zou kunnen zijn om een uniforme oplossing te maken. Natuurlijk zullen hier wel enkele excepties in voorkomen, maar deze zijn vrij gemakkelijk op te vangen. Deze scripts zullen worden aangeroepen door een wizard.

4. Werkzaamheden

Dit hoofdstuk beschrijft de werkzaamheden die hebben geleid tot de totstandkoming van het eindproduct en de gemaakte keuzes met motivatie. Het doel is om een duidelijk beeld te geven van de omvang en diepgang van de werkzaamheden die zijn uitgevoerd tijdens de afstudeerperiode. Dit hoofdstuk is opgesplitst op basis van de verschillende fases van het project, en vervolgens de verschillende onderdelen en subonderdelen. De hoofdstukken staan grofweg in chronologische volgorde van uitvoering.

4.1 Meelopen met equipment control

Het proces

Om het equipment control proces te vervangen met software-robots, is het belangrijk precies te weten wat dit proces inhoudt. Dit proces wordt gedocumenteerd in het PDD, ofwel Process Definition Document. Dit document is te vinden in de bijlage 2. Zeker voor grote projecten zoals deze. Op deze manier kan er worden gecontroleerd met de SME of het proces goed is begrepen, voorafgaand aan het ontwerpen van een oplossing. Er wordt voor het opstellen van de PDD een template gebruikt. Om het proces beter te begrijpen wordt er een aantal dagen meegelopen met de equipment control afdeling. Door mee te kijken, vragen te stellen, en ook het proces zelf uit te voeren, kan het proces gedefinieerd worden op papier. Het proces zoals equipment control het uitvoert is beschreven in bijlage 2, en samengevat in hoofdstuk 3.

Er zijn tijdens het meelopen een aantal belangrijke observaties over het proces gemaakt, namelijk:

- Bijna alle rapporten komen binnen via mail. Er zijn een paar rapporten die van een website gehaald moeten worden.
- De focus ligt vooral op het verwerken van data over lege containers. Door tijdsgebrek is het niet mogelijk om de data over volle containers ook te verwerken.
- De rapporten hebben onderling verschillende structuren, en hebben niet altijd dezelfde gegevens. Er ontbreken veel gegevens die equipment control wel moet verwerken.
- De bovengenoemde gegevens worden vaak gegokt. Bijvoorbeeld omdat equipment control weet dat deze locatie alleen lege containers heeft, waardoor ze alles als leeg kunnen verwerken.
- Tijdstippen van events worden vaak een standaardwaarde gegeven omdat deze gegevens lastig zijn om te achterhalen als ze niet in het rapport staan. Deze standaardwaarde hebben het doel om de containers in een chronologische volgorde in de database te zetten, en dus niet om een accuraat tijdstip te geven.
- de relevante informatie wordt door de SME heel snel uit het rapport gehaald. Dit is een kwestie van ervaring, aangezien de rapporten onderling verschillen van structuur. Dit is moeilijk te trainen aan een robot.

De menselijke kant

Een belangrijk onderdeel van RPA is de menselijke kant. Bij het automatiseren van een proces waar veel tijd in gaat, zou een manager misschien kunnen denken dat de afdeling na automatisering zonder de werknemer verder kan. Dat is echter niet de opzet van dit project. Het uiteindelijke doel is het verbeteren van de datakwaliteit, databeschikbaarheid, en tijdsbesparing. Er zou meer tijd vrijkomen voor andere taken waar de afdeling Equipment Control niet aan toe blijkt te komen, zoals het behandelen van excepties.

Het is van belang dat de SME zich goed voelt met dit project, zodat er een goede samenwerking kan plaatsvinden. Daarom zijn de doelstellingen van dit project besproken, en is de SME dichtbij betrokken bij het analyseren van het proces en het goedkeuren van de oplossing.

4.2 De oplossing

Na goedkeuring van het PDD door de SME en de opdrachtgever, wordt er begonnen aan het ontwerpen van de oplossing. Op dit moment is er een goed beeld geschetst van het bedrijfsproces. Nog even in het kort samengevat wat de robot moet doen:

- Inputs: Stock Rapporten en MoveRapporten
bron van inputs: Emails met rapportages in tekst, of bijgevoegde bestanden in verschillende formaten met (semi-gestructureerde) tekst.
- Outputs: Moves verwerkt naar de database, stock rapporten vergeleken met de database en onregelmatigheden gerapporteerd en opgestuurd naar equipment control

Daar tussenin moet een oplossing komen die de data uit de rapporten correct uitleest, en de ontbrekende data weet te achterhalen.

Het correct uitlezen van de data is waar de grootste moeilijkheden liggen. Er is daarom eerst gekeken naar een groot aantal voorbeeldrapporten, met als doel om te bepalen of er een enkele oplossing mogelijk is die elke type rapport zou kunnen uitlezen.

4.2.1 Analyse huidige databronnen

Voor deze analyse is er aan de equipment control afdeling gevraagd zo veel mogelijk verschillende rapporten op te sturen. Tijdens het analyseren worden verschillende observaties gemaakt over de bestanden zelf, en de inhoud van deze bestanden. Deze observaties zijn belangrijk voor het ontwerpen van een oplossing.

Observaties over de bestanden:

- Het merendeel van de rapporten zijn in een excel formaat, zoals .xlsx. Groot voordeel is dat de robot makkelijk met een enkel commando de gehele tekst kan inladen. Hierbij wordt eveneens de structuur (de verschillende rows en columns) bewaard gebleven door er een delimiter tussen te zetten.
- Er zijn ook een groot aantal PDF-rapporten. Bij PDF wordt er onderscheid gemaakt tussen twee soorten. De eerste soort computer-gegenereerde documenten. Dit type PDF kan gemakkelijk uitgelezen worden door het te openen in Adobe Acrobat en de tekst te kopiëren naar het klembord.
Het tweede type zijn gescande documenten. Het probleem met dit type is dat deze tekst lastig uit te lezen is. De robot moet in dit geval Optical Character Recognition(OCR) gebruiken, echter is er een grotere kans dat er fouten ontstaan bij het gebruik van OCR. Na een aantal tests bleek dat belangrijke informatie hierdoor soms verloren gaat. Van dit type is er echter maar één rapport. Dit rapport wordt voorlopig handmatig verwerkt en wordt buiten de scope van dit project gehouden.
- Verder zijn er een aantal Word bestandsformaten zoals .docx en .rtf. Deze types bestanden kunnen makkelijk uitgelezen worden door ze te openen in Word en de tekst te kopiëren naar het klembord, om ze vervolgens in een variabele te zetten.
- Er zijn ook een aantal rapporten die online te vinden zijn. Deze kunnen opgehaald worden door middel van UI automatisering. Er moet dan eerst ingelogd worden op de website van de desbetreffende locatie. Vervolgens wordt de data van de website

gehaald gekopieerd. Omdat per locatie een aparte workflow gemaakt moet worden, worden deze locaties voorlopig nog even buiten beschouwing gehouden.

- Tot slot komt er van een depot een email met het rapport in de body van de email in plaats van een attachment. Dit zou gemakkelijk uit te lezen moeten zijn.

De gegevens die nodig zijn om een move naar de database te schrijven zijn:

- ContainerNummer
- Datum van de move
- Tijdstip van de move
- Richting (Binnenkomend/uitgaand)
- Is de container geladen of leeg
- Is de container op/van een truck geladen, of op/van een schip/binnenvaart schip/trein
- De locatie waar de move heeft plaatsgevonden
- De status van de container
- Het type container

Er wordt gekeken of deze gegevens af te leiden zijn uit het rapport. Ook wordt er gekeken op welke manier deze gegevens af te leiden zijn uit een rapport. Dit kan bijvoorbeeld:

- in een tabel staan naast het containernummer van de container waar de move over gaat
- Buiten een tabel staan. In dit geval gaat deze informatie over moves in een (bepaalde) tabel in het rapport.
- Ontbreken en niet af te leiden uit het rapport.

Er zijn de volgende observaties gemaakt over de voorbeeldrapporten:

- Het containernummer staat altijd in een tabel
- De datum staat vaak ergens aan het begin van het document vernoemd. Soms staat het boven een tabel. In dit geval gaat de datum over de moves in de tabel waar het boven staat.
- Het tijdstip staat altijd binnen een tabel, maar ontbreekt bij de meeste rapporten
- De richting staat soms in een tabel, en soms boven de tabel waar dit over gaat. Soms staat de richting niet vernoemd, maar heeft de tabel de kolommen: "datum binnenkomst" en "datum vertrek". In dat geval betekent 1 datum dat het een binnenkomende container is. Als er 2 datums staan betekent het dat het een vertrekkende container is.
- Of de container vol of leeg is, staat vaak niet beschreven als de container leeg is. Als het vol is staat het beschreven bovenaan de tabel of het rapport, of binnen de tabel.
- Of de container een truck op/af komt, of een schip/binnevaartschip/trein, staat vaak niet beschreven in het rapport. Als dit wel beschreven is, is het vaak in de vorm van: een kenteken(voor trucks), een scheepsnaam (voor schepen), of een code (voor treinen).
- De locatie staat vaak beschreven als de naam van het bedrijf waar het rapport vandaan komt. Het kan ook afgeleid worden van de afzender van de email waar het rapport vandaan komt.
- De status wordt vaak niet duidelijk beschreven. Als alles in orde is, wordt het soms beschreven met termen als "ok". Als er schade is wordt dit wel altijd beschreven binnen een tabel over de betreffende container.

- Het type container staat soms beschreven in de tabel. Aangezien dit een constante waarde is die afgeleid kan worden van het containernummer, kan dit gegeven gemakkelijk opgezocht worden in de database.
- De gegevens en tabel kopjes gebruiken geen gestandaardiseerde taal. Zo worden er verschillende woorden of afkortingen, in verschillende talen, gebruikt om eenzelfde gegeven aan te duiden. Hierdoor is er een bepaald niveau van intelligentie nodig om dit te kunnen interpreteren.

Deze observaties zijn meegenomen in het bedenken en kiezen van oplossingsmethodes. De gekozen en onderbouwde oplossingsmethoden zijn in hoofdstuk 3 (oplossingsmethode) beschreven.

4.2.2 Voorlopig ontwerp RPA/Python oplossing

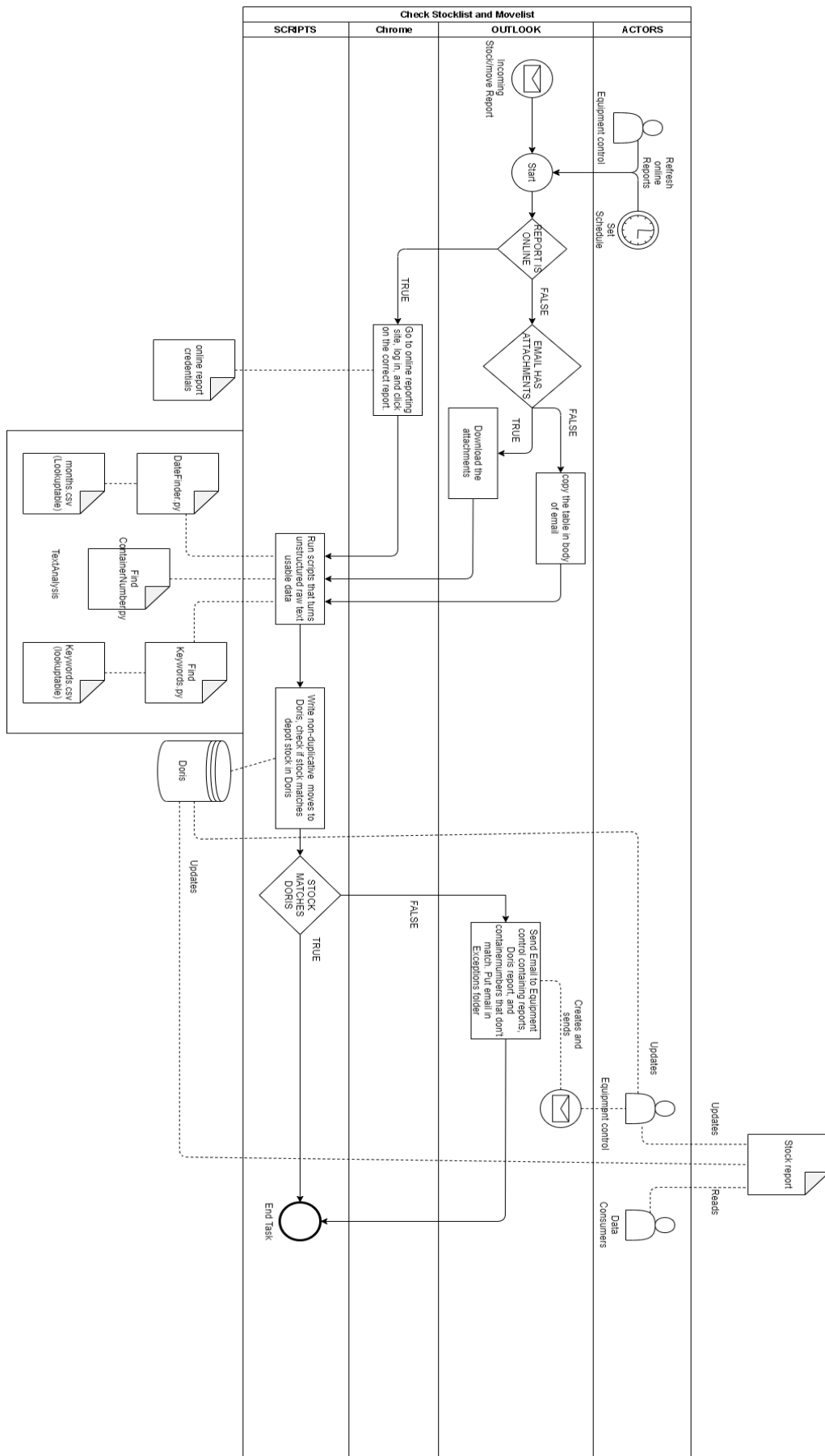
Als beschreven in subparagraaf 3.1.4 er besloten om een enkele oplossing te maken voor het uitlezen van alle rapporten. Dit kan worden gerealiseerd door gebruik te maken van tekstanalyse scripts in Python. Deze zullen aangeroepen worden door de robot om bepaalde kernwoorden uit een stuk tekst te halen.

Er zal door middel van look-up tabellen worden omgegaan met de verschillende talen en termen die in de rapporten worden gebruikt. Als er een nieuwe term voorkomt die nog niet herkend wordt, is het de bedoeling dat de afdeling equipment control deze nieuwe term en de betekenis ervan kan invoeren in de tabel via Excel.

Indien er informatie ontbreekt, wordt er een zo goed mogelijke indicatie gemaakt, gebaseerd op de geschiedenis van de container en de moves die verwacht worden op die dag. Deze gegevens kunnen achterhaald worden door de database te gebruiken.

Tot slot worden er verschillende checks uitgevoerd om de datakwaliteit te waarborgen. Ook hierbij zal er een beroep op de database gedaan worden.

Hieronder (afbeelding 6) staat een runtime diagram die gebruikt is voor de ontwikkeling van het robotscript. Deze is ook te vinden in het SDD (bijlage 3).



Afbeelding 6: Runtime Diagram voor de ontwikkeling van het robotscript

Het geautomatiseerde proces kan grofweg worden beschreven in de volgende stappen:

1. Het proces wordt gestart door een binnenkomende email
2. De robot downloadt de attachment van de email.
3. Met behulp van scripts worden de relevante gegevens uit het rapport gehaald.
4. De moves worden naar de database weggeschreven, en de stock wordt bewaard in een lijst.
5. Hierna wordt de stock, zoals deze in de database staat, gecontroleerd met de stock die is bewaard. Als er een verschil is wordt dit naar de equipment controller gerapporteerd via een dagelijkse email.

Meer details over het ontwerp zijn te vinden in het SDD(bijlage 3).

4.3 Minimum Viable Product

Het Minimum Viable Product (MVP) is een robot die een enkel rapport correct kan verwerken volgens bovenstaand diagram. Dit zorgt voor een goede basis, waar vanuit het opgeschaald kan worden om bijna alle rapporten uit te lezen. Dit is een behoorlijke klus, want om foute data invoer te voorkomen, moet er veel worden getest en getweakt.

Het MVP bestaat uit verschillende onderdelen waaronder:

- De data invoer
- De tekstanalyse
- Verwerking van de data
- Rapportering
- Testen

Het doel van dit verslag is een uitleg te geven van de uiteindelijke oplossing voor deze onderdelen en de verschillende problemen en bijbehorende oplossingen toe te lichten. Aan elk bovengenoemd onderdeel zal daarom een paragraaf worden gewijd. Alhoewel de paragrafen in chronologische volgorde van uitvoer binnen het script zijn beschreven, zijn ze niet in deze volgorde ontwikkeld. Er is eerst begonnen met de tekstanalyse, omdat dit een essentieel onderdeel is, waar geen 100% zekerheid van was dat het zou werken.

4.3.1 Uitlezen van E-mails

Het uitlezen van de e-mails wordt gestart met een zogenaamde Trigger binnen de Krypton Console. Binnen deze trigger kan een IMAP-emailserver worden geconfigureerd met een filter. Als er een email binnenkomt die aan het filter voldoet, kan het automatisch deze mail doorsturen naar de robot die het proces vervolgens start.

Op deze manier kan de robot dus beginnen met werken, zodra een email binnenkomt die aan de eisen voldoet. Eerdergenoemde trigger ziet er als volgt uit (afbeelding 7):

The screenshot shows the Krypton Console configuration for an email trigger. At the top, the 'Name' is 'ET-Equipment Control' and the 'Queue Priority' is 'Normal'. The 'Trigger' section is highlighted, indicating it's the active configuration. Below this, a message states: 'This trigger will initiate a task for every email that matches the following filter.' The 'Email Filter' section contains several fields: 'Subject', 'Body', 'From' (set to 'samskip.com'), 'To', and 'Folder' (set to 'Inbox'). There is also a checkbox for 'Has attachments'. The 'Task' section is visible at the bottom, showing a 'Wizard' dropdown set to 'Container Tracking'.

Afbeelding 7: Een emailtrigger

Hoewel dit onderdeel van het programma vrij eenvoudig leek, dienden er toch enige problemen aan.

Er is geen beschikking over een administrator account, waardoor instellingen binnen de emailserver niet gelijk zelf te veranderen zijn. Dit was echter niet het grootste probleem aangezien Samskip Support snel klaar staat om dit soort dingen aan te passen.

Een groter probleem was het volgende. In eerste instantie kan de eerder genoemde trigger wel een connectie maken met de mailserver en het juiste proces starten. Echter bleek er in de applicatie zelf iets fout te gaan.

Binnen deze applicatie staat een “get email messages” commando. Deze geeft een trigger key mee. Deze trigger key is een string met alle gegevens uit de email. Dit worden gebruikt om verschillende gegevens uit de email te halen. Wat er mis gaat is dat deze key wel ontvangen werd, maar de body, attachment en afzender de waarde ‘NULL’ kregen. De overige gegevens waren wel goed uit te lezen. Er moet dus iets tussen de emailserver en het uitlezen van de trigger key zitten, dat ervoor zorgt dat deze waardes niet uitgelezen kunnen worden.

Wat er precies misgaat is zeer lastig te achterhalen omdat ik niet in de code van kryon zelf kan kijken. Ook zijn de debug tools binnen Kryon Studio vrij gelimiteerd. Ik heb geprobeerd om in de logs van de console te zoeken naar errors, maar ik kon geen relevante errors vinden. Ook ben ik binnen de kryon community op zoek gegaan naar antwoorden, maar heb ik deze niet kunnen vinden. Ik heb ook de emailtrigger getest met een nieuw emailadres die ik daarvoor heb aangemaakt. Dit werkte prima zoals verwacht. Uiteindelijk heb ik een support ticket gemaakt bij Kryon Support in de hoop dat zij mij meer kunnen vertellen over dit verschijnsel.

Na meerdere sessies waarin ik samen met zowel Kryon support als Samskip support op zoek ben gegaan naar een oplossing zijn we helaas nog steeds niet erachter gekomen waardoor de emailtrigger op dit emailadres niet werkt.

Uiteindelijk hebben we een test gedaan met een ander emailadres die de robot kan gebruiken. Op dit emailadres werkte alles zoals verwacht. We hebben het probleem opgelost door een ander emailadres te gebruiken als workaround.

Email regels

Behalve de trigger moesten er ook een aantal e-mail regels geconfigureerd worden. De e-mails die binnenkomen op het account van Equipment Control worden gefilterd. Vervolgens worden deze e-mails met rapportages doorgestuurd naar de e-mail van de robot, waarna ze in de TODO-folder terechtkomen. Als dit proces mislukt, worden ze door de robot verslept naar de FAILED map. Als het slaagt, worden ze door de robot naar de SUCCEEDED-map verslept.

Op deze manier is er een overzicht van alle emails die niet geslaagd zijn. Als dit gebeurd zal er een error verstuurd worden naar de mail van team digital. Hierdoor wordt het team op de hoogte gebracht, en kan iemand de email vinden en de opdracht opnieuw laten uitvoeren.

Een formaat

In de analyse van de databronnen (4.2.1) worden de verschillende manieren om de data uit de email attachments te lezen benoemd. Deze attachments worden volgens die manier uitgelezen en in het klembord geplaatst. Vervolgens wordt de inhoud van het klembord naar een .txt bestand geschreven. Dit wordt op deze manier gedaan, omdat dit het beste de

inhoud en de structuur van de tekst bewaard. Ook is er op deze manier makkelijk terug te vinden wat de robot precies gekopieerd heeft. Ten slotte is de uitkomst van de vorige stappen nu altijd hetzelfde; een tekstbestand met de inhoud van de attachments als plain text.

De robot is nu klaar om door de tekst uit de eerste attachment heen te gaan.

4.3.2 Python tekstanalyse

De tekstanalyse is het onderdeel waar het meeste fout kan gaan. Er komt willekeurige input binnen, en dit moet op een uniforme manier verwerkt worden tot de juiste resultaten. Door goed naar de verschillende rapporten te kijken en naar verbanden te zoeken, zijn er een aantal regels vastgesteld die helpen met het geven van context aan de data. Ook wordt er in deze paragraaf dieper ingegaan op de verschillende python scripts die gebruikt worden en de verschillende variabelen die gevonden moeten worden (en de problemen die daaruit voorkomen).

Algemene ontwerpbeslissingen

Op basis van de analyse van de databronnen (subparagraaf 4.2.1). Zijn er een aantal ontwerpbeslissingen genomen voor de ontwikkeling van het tekstanalyse algoritme. Het doel is om ieder rapport uit te kunnen lezen. Dit wordt bereikt door regels voor het uitlezen te bedenken die voor ieder rapport zou gelden.

Een aantal ontwerpbeslissingen en regels van de tekstanalyse:

1. De tekst wordt regel voor regel uitgelezen. Dit wordt gedaan, omdat in een tabel alle informatie die bij elkaar hoort op dezelfde regel staat.
2. Om te kijken of een regel in een tabel staat, wordt er gezocht naar een containernummer. Dit is het gegeven dat altijd in een tabel rij staat en geldt als de Primary Key in de database.
3. Er wordt via verschillende scripts gezocht naar bepaalde woorden in de regels. Dit zijn containernummers, datums, tijdstippen, richting (binnenkomend/uitgaand), gewicht(full/empty), status (ok, beschadigd, te koop), of het om stock of moves gaat. Dit is de data die nodig is om de move goed in de database te verwerken.
4. Bij het ontbreken van een tijdstip of datum wordt de datum waarop het rapport is verzonden gebruikt.
5. Of een container via de zee of het land binnenkomt of vertrekt, wordt niet bepaald via tekstanalyse, maar via de database. Deze data staat namelijk bijna nooit in de rapporten.
6. Als er geen containernummer in een regel staat (en de regel dus niet binnen een tabel staat), maar wel één van de andere variabelen wordt gevonden, dan geldt deze variabele voor iedere volgende rij in de tabel, totdat een tegensprekende variabele buiten de tabel dit overschrijft. Dit geldt niet als er een tegensprekende variabele binnen de tuple staat.

Hieronder staat een voorbeeld van hoe het algoritme de richting(direction) bepaald. Waarbij de output van het algoritme schuingedrukt in het rood staat:

Incoming				
Containernummer	Datum	Gewicht	Direction	
SANU4021021	5/5/2020	empty	-	<i>direction=incoming</i>
SANU4912021	5/5/2020	empty	-	<i>direction=incoming</i>

SANU4921942	5/5/2020	empty	-	<i>direction= incoming</i>
Outgoing				
Containernummer	Datum	Gewicht		
SANU9412904	5/5/2020	empty	-	<i>direction=outgoing</i>
SANU2941041	5/5/2020	empty	incoming	<i>direction=incoming</i>
SANU9214024	5/5/2020	empty	-	<i>direction=outgoing</i>

De reden waarom voor deze methode is gekozen is omdat informatie over een container soms boven de tabel staat maar nooit eronder. Deze informatie komt soms ook binnen de tabelregel voor. Soms worden uitzonderingen op data ook op deze manier binnen de tabel beschreven. Uit analyse van de verschillende rapporten blijkt dat deze regel geldt voor elk rapport.

4.3.2.1 Datum en tijd vinden

De datum en tijd vinden bleek enigszins problematisch tijdens de uitvoering. Vooral bij het vinden van de datum kan veel misgaan. Bij de datum 20-4-19 kan bijvoorbeeld niet met zekerheid worden gezegd of het jaar 2020 of 2019 is. Voor 3/4/2020 is het moeilijk om te weten of het om 3 april of 4 maart gaat. Zelfs binnen een enkel rapport worden soms verschillende formaten gebruikt. Er is daarom een python script ontwikkeld met Jupyter notebook EDI.

Datum

De datum wordt gevonden door het DateFinder.py script. Dit script heeft als input een string. Als output de laatste datum die gevonden is in de tekst en de hoeveelheid datums. Origineel was het idee om alleen een datum terug te sturen. Later bleek echter dat sommige regels meerdere datums bevatten en dat de laatste datum altijd degene is die het meest relevant is. Dit kan voorkomen bij de volgende tabel bijvoorbeeld:

ContainerNum ber	Date in	Date out	Weight
SANU9241923	1-1-2020	3-1-2020	Empty

In dit specifieke voorbeeld staat niet beschreven of het een ingaande of uitgaande containermove is. Maar voor dit specifieke rapport geldt dat als er twee datums staan, het een uitgaande move is, omdat dat veld anders leeg is. Hier kan met een exceptie, specifiek voor dit soort rapporten, gelijk bepaald worden welke richting een container heeft.

Het script werkt als volgt:

- 1) De data wordt genormaliseerd. Alle '-', ':', '.', en ',' tekens worden vervangen met een '/' teken
- 2) Er wordt met verschillende regular expressions gezocht naar strings die een datum kunnen zijn. Bijvoorbeeld als het er zo uit ziet: `[d]{2,4}/.{1,10}/[d]{2,4}`
Hierbij is `\d` een getal waarvan er 2 of 4 kunnen zijn,
Vervolgens moet er een / staan
Vervolgens mogen er een aantal karakters staan (maanden kunnen voluit geschreven zijn)
Vervolgens moet er een / staan
Vervolgens moeten er weer 2 of 4 digits staan.

- 3) Neem de laatste substring die hieraan voldoet en bewaar de hoeveelheid strings die hieraan voldoen
- 4) Split deze substring in 3 onderdelen met de delimiter /
- 5) Als het deel dat de maand is, volledig staat uitgeschreven (dus: "December" en niet "12"), dan wordt deze met behulp van de PANDAS library in een lookuptable opgezocht om het juiste maandgetal te krijgen. Hieronder staat een voorbeeld van deze tabel (afbeelding 8)
- 6) Er wordt bepaald op welke positie de dag maand en jaar staat door eerst logisch te kijken. (Alleen jaar kan 4 karakters hebben, dag is tussen de 1 en 31, maand tussen 1 en 12)
 - a) Als er niet met zekerheid bepaald kan worden wat de maand of dag is, wordt de meest relevante datum genomen die deze onderdelen kunnen vormen. Er wordt namelijk bijna altijd met hele recente data gewerkt (van gisteren bijvoorbeeld). Daarom wordt zo bijna altijd de goede volgorde gepakt.
- 7) Herbouw de datum naar een standaardformaat (dd/MM/yyyy). En stuur terug naar de robot.

monthnumber								
1	january	jan	studz	ian	led	jan	jan	jaan
2	february	feb	liut	fevr	ún	febr	feb	veebr
3	march	mar	sak	mart	brez	marts	mar	märts
4	april	apr	kras	april	dub	april	apr	apr
5	may	may	mai	mai	kvet	maj	may	mai
6	june	jun	cherv	iuni	cerv	juni	june	juuni
7	july	jul	lip	iuli	cerven	juli	july	juuli
8	august	aug	zhniven'	avg	srp	aug	aug	aug
9	september	sep	veras	sept	zárij	sept	sept	sept
10	october	oct	kastr	okt	rij	okt	oct	okt
11	november	nov	list	noem	list	nov	nov	nov
12	december	dec	snezh	dek	pros	dec	dec	dets

Afbeelding 8: Een deel van de lookup table die gebruikt wordt om maanden te vertalen naar het bijbehorende maandgetal.

Tijd

Het tijdstip is een stuk makkelijker te vinden dan de datum. Deze staat bijna altijd als hh:mm beschreven. Hier wordt in een soortgelijk, maar simpeler script naar gezocht. Helaas geven de meeste depots en terminals het tijdstip niet mee. De afdeling Equipment Control gebruikt in dat geval een standaard tijdstip gebaseerd op de statuscode (combinatie van richting, belading, status, en modaliteit).

Zo kunnen er echter alsnog fouten in de database komen, omdat door deze werkwijze soms de moves in een verkeerde volgorde terecht komen te staan. Als een container op locatie A (zonder EDI) binnenkomt, en gelijk doorgaat naar locatie B (met EDI), zou het volgende kunnen gebeuren. Equipment control geeft een standaardtijdstip voor de events bij locatie A, maar dit tijdstip is later dan de daadwerkelijke tijdstippen bij locatie B. Hierdoor lijkt het alsof de container bij locatie A is, en niet bij B.

In de robot wordt er bij het ontbreken van een tijdstip gebruik gemaakt van de voorspelde tijd van een move zoals het in het planningssysteem van Samskip staat. Dit wordt gedaan door gebruik te maken van de volgende query:

```

select Modality, Laden_empty, planned_event_time, event_expected
from gos.v_booking_events
where Container_number = $containerNumber$
and (Planned_Event_date = $gosDate$)
and direction = $altDirection$
order by Planned_event_date asc, planned_event_time asc

```

Deze query haalt de gegevens op van de geplande move in het systeem. Het is een vrij simpele query, doordat de data engineer een view heeft aangemaakt waarin verschillende tabellen uit planningssystemen al zijn samengevoegd.

Er kan hierbij wel het volgende probleem ontstaan. Als er meerdere moves op dezelfde dag gepland zijn voor een container, wordt er voor elke teruggestuurde, geplande move bepaald of deze al voorkomt in de container tracking database. Zo niet, dan is dat de move die we nodig hebben.

Als er geen move gepland is voor die dag, wordt er nog naar een dag eerder of later gezocht (voor vertragingen). Als er dan nog niks staat, wordt er gebruik gemaakt van de standaard tijden die equipment control op dit moment hanteert.

4.3.2.2 Containernummer vinden

Het containernummer is een belangrijk gegeven om te vinden. Dit is namelijk de primary key waar alle andere data aan gekoppeld moet worden. Voor het containernummer geldt de regel dat het altijd bestaat uit 4 letters gevolgd door 7 cijfers. Hier wordt met een hele simpele regular expression naar gezocht in een lijn tekst. Als er een substring voorkomt die hieraan voldoet, wordt er in de equipment lijst in de database gecontroleerd of deze substring daadwerkelijk een geldig containernummer is.

4.3.2.3 Sleutelwoorden vinden

Er zijn een groot aantal sleutelwoorden die gevonden moeten worden in de tekst. Hieronder volgt een overzicht van een deel van de sleutelwoorden (afbeelding 9):

1	header												
2	stock	stock	voorraad	depositi	inspected	available	repaired						
3	incoming	incoming	ankünfte	import	ingang	ingangsb	inbounds	inbounds	in	entrata/in	entrata	inbount	entry
4	outgoing	outgoing	ladeeinhe	usciti	export	ausgang	ausgangsb	outbound	out	on	exit	gate-out	sai
5	sale	vendita	sale	venda	for sale	for-sale	su	verkoop					
6	planned	planned	reserve										
7	to be insp	surveying	ispezional	unchecker	awaiting	inspected	ai	ae	awaiting	pending			
8	ok	ok	buoni	surveying	verfügbar	verfügbar	available	available	av	pronto	free	libre	sound
9	damage	dm	damage	in repair	damaged	dam_no_e	damage:	ur	br	in reparat	dam	dañar	dañada
10	full	b	beladen	f	volle	full	pieno	cheia	cheio	piena	voll	vol	unspecifie
11	empty	l	leer	e	empty	leere	vuoto	esvaziar	vuota	vuoti	leeg	vacio	

Afbeelding 9: overzicht type sleutelwoorden waarnaar wordt gezocht

Deze woorden zijn belangrijk om te bepalen wat de move precies inhoudt.

Omdat de manier van notering per rapport sterk kan verschillen, wordt er gebruik gemaakt van een lookup tabel, net als bij het bepalen van de maand. Hiervoor is een python script gemaakt, die een string als input heeft. De output is een lijst van de gestandaardiseerde woorden, die zijn tegengekomen in de tekst. De reden dat er gekozen is voor een lookup tabel is omdat dit eenvoudig uit te breiden is voor nieuwe rapporten, doordat de equipment control afdeling nieuwe woorden in de tabel kunnen toevoegen via Excel. Op die manier hoeft het script niet opnieuw geprogrammeerd te worden en kunnen veel verschillende talen worden begrepen.

De lijst met sleutelwoorden die voorkomen worden gebruikt om de variabelen te vullen die nodig zijn voor een move of stock. Als er een bepaald stuk informatie niet gevonden kan worden, kan deze gevuld worden met de gevonden setting voor dit document (dus als er bijvoorbeeld 'incoming' boven de tabel staat, maar niet in de tabel).

De robot heeft nu een lijst met variabelen die gaan over een bepaalde container.

4.3.2.4 Statuscode bepalen

De statuscode geeft in drie letters aan wat er met een container is gebeurd. Als de code MEA is, betekent dat de container leeg is binnengekomen via een truck. Als het MFS is, staat de container te koop. MED betekent dat de container beschadigd is. IDF tenslotte betekent Import Discharge Full, oftewel een volle container is gelost van een trein of schip. Door gebruik te maken van een script met if-else logica, die de verschillende variabelen als input gebruikt, kan er nu een statuscode worden gegenereerd. Indien een van de variabelen ontbreekt, wordt er eerst nog gekeken of deze wel in de geplande moves staat (net zoals het tijdstip zoals eerder beschreven). Als de variabele op dat moment nog steeds ontbreekt (dit kan gebeuren bij een erg schaars rapport, waar de move niet gepland is in het planningssysteem), dan wordt er een standaardwaarde gebruikt.

Er wordt geprobeerd om standaardwaardes zoveel mogelijk te voorkomen. Dit kan immers een reden zijn dat er verkeerde data in de database terechtkomen. Zo gebruikt Equipment Control standaard "truck" als modaliteit, als deze niet in een rapportage vermeld staat. Omdat de modaliteit vaak niet vermeld staat, staan er veel foute data in de database. Het handmatig aanvullen van ontbrekende modaliteit wordt nu door de afdeling Equipment Control gedaan.

Door gebruik te maken van voorspelde moves, kan de datakwaliteit dus verhoogd worden ten opzichte van het handmatige proces.

4.3.2.5 depot code bepalen

De depot code, of locatie code, is een gegeven dat alleen bij Samskip bekend is. Vaak staat er een naam of een logo van een depot/terminal in een rapport. Het blijkt lastig om te weten te komen over welke locatie het precies zou moeten gaan.

Om die reden is ook hier weer gebruik gemaakt van een lookup table. In deze tabel staan gevoelige gegevens zoals emailadressen. Daarom worden deze niet in dit verslag vermeld. In plaats daarvan volgt een uitleg aan de hand van een fictief voorbeeld:

Code	Name	Email	Identifier
Abc01	Amsterdam terminal	amsterdam@voorbeeld.nl	Terminal
Abc02	Amsterdam depot	amsterdam@voorbeeld.nl	Depot
Bai01	Bergen	piet@voorbeeld.no	
dek20	voorbeeld terminal	Jantje@voorbeeld.de	

De robot zal eerst kijken naar het emailadres van de afzender van de email. Als dit emailadres voorkomt in de tabel, wordt de code gebruikt als de depotcode. Als het emailadres niet voorkomt, dan zal er een mail gestuurd worden naar equipment control om dit emailadres toe te voegen aan de tabel met de juiste depotcode.

Er is echter een probleem als rapporten van meerdere locaties van hetzelfde emailadres komen. Gelukkig wordt er dan altijd in het onderwerp van de email gespecificeerd van welke locatie het afkomt. In deze gevallen wordt dit onderwerp gezocht in de Identifier kolom om zo

toch de juiste code te vinden. Op deze manier kunnen nieuwe locaties dus ook makkelijk aangesloten worden door ze toe te voegen aan een excel sheet.

De Depot Code wordt onder andere gebruikt om de correcte locatie door te geven aan de database, maar ook om bijvoorbeeld de loggingsbestanden en attachments automatisch te sorteren zodat ze makkelijk terug te vinden zijn.

4.3.3 Verwerking van de resultaten

Nu alle informatie uit het rapport is gehaald, moet deze informatie verwerkt worden in de systemen van Samskip. Het is belangrijk dat de informatie op een manier wordt verwerkt, waardoor de uitkomst hetzelfde of zelfs beter is, dan wanneer de equipment control afdeling dit handmatig doet. In de volgende subparagrafen wordt er dieper ingegaan hoe de verschillende soorten informatie verwerkt worden.

4.3.3.1 Verwerking van stock

In principe zou op ieder moment bekend moeten zijn welke container op welke plek is. Dit zou mogelijk moeten zijn door door de laatste 'destination' (bestemming) in de containertracking tabel, waar moves naar geschreven worden, te raadplegen. Er blijkt echter dat er vaak moves ontbreken in deze tabel (?). Dit kan komen doordat de EDI ontbreekt, of verkeerd is vermeld. Soms worden er geen, of te laat rapporten hierover verstuurd.

Wat is de werkwijze van Equipment Control?

De moves in rapporten worden gebruikt als controle om te checken of de data in het systeem van Samskip nog klopt met de werkelijkheid. De huidige werkwijze om dit te doen gaat als volgt:

1. Er wordt aan het begin van de dag een tabel gegenereerd met de stock per locatie.
2. Per nog niet verwerkt rapport worden de moves verwerkt en de stock aangepast aan de hand van de gegevens van dat rapport.
3. Als alle rapporten verwerkt zijn, wordt er gewacht tot de moves in het systeem van Samskip staan. Vervolgens wordt er een nieuwe tabel gegenereerd van de stock per locatie (nu met de verwerkte moves).
4. De twee tabellen worden met elkaar vergeleken. Als er iets niet overeenkomt, wordt er uitgezocht wat er is gebeurd. Dit heeft meestal een van de volgende oorzaken:

Oorzaak A: Er is al een nieuwere move binnen voor deze container (bijvoorbeeld via EDI wat sneller is)

Oorzaak B: Er is een move uit het rapport niet verwerkt, of er ontbreekt een move uit het rapport.

Bij oorzaak A hoeft niks gedaan worden, omdat de data in het systeem klopt. Bij oorzaak B moet er alsnog een move geschreven worden naar de database.

Hoe doet de robot het?

De robot doet vrijwel hetzelfde als equipment control. Er zijn echter een paar verschillen namelijk:

- 1) De robot voert de controle uit per email. De moves worden verwerkt, de stock wordt opgeslagen, en vervolgens wordt er gecontroleerd of de stock overeenkomt met de laatst bekende locatie in de database. Dit wordt gedaan per inkomende email, omdat een email meerdere rapporten van 1 locatie kan bevatten, zoals een stockrapport en een moverapport.

Hierdoor verandert het proces van iets dat 1x per dag wordt uitgevoerd naar iets dat wordt uitgevoerd zodra er een rapport binnenkomt.

- 2) De vergelijking wordt niet handmatig gedaan via twee tabellen, maar via een Oracle SQL-query op de containertracking tabel:

```
select top 1 DESTINATION, CON_STATUS, STATUS_DATE  
from dorissmcl.sce_con_tracking sct  
where CONTAINER_NUMBER = $containerNumber$  
order by STATUS_DATE desc, STATUS_TIME desc
```

Deze query zoekt naar de meest recente event van een bepaalde container. Als dit niet overeenkomt, betekent het dat er iets is fout gegaan (eerder genoemde oorzaak A of oorzaak B).

Er wordt eerst gecontroleerd of oorzaak A het geval is. Dezelfde Oracle SQL-statement wordt dan gebruikt, maar in dit geval worden de top 5 records genomen. Als de move wel al eerder is verwerkt in de database, is er niks aan de hand. Er is dan inmiddels al een meer recentere move verwerkt in de database.

Als het oorzaak A niet is dan is het oorzaak B: er is nog geen move voor deze container. In dit geval wordt de gesuggereerde move en relevante data in een rapport gezet en naar equipment control gestuurd zodat zij deze exceptie handmatig kunnen oplossen.

Op deze manier zijn de uitkomsten en de controlestappen van het handmatige proces intact gebleven. Ook is er een menselijk element in dit proces gebleven waardoor de fouten die soms best ingewikkeld kunnen zijn, handmatig opgelost kunnen worden.

4.3.3.2 Verwerking van moves

Werkwijze Equipment Control:

Moves worden door de equipment control afdeling in een excel uploadsheet geschreven. Hiervoor wordt een heel specifiek formaat gebruikt. Deze excelsheet wordt geëxporteerd naar een applicatie die de sheet automatisch uitleest. Vervolgens wordt de informatie in de EDI in tabellen in de database gezet. De database voert een aantal controles uit, zoals duplicate checks en of de containernummers geregistreerd staan. Als de moves niet door de controle komen, stuurt de applicatie een email terug naar equipment control, die vervolgens de fouten moeten oplossen. Als het wel goed gaat, komen de moves uiteindelijk in de containertracking tabel terecht. Vervolgens kan equipment control met de nieuwe data werken, zodra zij de rapporten verversen.

Werkwijze robot:

In de RPA-oplossing wordt dit iets anders gedaan. Dat heeft te maken met de controle die uitgevoerd moet worden bij het verwerken van de stock. Doordat de controle van de stock direct na het verwerken van de moves gebeurt, is tijd een probleem. Er zit namelijk een vertraging van 10 minuten tussen het invoeren van de data in de EDI-in tabel en de verwerking van de data in de container tracking tabel. Deze vertraging komt doordat er een stored procedure om de 10 minuten draait die een batch aan containermoves in een keer controleert en verwerkt.

In principe zou er direct een insert gemaakt kunnen worden naar de equipment control tabel. Toch is ervoor gekozen om een iets ingewikkeldere omweg te maken en het via de EDI-in tabel te doen. Hier is voor gekozen, omdat er op deze manier dezelfde controle stappen

voor de data komen als voor normale EDI-berichten. Om het probleem op te lossen van de vertraging tussen de insert en de verwerking naar de container tracking tabel, wordt de stored procedure die elke 10 minuten draait handmatig aangeroepen.

Om een move te verwerken worden de volgende stappen ondernomen:

- 1) Een duplicate check via Oracle SQL wordt uitgevoerd omdat sommige informatie al met EDI kan zijn ontvangen.
- 2) Een Control number word gegenereerd met de volgende statement:
`select 'TRI'||sce.edi_samskip_ctrl.nextval from dual`
Dit wordt gebruikt als primary key voor de batch in de volgende stappen.
- 3) Er wordt voor elke move een insert gemaakt. Hierin staat alle benodigde informatie voor een move.

```
insert into EDI_CONTAINER_TRACKING
(CONTROL_NUM, EDI_SEQUENCE, RESOURCE_CODE, ISO_CODE, MOVEMENT_CODE, STATUS_DATE,
STATUS_TIME, DEPOT_CODE, WEIGHT, TO_DEPOT, FUNCTION_CODE, EDI_TRADING_PARTNER,
TRANSLATED_MOVEMENT_CODE, CHECK_JOB_REF_FOR_EXT_REF
)
values
('TRI5167587', '1', 'SANU7700366', '45G1', 'MEA', TO_DATE('22.07.2020', 'dd/mm/yyyy'), '1401', 'ITBLQ01', 'O',
'STREETPOOLEUR', 'O', 'LEO', 'MOS', 'X')
```

- 4) Er wordt een Header gemaakt

```
insert into EDI_MESSAGE_HEADER
(EDI_MESSAGE_CODE, CONTROL_NUM, EDI_PROFILE_CODE, EDI_TRADING_PARTNER, EDI_DIRECTION,
TOTAL_RECORD_COUNT, CREATED_DATE_TIME, XFER_DATE_TIME, STATUS, APPLICATION
)
values('EDI_MSG', 'TRI5167587', '000', 'LEO', 'I', '1', TO_DATE('24.06.2020', 'dd/mm/yyyy'), TO_DATE('24.06.2020',
'dd/mm/yyyy'), 'O', 'TRACKING')
```

- 5) Vervolgens wordt er in een loop de stored procedure aangeroepen om deze moves te verwerken.

```
declare
--
cursor container_events is --cursor is counter
select EDI_SEQUENCE, CONTROL_NUM
from EDI_CONTAINER_TRACKING
where 1=1
and CONTROL_NUM = '$CONTROL_NUM$'
;

begin
for event in container_events
loop
begin
DP_EDI_TRADING_UPLOAD.PROCESS_RECORDS(event.CONTROL_NUM, event.EDI_SEQUENCE, 'F');
COMMIT;
EXCEPTION
WHEN OTHERS THEN
NULL;
end;
END LOOP;
```

Op deze manier zijn alle moves direct verwerkt en kan er begonnen worden aan de stock controle. De grootste uitdaging bij deze stappen was het begrijpen van de EDI-applicatie en tabellen. Hierbij is om hulp gevraagd

4.3.4 Rapportering

Excepties die plaatsvinden tijdens de move/stock check worden automatisch in een Excel bestand gezet. Hierin wordt de relevante data uit een aangeleverde rapport gedocumenteerd, de bestandslocatie van het rapport en de reden waardoor de controle niet is gelukt. Eveneens worden de laatste 5 moves van de betreffende container uit de database gehaald en in het rapport gezet. Equipment control kan vaak op basis van deze data al zien wat er precies gebeurd is.

Dit Excel bestand wordt per binnenkomende email gestuurd (als er excepties zijn). Op dit moment zijn er nog niet heel veel rapporten die geautomatiseerd worden. Als dit zou worden opgeschaald, kan het zijn dat deze e-mails als storend worden ervaren. Er zou weliswaar gekozen kunnen worden voor één keer per dag een email te versturen, echter met als gevolg dat er een grotere vertraging van de juiste data kan opleveren. Die vertraging is nu juist wat voorkomen moet worden.

Als er tijdens het proces een foutmelding plaatsvindt, of de juiste data niet gevonden kan worden wordt er nog een email gestuurd, dit maal naar het team digital gestuurd. (wat voor een afdeling is dat?) Vervolgens kan er gekeken worden wat er fout is gegaan, waarna er een patch gemaakt kan worden. Ook kan het rapport worden opgestuurd naar equipment control, zodat ze hem deze keer handmatig kunnen verwerken, mocht de robot het niet op tijd zelf kunnen doen.

Op deze manier kan equipment control zelf de excepties oplossen, en kan ik vanuit mijn kant direct zien als er nieuwe problemen voorkomen.

4.3.5 Testen

Testen is een belangrijk onderdeel van software development. Door te testen kan je vaststellen in hoeverre de software aan de eisen voldoet. In mijn ervaring wordt er bij RPA minder getest dan bij andere soorten software. Hier zijn een aantal redenen voor.

Testen duurt lang, en bij RPA heb je vaak kleine projectjes van 1 of 2 weken. Hiervoor wordt er vaak geen uitgebreid testplan opgesteld en worden er geen uitgebreide testsystemen gebruikt, en geen unittests gemaakt.

Zelfs al zou er van unittests gebruik gemaakt worden, wordt er vaak bij RPA over User Interface Automation gesproken. Er is zo een afhankelijkheid van externe applicaties, alsmede een afhankelijkheid van de input data. Als een paar automatiseringstappen al getest zijn en opeens niet meer werken, ligt het vaak of aan veranderingen in de externe applicaties, of aan foutieve input data. Er worden in stappen waar je userinterface automatiseert bijna nooit aanpassingen gemaakt, want als het werkt, werkt het. Hierdoor is het overbodig om er aparte unit tests voor te maken

Kryon heeft overigens geen test ondersteuning. Desondanks is alsnog een eigen unit test in elkaar gezet door afstudeerder.

Bij RPA is het namelijk belangrijk om acceptance tests uit te voeren. Hiermee kan door de SME en PO gecontroleerd worden of datgene wat gemaakt is ook daadwerkelijk aan de eisen voldoet. Ook kan er gekeken worden, of de equipment control afdeling hierdoor niet onverwachts extra werk krijgt (bijvoorbeeld als de robot fouten maakt en deze moeten worden rechtgezet). Het equipment control proces is geen standaard RPA-project, aangezien er heel weinig UI Automation plaatsvindt, en relatief veel gebruik wordt gemaakt

van code. Ook is dit project veel groter dan een standaard RPA-proces. Ik heb daarom besloten om toch een aantal unit-tests te ontwikkelen voor de verschillende tekstanalyse scripts. Ook ligt er een grote nadruk op goede exceptie behandeling. Verder is er een user acceptance test uitgevoerd.

Het doel hiervan is om ten eerste te kijken of dit project een goede vervanging is van het huidige equipment control proces, ten tweede of de gebruikers hier goed mee kunnen werken, en tenslotte of het de opgestelde doelstellingen heeft weten te behalen. De doelstellingen waren: a. een hogere beschikbaarheid van de data, b. een hogere kwaliteit van de data, en c. een tijdsbesparing voor de equipment control afdeling.

In het testplan (bijlage 4) wordt uitgebreid beschreven hoe bovengenoemde tests zijn opgesteld. In de subhoofdstukken hieronder staan de samenvattingen. Ook gaan deze hoofdstukken dieper in op de problemen en oplossingen die voorkomen tijdens het testen.

4.3.5.1 Unit tests

Zoals hierboven genoemd is, is het lastige aan unit tests, dat er geen unit test frameworks bestaan voor RPA van Kryon. Daarom is er gekozen om zelf iets te ontwikkelen met behulp van Excel. In Excel worden testcases opgeschreven met behulp van de input data. Er wordt voor deze inputdata de verwachte uitkomst opgeschreven. Vervolgens leest de robot de input data uit Excel, voert het in het te testen script, en vergelijkt de uitkomst met de verwachte uitkomst. Vervolgens schrijft de robot in Excel de uitkomst en of de test geslaagd is. In de afbeelding hieronder (afbeelding 10) staat een voorbeeld van hoe de Excel testresultaten eruitzien.

ID	Direction	Weight	Modality	Status	Result	Exptected	Passed?
1	Incoming	empty	sea	ok	MDE	MDE	TRUE
2	Incoming	empty	sea	damaged	MED	MED	TRUE
3	Incoming	empty	sea	sale	MFS	MFS	TRUE
4	Incoming	empty	street	ok	MEA	MEA	TRUE
5	Incoming	empty	street	damaged	MED	MED	TRUE
6	Incoming	empty	street	sale	MFS	MFS	TRUE
7	Incoming	full	sea	ok	IDF	IDF	TRUE
8	Incoming	full	sea	damaged	MED	MED	TRUE
9	Incoming	full	sea	sale	MFS	MFS	TRUE
10	Incoming	full	street	ok	XRX	XRX	TRUE
11	Incoming	full	street	damaged	MED	MED	TRUE
12	Incoming	full	street	sale	MFS	MFS	TRUE
13	outgoing	empty	sea	ok	MOE	MOE	TRUE
14	outgoing	empty	sea	damaged	MED	MED	TRUE
15	outgoing	empty	sea	sale	MFS	MFS	TRUE
16	outgoing	empty	street	ok	MOS	MOS	TRUE
17	outgoing	empty	street	damaged	MED	MED	TRUE
18	outgoing	empty	street	sale	MFS	MFS	TRUE
19	outgoing	full	sea	ok	XOF	XOF	TRUE
20	outgoing	full	sea	damaged	MED	MED	TRUE

Afbeelding 10) Het Excelbestand van de StatusCodeScript unit test.

De bedoeling van deze unittests is om bij veranderingen in deze scripts te controleren dat het nog steeds goed functioneert. Daarom is het belangrijk om na veranderingen in deze scripts, deze test opnieuw uit te voeren (opnieuw te draaien)en te kijken of de test slaagt. Voordat deze wizard in productie zit, is het geen ramp als er foutjes in deze scripts staan, maar als dit in de productieomgeving gebeurt, is het misschien niet in een keer zichtbaar en kunnen er grote problemen ontstaan.

4.3.5.2 User acceptance test

Op het moment van schrijven is de User Acceptance Test nog niet volledig uitgevoerd. Het is helaas niet gelukt om tot uitrollen te komen tijdens de afstudeerperiode. Dit komt door de vele opgelopen vertragingen. In de reflectie wordt hier verder op ingegaan.

Het resterende hoofdstuk zal gaan over het testplan en de uitgevoerde onderdelen van de User Acceptance Test.

Een User Acceptance Test (UAT) is meestal de laatste test die wordt uitgevoerd voor de oplevering van een product. Tijdens deze test wordt er afgevinkt, of het product voldoet aan de doelstellingen. Er wordt ook gecheckt hoe bruikbaar dit product is voor de stakeholders en of de cruciale succesfactoren, zoals beschreven in het PVA (*voluit schrijven?*), worden behaald.

Doelstelling

De doelstelling zoals beschreven in het PVA is:

“Het doel van de opdracht is om een softwarerobot te ontwikkelen die het data-entry proces in zijn geheel of gedeeltelijk kan overnemen. Aan het einde van de opdracht zou deze softwarerobot 24/7 beschikbaar moeten zijn om emails van verschillende depots te lezen en foutloos te verwerken.”

Op het moment van schrijven is de doelstelling nog niet helemaal behaald. De robot neemt namelijk op het moment van schrijven nog geen gedeelte van het data-entry proces over, omdat het eerst nog volledig uitgerold moet worden.

Wel is er een software robot ontwikkeld die 24/7 beschikbaar is om e-mails foutloos te verwerken.

User acceptance

Dit project is vrij bijzonder op dit gebied, omdat er geen gebruikers zijn van de software. In principe draait alles automatisch. Bij de UAT wordt er daarom vooral onderzocht of de robot het gewenste effect heeft op het algehele proces. Dit wordt getest door middel van interviews met de SME, omdat zij het beste zicht heeft op het gehele equipment control proces. Zo wordt er getest of de ingevoerde data correct is door het te vergelijken met de data die de SME zou invoeren voor een bepaald rapport. Ook wordt er gekeken of de container telling, die de SME uitvoert, correct is.

Voordat een rapport goedgekeurd wordt voor automatisering, zal deze eerst een tijdje spookdraaien. Dat houdt in dat het robotische proces wel draait, maar er geen SQL Insert wordt uitgevoerd om de resultaten te verwerken. De resultaten worden in plaats daarvan opgeslagen en vergeleken met hoe de SME deze resultaten verwerkt. Als de resultaten meerdere malen foutloos verwerkt worden, zal de SQL Insert worden toegestaan en hoeft de SME dit rapport niet meer te verwerken. De rapporten worden op deze manier stuk voor stuk toegevoegd. Dit wordt niet allemaal tegelijk gedaan, omdat rapporten vaak voor een paar aanpassingen of uitzondering zorgen en om het overzichtelijk te maken als er iets fout gaat.

Cruciale succesfactoren

De cruciale succesfactoren van dit project zoals beschreven in het PVA zijn:

1. De data is veel sneller beschikbaar, doordat de robot 24/7 kan werken.

2. Verhoging van de datakwaliteit, doordat software robots minder gevoelig zijn voor fouten.
3. Tijdsbesparing van ongeveer 20 uur per week, doordat het werk wordt overgenomen van een werknemer.

Omdat het uitrollen nog niet volledig is voltooid, kan er nog geen definitieve conclusie worden getrokken over alle doelstellingen. Wel zijn er tijdens het testen al een aantal onderdelen opgevallen.

Beschikbaarheid

Dit is erg moeilijk te meten. Voor de rapporten die geautomatiseerd zijn, zou de data altijd sneller beschikbaar moeten zijn. Het tijdsverschil ligt aan het moment van binnenkomst van het rapport en het moment van verwerken van de SME.

Datakwaliteit

De datakwaliteit is aanzienlijk hoger dan het manuele proces. Dit komt vooral doordat de robot gemakkelijk data kan opzoeken in de database. In het manuele proces wordt niet alle data uit het rapport meegenomen, omdat het soms te veel moeite is (tijd kost?) om dit over te kopiëren. In plaats daarvan worden standaardwaardes ingevuld die fout kunnen zijn, of minder accuraat. Bijvoorbeeld: het invullen van Truck bij modaliteit als het niet zeker is (kan fout zijn), of het invullen van een standaard tijdstip in plaats van het aangegeven tijdstip (minder accuraat).

Ook is er een verhoging van de datakwaliteit, doordat de robot gemakkelijk informatie kan ophalen uit verschillende systemen om een voorspelling te maken van wat de data hoort te zijn. Hierdoor hoeft er geen standaardwaarde ingevuld te worden en is deze data vaker correct en meer accuraat.

Verder is er ook meer data beschikbaar, doordat de robot ook de volle containers verwerkt. Deze worden door tijdgebrek vaak niet verwerkt door de SME. De reden hiervoor is: “Deze containers komen toch wel op hun plek terecht”, aldus de SME. Dit is heel begrijpelijk, omdat het voor de equipment control afdeling niet relevant is. Echter ten behoeve van de database (*vanuit de data kant*) is volledigheid aan data wel van belang, zeker met inachtneming met de toenemende vraag naar track and trace. Voor de robot is het geen extra moeite om volle containers ook te verwerken. Helaas kunnen de volle containers niet gecontroleerd worden tegenover de resultaten van de SME. Deze zullen daarom voorlopig gecontroleerd worden door de ontwikkelaar.

Tijdsbesparing

Voor elk rapport dat geautomatiseerd is, hoeft de SME er één minder te doen. De geschatte gemiddelde tijdsduur voor het verwerken van één rapport door de SME is drie minuten. Op het moment van schrijven is er over 10 rapporten zekerheid dat deze correct worden verwerkt. Deze doelstelling is nog niet behaald, omdat er nu maar een paar uur per week wordt bespaard. Zodra de UAT geslaagd is, kan er met deze oplossing wel heel snel opgeschaald worden. Momenteel is er echter nog te weinig getest om op te schalen.

4.3.6 Het uitrollen

Tijdens de uitrol fase wordt het robotische proces daadwerkelijk in productie gezet. Zoals eerder besproken, gaat dit niet in één keer gebeuren. Eerst wordt er een basisversie uitgerold die maar een paar rapporten kan verwerken.

De prioritering van de rapporten gebeurt op basis van input vanuit de business. Er wordt aan de SME gevraagd welke rapporten het meest lastig zijn om manueel te verwerken. Er wordt aan de IT kant gevraagd van welke locaties de hoeveelheid data of datakwaliteit het slechtst is. Op deze manier hebben de stakeholders input over de prioritering.

Na het uitrollen wordt er heel nauwlettend in de databases opgelet of alles goed gaat. Dit wordt in samenwerking met de equipment control afdeling gedaan. Zo kunnen er op tijd fouten worden ontdekt en kan de schade beperkt blijven. Er wordt eerst voor een enkel rapport insert statements ingevoerd. Er zal een delete statement klaarstaan om alle toegevoegde waardes te kunnen verwijderen. Op deze manier kan de schade teruggedraaid worden. Als dit gebeurt, zal er worden gekeken wat er precies mis is gegaan en zullen er verbeteringen worden gemaakt.

Als dit goed gaat, is in principe de UAT geslaagd. Vervolgens worden er zo steeds meer rapporten getest. Als deze rapporten foutloos verwerkt worden, zullen deze goedgekeurd worden voor automatisering.

Er wordt in een apart Excel bestand bijgehouden a.) welke rapporten goedgekeurd zijn voor automatisering, b.) welke getest worden, en c.) welke nog niet getest zijn. Aan de hand van dit bestand kan er worden bijgehouden welke rapporten nog getest moeten worden.

In deze fase is het belangrijk om nauw contact met de business te hebben om problemen op tijd te detecteren en op te lossen. Dit wordt ook wel de hypercare fase genoemd.

Zodra het proces stabiel draait en er geen rapporten meer worden toegevoegd, zal dit proces op de achtergrond blijven draaien. Door middel van rapportering zal er in de gaten worden gehouden of alles nog werkt. Mocht er alsnog iets misgaan, dan kan dit, afhankelijk van het probleem, op twee manieren opgelost worden.

- 1) Als er een bepaalde waarde niet wordt herkend kan de equipment control deze waarde handmatig toevoegen aan de look-up tabellen.
- 2) Als er een fout zit in het systeem, zal de ontwikkelaar deze fout repareren.

Er zullen zo nu en dan wat aanpassingen gemaakt moeten worden. Bijvoorbeeld als er een nieuwe versie is van software die gebruikt wordt, of er een aanpassing is in een SQL-tabel. Hiervoor is goede rapportering belangrijk, zodat zelfs na het uitrollen en de hypercare fase mogelijke problemen worden geïdentificeerd en opgelost.

5. Evaluatie

Een belangrijk onderdeel van dit project is het evalueren van de producten en de aanpak. Door te evalueren kan er helder in kaart worden gebracht wat de resultaten zijn van het project. Ook biedt het de mogelijkheid om kritisch te kijken naar de aanpak. Hierdoor kunnen we kijken wat er in de toekomst beter kan zodat we hiervan kunnen leren.

5.1 Evaluatie van de producten

In dit gedeelte wordt per deelproduct benoemd wat hun doel is, of dit doel is bereikt, wat er goed ging en wat er beter kan.

5.1.1 Plan van Aanpak

Wat is het doel?

Het doel van het Plan van Aanpak is om duidelijk te maken wat de opdrachtgever kan verwachten. De probleemstelling wordt beschreven, de opdracht beschreven, de aanpak wordt bepaald. Ook wordt er een voorlopige planning gemaakt.

Het Plan van Aanpak is goed, als het een goede basis geeft om van daaruit het project te beginnen. Eveneens geeft het de opdrachtgever een goed beeld van wat er wordt gedaan.

Is dit doel bereikt?

Het plan van aanpak heeft mij goed geholpen met het opzetten van het project. Ook weet de opdrachtgever waar hij aan toe is.

Wat ging er goed?

De communicatie met de opdrachtgever en de stakeholders was duidelijk. Ik kreeg vrij snel een goed beeld van de opdracht, doordat ik veel tijd had om te sparren met de opdrachtgever. Verder kon ik goed overweg met de equipment control afdeling en had ik veel tijd genomen om het te automatiseren proces goed te begrijpen.

Wat kan er beter?

Plannen is nooit mijn sterkste kant geweest. Het is moeilijk om in te schatten hoeveel tijd iets duurt, omdat er zich altijd onverwachte dingen kunnen aandienen. Hierdoor is er een te krappe planning gemaakt. Ook is er te veel geschreven over hoe het robotscript is gemaakt, voordat er überhaupt een Process Definition Document en Solution Design Document waren gemaakt. Hierdoor wijkt het eindproduct iets af van het product beschreven in het Plan van Aanpak.

5.1.2 Process Definition Document

Wat is het doel?

Het doel van het Process Definition Document (PDD) is het in kaart brengen van het huidige bedrijfsproces. Zo kunnen zowel ik als anderen gemakkelijk het bedrijfsproces gaan begrijpen. Dit zorgt ervoor dat er zekerheid is, dat de ontwikkelde automatisering doet wat het moet doen. Namelijk een deel van het bedrijfsproces overnemen.

Is dit doel bereikt?

Dit is zeker bereikt. Het PDD geeft door middel van diagrammen een goed overzicht van het huidige bedrijfsproces. Dit is teruggekoppeld naar de equipment control afdeling voor goedkeuring. Daar waren ze het erover eens dat ik het bedrijfsproces goed heb begrepen.

Wat ging er goed?

Door de goede communicatie met equipment control kon er snel en gemakkelijk een overzicht gemaakt worden van het bedrijfsproces. Door goed te interviewen en notities te maken voor de excepties binnen dit proces, kon er in een keer een goed PDD gemaakt worden. Alhoewel er later nog een aantal verbeteringen kwamen.

Wat kan er beter?

Er had tijdens het maken van dit document al kritischer gekeken kunnen worden naar de excepties binnen dit proces. Zo bleek dit project initieel veel lastiger, omdat equipment control alleen de lege containers verwerkte, maar bij bepaalde depots deden ze ook de volle. Later kwam ik er pas achter dat ze dit alleen maar door tijdsgebrek deden en dat het geen probleem is om de robot ook de volle containers te laten doen. Eigenlijk dacht ik dus dat er een ingewikkeld probleem was, maar dat bleek later helemaal niet zo te zijn. Hier had ik tijdens de interviews sneller achter kunnen komen.

5.1.3 Solution Design Document

Wat is het doel?

Er is een ontwerpdocument gemaakt van het uiteindelijke robotscript. Er staat een flowchart die de flow van het script afbeeldt. Verder staan er technische details van het project. Het doel is om hiermee het project over te kunnen dragen aan andere ontwikkelaars. Ook wordt de flowchart gebruikt om de wizard te ontwikkelen.

Is dit doel bereikt?

Het doel is bereikt omdat het een goed werkend document is, dat helpt bij het ontwikkelen van het robotscript. Als een andere ontwikkelaar iets moet aanpassen aan de robot helpt dit document daarbij.

Wat ging er goed?

Ik kwam vrij snel tot een ontwerp. Dit ontwerp bleek ook goed te zijn aangezien ik niet van de flowchart heb moeten afwijken tijdens de ontwikkeling.

Wat kan er beter?

Er zouden meer details bij kunnen. Ook zou er een uitgebreide beschrijving bij elke stap kunnen staan, met daarbij referenties naar hoe deze er precies uitzien in de code. Dit zou een behoorlijke klus zijn, die ik aan het eind van het project nog wel zou willen toevoegen. Dit zou de code makkelijker overdraagbaar maken.

5.1.4 Robotscript

Wat is het doel?

Het doel is om met dit script het equipment control proces te automatiseren. Dit script wordt uitgevoerd zodra er een email wordt doorgesluisd vanaf het equipment control emailadres naar het robot emailadres. De invoer is de inhoud van de email. De uitvoer zijn query's naar de database met daarin de moves en een rapportage naar de equipment control afdeling indien er uitzonderingen zijn.

Is het doel bereikt?

Deels wel. Er moet nog een UAT plaatsvinden, dus tot die tijd kan er niet worden gesproken over bereikte doelen. Wel heb ik met de tests, die al zijn gedaan, vast kunnen stellen, dat het de rapporten correct kan uitlezen. Het haalt de juiste informatie op een goede manier

eruit en zet het om naar het gewenste formaat, en het maakt goede, werkende, query's naar de database. Bovendien heeft de opdrachtgever het MVP wel al goed beoordeeld.

Wat ging er goed?

Over het algemeen was het grootste risico, dat we niet zeker weten of met een generieke methode al deze rapporten uitgelezen kunnen worden. Door middel van een aantal tekstverwerkingsscripts en een groot stuk logica is het wel gelukt om deze rapporten grotendeels uit te lezen. Dit is getest door de output van de robot te vergelijken met de informatie die de SME uit datzelfde rapport heeft gehaald. Vaak hoeft er maar 1 exceptie in een bepaald onderdeel gemaakt te worden om het script te laten werken op een nieuw formaat rapport. Ik ben dus erg tevreden over de werking van dit onderdeel.

Wat kan er beter?

De vele onverwachte vertragingen zijn vooral ontstaan door langzame communicatie.

Een voorbeeld:

Ik had informatie nodig voor de Insert in de EDI-IN tabel. Daarvoor heb ik in opdracht van chris een email gestuurd naar de collega die hierover gaat. De reden dat ik dit deze insert statement niet zelf heb ontworpen is omdat het een belangrijk onderdeel is waar het beter is om de kennis van een expert te gebruiken. Hier krijg ik dan een reactie op zoals: "Is goed, ik maak de SQL-statement voor je.". Vervolgens stuur ik na een week een reminder met "Kan je een indicatie geven wanneer ik dit kan verwachten?". Hierop krijg ik het antwoord dat ik het eind van de week krijg. Vervolgens krijg ik het verkeerde terug. Na dat teruggekoppeld te hebben, moest ik opnieuw wachten omdat de collega deze informatie aan een derde partij moest vragen, die er vervolgens ook weer een week over doet.

In dit voorbeeld is er voor de aanvraag van wat essentiële informatie 1 maand voorbij gegaan voordat het is opgelost. Dit is natuurlijk veel te lang. In die tijd heb ik wel kunnen doorwerken aan het testen op nieuwe type rapporten, maar ik kon de basis van de applicatie niet afmaken.

Er zijn een aantal manieren hoe ik beter had kunnen handelen in deze situatie:

- In plaats van een Email, had ik gelijk kunnen bellen. Hierdoor kan ik meer informatie overbrengen. Hierdoor ben ik duidelijker en hoef ik niet te wachten op een antwoord.
- Ik had de prioriteit en het belang beter duidelijk kunnen maken, en gelijk om een tijdsindicatie kunnen vragen in plaats van eerst te wachten.
- Ik had beter vooruit kunnen kijken en deze informatie weken van tevoren kunnen aanvragen.

De reden dat ik niet had gebeld is, omdat ik soms het gevoel heb dat ik daar collega's mee lastig val. Dit is natuurlijk niet zo, en volgens mij vinden mijn collega's het helemaal geen probleem om te helpen, maar De drempel om te bellen ligt persoonlijk wat hoger voor mij. Sinds deze vertraging ben ik overigens wel vaker gaan bellen.

De corona crisis heeft in deze ook een rol gespeeld. Het nadeel is dat we niet allemaal op kantoor zijn. We niet kunnen dus niet even langs elkaars bureau lopen om iets te bespreken. Hierdoor ontstaan makkelijker miscommunicaties en verloopt communicatie net wat stroever. Nog meer redenen om te bellen dus.

5.1.5 Testplan

Wat is het doel?

Het doel van testen is om te onderzoeken of het gemaakte product op de correcte manier werkt. Het doel van een testplan is om van tevoren een duidelijk plan te hebben van wat er getest gaat worden, en hoe dit wordt gedaan.

Is het doel bereikt?

Er is een testplan opgesteld. Daarin staan enkele unit tests besproken en op welke manier deze worden gemaakt. Ook staat de User Acceptance Test beschreven.

Wat ging er goed?

De unit tests verliepen goed en waren erg waardevol. Elke keer als er veranderingen moesten worden gemaakt in de tekstanalysescripts, kon er via de unit test gecheckt worden of het werkt. Ook kon er hierdoor gekeken worden, of er nu niet iets anders wat voorheen wel goed ging, kapot is gegaan.

Wat kan er beter?

Alhoewel unit tests vrij lastig zijn met RPA, zou er een hogere test dekkinggraad kunnen zijn. Ook is de User Acceptance test nog niet uitgevoerd. Dit komt door de vele vertragingen waardoor het project nog net niet klaar is voor de test.

Veel tests worden al automatisch via de Kryon console uitgevoerd. Zo is er makkelijk te zien of de connectie met de robot goed is en of de verschillende uitgevoerde processen goed verlopen. Ook is er een handig dashboard beschikbaar waarin problemen snel ontdekt kunnen worden. Hierdoor is naar mijn mening de UAT en de unit tests voldoende. Echter zou ik graag toch meer testvaardigheden hebben willen demonstreren. Bijvoorbeeld door meer automatische unit tests te maken.

5.2 Evaluatie van de gekozen aanpak

In dit onderdeel wordt de gekozen aanpak (hoofdstuk 5.1.4) geëvalueerd. Zoals in 5.1.4 beschreven, is er gebruikgemaakt van SCRUM met een aantal elementen van Waterfall. Er is namelijk eerst vooraf een uitgebreide analyse van het proces uitgevoerd en een ontwerp van het script gemaakt. Dit is gedaan omdat de verwachting is dat de oplossing niet ver van het ontwerp zou afwijken en dit is ook zo gebleken.

Verder was het de bedoeling, dat er nauw samengewerkt wordt met de SME en de opdrachtgever. Zo kon feedback snel worden omgezet naar verbeteringen en er toch nog agile gewerkt worden. Dit is in het begin zeker gelukt.

Toen iedereen tijdens de corona crisis ging thuiswerken, is mijn team overgestapt op dagelijkse stand ups en wekelijkse langere meetings. Dit gaf mij de mogelijkheid om de opdrachtgever elke dag updates te geven over het werk en problemen aan te kaarten.

Tegelijkertijd had ik geen wekelijks moment meer met de SME. Hierdoor waren er ook minder momenten om feedback te vragen. Enkele keren is het product wel in een vergadering aan de orde gekomen waar de SME bij aanwezig was. In die vergaderingen was de SME wel akkoord met het programma en bleek zij niet echt feedback te hebben. Omdat er niks veranderd was aan het bedrijfsproces, hoefde er dus ook niet iets gedaan te worden aan mijn programma.

Dit is eigenlijk best zonde, omdat ik hierdoor vaak vergat om bij haar om feedback te vragen. De paar keer dat we een vergadering hierover keurde ze mijn uitwerkingen goed. Ze had niet echt feedback voor me.

Tijdens het analyseren van nieuwe rapporten heb ik overigens wel vaak om haar hulp gevraagd, omdat ik soms echt geen idee had wat er nou stond in het rapport. En als ik het niet weet, dan weet de robot het al helemaal niet. Hier was ze altijd heel behulpzaam bij, wat ik erg fijn vond. Over het algemeen vind ik dat de communicatie redelijk goed is verlopen, maar het had sowieso beter gekund door gestructureerd een keer per week een update te geven aan de SME.

Het werken via SCRUM ging niet zoals verwacht. Dit kwam grotendeels doordat sommige taken veel langer duurden dan verwacht. Voor een taak zoals “opzetten email trigger” had ik origineel 4 uur gepland. Het heeft uiteindelijk dichterbij de 60 uur gezeten, waarin de meeste tijd ging zitten in het debuggen, telefonisch contact met Kryon Support, online onderzoek doen naar emailservers, IMAP, SSL, en andere mogelijke oorzaken voor de fout. Dit soort tegenslagen (vertragen of verpesten?) al snel de mooie burn-down chart en planning. Het bleek soms verstandiger om een sprint iets langer open te houden in plaats van hem te sluiten.

Uiteindelijk is het Scrumbord meer een Kanbanbord geworden. Dit was alleen maar mogelijk doordat er niet in teamverband werd samengewerkt en het niet beëindigen van sprints geen invloed had op het werk van anderen. Op zich is het spijtig dat het scrum werken niet helemaal is gelukt, maar ik vind zelf dat er alsnog gestructureerd gewerkt is. Het blijkt nou eenmaal dat een Kanban bord op dit moment beter werkt voor dit specifieke project.

De invloed van de corona-crisis

Zoals een aantal keren eerder genoemd, heeft door de corona uitbraak het merendeel van de afstudeerperiode thuis plaatsgevonden. Dit heeft een aantal nadelen en voordelen met zich meegebracht, maar naar mijn mening zijn het vooral nadelen. Omdat ik zowel voor als na de start van het thuiswerken bij Samskip heb gewerkt, kan ik dit ook goed vergelijken.

Het grootst genoemde voordeel van het thuiswerken is de flexibiliteit die het met zich meebrengt. *(Het is makkelijk om even een boodschap te doen, te koken, of te ontspannen in de pauze. Hier heb ik zelf ook gebruik van gemaakt door even tussendoor een half uurtje te sporten in de pauze.)* Deze flexibiliteit brengt alleen ook verantwoordelijkheden mee. Zo is het makkelijker om afgeleid te raken en me met niet werk gerelateerde zaken bezig te houden. *(bijvoorbeeld even een interessant YouTube filmpje te bekijken.)* Dit vereist dus veel zelfdiscipline, die ik niet te allen tijden heb. Daarom vind ik het zelf effectiever en prettiger om op kantoor te werken, waar ik gemakkelijker aan mn werk gerelateerde verantwoordelijken kan houden.

Een ander nadeel is dat het toch wel lastiger is om te communiceren met collega's. De afstand om iemand te benaderen is letterlijk en figuurlijk groter.

Tenslotte is het een stuk saaier en eenzamer om niet onder de mens te zijn. In het begin lijkt het vanuit huis werken wel comfortabel, maar op gegeven moment begon ik mijn collega's toch wel erg te missen. Ik zag het merendeel wel elke dag even kort bij de daily standup, maar helaas hield het daar ook op.

Kortom, over het algemeen kon het werk dus wel gemakkelijk van huis uit gedaan worden, omdat er materieel voldoende middelen waren om alles goed uit te kunnen voeren. Het is echter lastiger werken door de benodigde zelfdiscipline en de communicatie op afstand.

5.3 Gedemonstreerde beroepstaken

In dit subhoofdstuk wordt er beschreven op welke manier de van tevoren opgestelde beroepstaken zijn gedemonstreerd.

5.3.1 Beroepstaak A1

Analyseren probleemdomein & opstellen probleemstelling

Deze beroepstaak is uitgevoerd tijdens de intakegesprekken met de opdrachtgever en de procesanalyse van het equipment control proces. Door de intakegesprekken heb ik een beeld gekregen van het probleem, waar in het bedrijf het probleem zich plaatsvindt, en wie, op welke manier, geraakt wordt door dit probleem. Dit is beschreven in het afstudeerplan, het plan van aanpak, en in de probleemstelling van het afstudeerverslag.

Ik heb in de probleemstelling beschreven hoe het huidige equipment control proces problemen oplevert voor de equipment control afdeling en het team digital solutions. Ook heb ik dit proces, en waar het zich in het bedrijf plaatsvindt, in kaart gebracht.

5.3.2 Beroepstaak C6

Ontwerpen van software

Deze beroepstaak is grotendeels uitgevoerd tijdens het ontwerpen van een oplossing. Dit is terug te vinden in het Plan van Aanpak en het SDD. De best-practices voor het ontwerpen van een robotisch proces zijn gehanteerd door gebruik te maken van een SDD als ontwerpdocument. Er is een onderbouwde keuze gemaakt uit verschillende oplossingsmethoden. In het SDD is er ook een workflow-diagram gemaakt. Deze is gebaseerd op de vooraf uitgevoerde procesanalyse en de gekozen oplossingsmethode. Alhoewel het vanaf het bedrijf een vereiste is dat er gebruik wordt gemaakt van Kryon Robotic Process Automation, is er wel een onderbouwde keuze gemaakt om de tekstanalyse via Python te ontwikkelen.

5.3.3 Beroepstaak D14

Realiseren van Software

Ik heb op basis van het software-ontwerp een werkend prototype ontwikkeld. Dit prototype bestaat uit meerdere modules, zoals een data-ingestion module, een verwerkingsmodule, en een rapporteringsmodule. Er zijn ook verschillende scripts ontwikkeld die aangeroepen worden binnen deze modules. Deze scripts zijn geschreven in Python, en maken onder andere gebruik van de PANDAS en RegEx libraries. Er is ook een tekstverwerkingsalgoritme ontwikkeld en geoptimaliseerd.

5.3.4 Beroepstaak D15

Testen en Evalueren

Deze beroepstaak is uitgevoerd tijdens de gehele afstudeeropdracht. Er is voorafgaand het ontwikkelen een testplan opgesteld waarin wordt beschreven welke tests er worden gemaakt.

Een paar van deze tests zijn unit tests om het tekstverwerkingsalgoritme te testen. Deze unit tests zijn deels geautomatiseerd door middel van een aparte testapplicatie die hiervoor is ontwikkeld.

Er is ook een gebruikersacceptatietest ontworpen. Deze kon helaas nog niet uitgevoerd worden.

5.3.5 Beroepstaak E18

Het beheren van applicaties, data en technische infrastructuur

Tijdens de afstudeerperiode is er gebruik gemaakt van versiebeheer via de Kryon applicatie. Ook worden incidenten die plaatsvinden binnen het ontwikkelde proces gedocumenteerd, en worden er op basis van deze incidenten nieuwe issues gemaakt voor de back log. Er wordt naast de geïmplementeerde rapportering, ook logging gebruikt om problemen te kunnen detecteren en de datakwaliteit binnen de systemen in Samskip te onderhouden.

5.3.6 Beroepstaak Gc

Kritisch en methodisch werken

Ik heb een dagelijkse meeting met de opdrachtgever waarin ik vragen kan stellen en om feedback kan vragen. Het werk controleer ik zelf op kwaliteit. Als ik denk dat de kwaliteit goed is, laat ik de opdrachtgever of de SME de kwaliteit en bruikbaarheid controleren. Wanneer ik een probleem tegenkom, spar ik hier vaak over met de opdrachtgever. Hierdoor kunnen we onze ervaringen delen en komen we vaak tot een betere oplossing dan als ik het alleen zou proberen op te lossen.

5.3.7 Beroepstaak Gf

Leren leren

Tijdens de afstudeerperiode heb ik veel kennis opgedaan over de logistiek. Dit was een veld waar ik nog niet veel vanaf wist, dus alleen al in dit opzicht viel er heel veel te leren. Dit kon ik doen door de on-boarding cursussen vanuit Samskip, maar vooral door dingen te vragen aan collega's. Mijn afstudeeropdracht is vooral gelimiteerd tot de Equipment Control afdeling en de afdeling Digital Solutions. Het RPA werk zelf kan echter in bijna de hele organisatie gebruikt worden. Zo leer ik veel over de business vanuit veel verschillende kanten.

Ik heb veel geleerd over RPA met Kryon Systems. Dit heb ik gedaan door eerst een ontwikkelaars cursus te volgen. Vervolgens leerde ik veel door ervaring met mijn afstudeeropdracht en andere kleine projectjes die er tussendoor kwamen. Ook heb ik deel mogen nemen aan een geavanceerde cursus die begeleid is door een aantal ervaren ontwikkelaars. Hierdoor ben ik in contact geraakt met andere Kryon RPA-ontwikkelaars, waarvan ik veel nieuwe technieken en best practices heb geleerd.

Ik heb deze beroepstaak gedemonstreerd door te blijven leren tijdens de gehele afstudeeropdracht.

6. Literatuur

- 1) *pandas - Python Data Analysis Library*. (2020). PANDAS Library.
<https://pandas.pydata.org/>
- 2) *Check if a string only contains numbers - Regex Tester/Debugger*.
(2020). RegEX. <https://www.regextester.com/21>
- 3) *Kryon Systems - Transform Your Business with AI-based RPA*. (2020, September 14). Kryon. <https://www.kryonsystems.com/>
- 4) *Form Recognizer – Documentextractieservice met AI*. (2020). Microsoft Azure. <https://azure.microsoft.com/nl-nl/services/cognitive-services/form-recognizer/#features>

7. Woorden en Afkortingenlijst

Woord/afkorting	Betekenis
Delimiter	een leesteken dat gebruikt wordt als indicatie dat een bepaald element beëindigd en een nieuwe begint. Zoals een Komma in .CSV (Comma Separated Values).
Depot	Een plek waargrote hoeveelheden containers kunnen worden opgeslagen.
EDI	Electronic Data Interchange. In de context van dit verslag gaat het over het versturen van gegevens over moves vanuit een depot/terminal direct naar de systemen van samskip.
Equipment Control	De afdeling die verantwoordelijk is voor het bijhouden van de status en locaties van apparatuur(zoals containers) van Samskip.
Excel Upload Sheet	Een manier van data invoer voor de database. De data wordt in een excel bestand gezet, en deze wordt gemailt naar een applicatie die de data in de database zet.
IMAP	Internet Message Access Protocol. Dit is een protocol dat gebruikt wordt om met een extern programma een emailserver uit te lezen.
Jira	Platform dat wordt gebruikt voor Scrum en andere planningsactiviteiten.
Kryon	De leverancier van de robotsoftware.
Move	Een eventwaar bij een container een terminal of depot in- of uitgaat. Dit gebeurt altijd via een voertuig (zoals truck, schip, of trein).
MVP	Minimum Viable Product. Het MVP is een product dat in principe in gebruik kan worden genomen. Hierop kunnen extra functionaliteiten of uitbreidingen worden gemaakt.
OCR	Optical Character Recognition. Een methode om tekst uit afbeeldingen of user interface te lezen.

PDD	Process Definition Document. Een gedetailleerde beschrijving van het te automatiseren proces.
PO	Process Owner. De product owner is de persoon die verantwoordelijk is voor het proces. Dit kan bijvoorbeeld een manager zijn, maar het kan ook de SME zijn.
Robot	Een digitale medewerker. Een stuk software dat bepaalde scripts kan uitvoeren op een machine.
RPA	Robotic Process Automation. Het automatiseren van bedrijfsprocessen door gebruik te maken van (digitale) robots.
SDD	Solution Design Document. Een gedetailleerde beschrijving en ontwerpen van de oplossing.
SME	Subject Matter Expert. Dit is de persoon die het meest afweet van het te automatiseren bedrijfsproces. dit is vaak degene die dit proces uitvoert voordat het proces geautomatiseerd is.
Stock	De voorraad containers op een depot of terminal.
Terminal	Een plek waar containers kunnen worden geladen/afgeladen op/van een voertuig.
Trigger	De definitie van de events en condities die nodig zijn om een robotisch proces te starten. Bijvoorbeeld als er een email binnenkomt(event) van de afzender Equipment Control (conditie), start het equipment control proces (robotisch proces).
Wizard	Het script dat de taken gedefinieerd die de robot moet uitvoeren voor een bepaald proces. De wizard wordt ontwikkeld in Kryon Studio.