



Afstudeerscriptie: Agent migratie
Door: Maikel Hoogeveen
Datum: 08-06-2011 Eindhoven
Versie: 1.0

Afstudeerder	: Hoogeveen M. (Maikel)
Datum	: 08-06-2011
Versie	: 1.0
Afstudeerbedrijf	: Logica Nederland BV
Divisie	: Professional Services
Practice	: Infrastructure, Security & Architecture
Practice manager	: Dissel, A. Th. (Arnold)
Bedrijfsbegeleiders	: Olderen, J.C. (Jeroen) van
Bedrijfsmentor	: Driel, T.W. (Tom) van
Information Architect	: Ommen, J.H.E. (Eric) van
Onderwijsinstelling	: Fontys Hogeschool ICT
Opleiding	: IMS
Afstudeercoördinator	: Heijden, M.J.W. (Rene) van der
1e Assessor	: Zijlmans, W.F.J.J. (Wim)
2e Assessor	: Friebe, Y.V. (Yvette)

Getekend voor gezien door	: Olderen, J.C. (Jeroen) van
Datum	:
Handtekening	:

CONTACTGEGEVENS

Examinandus

Naam : Hoogeveen M. (Maikel)
Straat / Nummer : Ijsselstraat 3
Postcode / Plaats : 5704 HM Helmond
Telefoonnummer : +31 6 1151 6852
E-mailadres : maikel.hoogeveen@logica.com
Privé E-mailadres : maikel.hoogeveen@mobileemail.vodafone.nl
Onderwijsinstelling : Fontys Hogeschool ICT
Afstudeerrichting : IMS
Studentnummer : 2110484

Opleiding:

Naam : Heijden, M.J.W. (Rene) van der
E-mailadres : r.vanderheijden@fontys.nl
Telefoonnummer : +31 8 7787 1070
Functie : Afstudeercoördinator

Naam : Zijlmans, W.F.J.J. (Wim)
E-mailadres : w.zijlmans@fontys.nl
Telefoonnummer : +31 8 7787 0910
Functie : 1^e Assessor

Naam : Friebe, Y.V. (Yvette)
E-mailadres : y.friebe@fontys.nl
Telefoonnummer : +31 8 7787 8727
Functie : 2^e Assessor

Bedrijfsgegevens:

Naam : Logica Nederland BV
Afdeling : Working Tomorrow
E-mailadres : workingtomorrow.nl@logica.com
Straat / Nummer : Flight Forum 3250
Postcode / Plaats : 5657 EW Eindhoven

Naam : Olderen, J.C. (Jeroen) van
Telefoonnummer : +31 6 5261 3779
E-mailadres : jeroen.van.olderen@logica.com
Functie : Project leader Logica Working Tomorrow Eindhoven

Naam : Ommen, J.H.E. (Eric) van
Telefoonnummer : +31 6 2036 7875
E-mailadres : eric.van.ommen@logica.com
Functie : Information Architect

Naam : Driel, T.W. (Tom) van
Telefoonnummer : +31 6 4524 3265
E-mailadres : tom.van.driel@logica.com
Functie : IT Consultant, bedrijfsmentor

Naam : Dissel, A.Th. (Arnold)
Telefoonnummer : +31 40 295 7716
E-mailadres : arnold.dissel@logica.com
Functie : Practice manager Infrastructure, Security & Architecture

VOORWOORD

Deze afstudeerscriptie is geschreven in het kader van het project *Agent Migratie* dat door Maikel Hoogeveen is uitgevoerd bij Logica. Deze stage is het laatste onderdeel van het succesvol doorlopen van mijn opleiding aan de Fontys Hogescholen ICT te Eindhoven. Deze afstudeeropdracht is uitgevoerd in het kader van de Minor Information, Management en Security.

De afdeling waar ik onder val als afstudeerder is Working Tomorrow, dit is een programma vanuit Logica dat zich richt op het begeleiden van studenten met het afstuderen en het bieden van een plek waar studenten uitdagende en innovatieve afstudeerprojecten uit kunnen voeren. Tijdens deze afstudeerstage ben ik samen met acht andere studenten bij Working Tomorrow begonnen aan de laatste stap van de opleiding.

Ik heb voor Logica gekozen als stage biedende organisatie vanwege de mogelijkheden die het Working Tomorrow programma te bieden heeft. Hiernaast levert het werken aan een innovatief project veel kennis en ervaring op omdat het nieuwe materie betreft.

De opdracht die ik heb gekozen om uit te voeren heeft mijn interesse gewekt vanwege het open begin, de opdracht kon naar eigen inzicht worden ingevuld om datgene eruit te halen wat er in zat. Voor mij heeft dit het voordeel gehad dat ik mijn bedrijfskundige kant kon combineren met de leergierige kant. Het ontwikkelen van een systeem in JAVA was voor mij dan ook een leuke uitdaging waar ik plezier maar ook leermomenten aan te danken heb.

Ik wil graag iedereen bedanken die heeft bijgedragen bij het afronden van deze opdracht, in het bijzonder:

- Tom van Driel

Tom is als mentor de persoon geweest waar ik met mijn vragen terecht kon, dankzij de ervaring die hij heeft opgedaan tijdens zijn carrière kon hij me advies geven bij het uitwerken van mijn opdracht. Hij heeft me tevens gewezen op de kleine details tijdens het ontwikkelen van het systeem die ik soms over het hoofd heb gezien.

- Jeroen van Olderen

Jeroen heeft mij tijdens het gehele project voorzien van procesbegeleiding en handvaten om aan te grijpen tijdens de uitdagingen en keuzes die gemaakt moesten worden. Hij heeft naar mijn mening de goede verhouding tussen professionaliteit, collegialiteit en humor laten zien die bij het begeleiden van de groep studenten die hij kreeg toegeworpen hoort.

- Eric van Ommen

Eric heeft mij meerdere malen voorzien van een review op de documenten die ik hem heb toegestuurd. Dankzij zijn professionele blik op de documenten, de onderlinge samenhang en de tips met betrekking tot het bericht dat ik over wil brengen aan de lezer ben ik er in geslaagd om de documentatie op orde te krijgen.

- Arnold Dissel

Arnold heeft mij de mogelijkheid gegeven om gebruik te maken van de kostbare tijd van Tom en heeft mij en de andere studenten die aan zijn practice werden toebedeeld opgenomen in het team. De teammeetings waren erg gezellig en gaven een andere blik in de keuken van Logica die je tijdens het project misschien over het hoofd zou zien.

- Yvette Friebe

Yvette heeft als 2^e assessor en begeleidster vanuit Fontys ervoor gezorgd dat de communicatie tussen mij en de opleiding goed is verlopen. Mede dankzij haar feedback is het me gelukt om tijdens het schrijven van mijn scriptie uit de technologie te stappen om het proces van het afstuderen toe te lichten.

- Mede Working Tomorrow studenten

Mijn medestudenten hebben ervoor gezorgd dat er altijd iets te beleven viel op de werkvloer, waren het niet de voorbij rijdende projecten dan waren het wel de momenten dat we tijdens de lunch onze tafelvoetbal kwaliteiten aan het verbeteren waren. Bedankt voor de prettige werksfeer.

- Familie en vrienden

Ik wil graag mijn familie en vrienden bedanken voor het zorgen van een omgeving waar ik tot rust in kon komen na het werk en mijn batterij weer kon opladen voor de volgende dag.

- Rieneke van Santen

Ik wil mijn vriendin graag bedanken voor het geduld dat ze heeft opgebracht tijdens de momenten dat ik mijn tijd in het onderzoek heb gestopt, waardoor ik weinig tijd heb overgelaten om samen te spenderen. Gelukkig kan ik deze tijd ruimschoots inhalen zodra we een eigen huisje hebben!

INHOUDSOPGAVE

Samenvatting	7
Summary	8
Verklarende woordenlijst	9
1 Inleiding	10
2 Het Bedrijf	11
2.1 Logica	11
2.2 Working Tomorrow	11
2.3 Fontys Hogescholen	13
2.4 Begeleiding	14
3 De opdracht	15
4 Aanpak onderzoek	16
4.1 Onderzoek	16
4.2 Systeemontwikkeling	16
5 Onderzoek	17
5.1 Agents	17
5.2 Agent platform	18
5.3 Migratie	20
5.4 Doel	21
5.5 Impact	21
5.6 Security	22
5.6.1 Trusted Agents Add-on	22
5.6.2 Security Add-on	22
5.6.3 SSL IMTP	23
5.7 Controle	24
5.8 Ontwikkelmethodes	25

6	Toepassing theorie	26
6.1	Doel	26
6.2	Impact	28
6.3	Controle	33
7	Systeemontwikkeling	34
7.1	Toelichting concept	34
7.2	Globaal model	35
7.3	Functionaliteiten	36
	7.3.1 De GUI	36
	7.3.2 De Controller	36
	7.3.3 Mobile Agents	36
7.4	Planning	36
7.5	Resultaat	37
	7.5.1 GUI	37
	7.5.2 Systeem	38
8	Conclusie(s) en aanbeveling(en)	40
	Evaluatie	41
	Literatuurlijst	42
	Bijlagen	43

SAMENVATTING

Deze afstudeerscriptie is het resultaat van het onderzoek *Agent Migratie* en is geschreven door Maikel Hoogeveen. Het onderzoek geeft een antwoord op de vraagstelling: "*Hoe voorkom je dat agenten zomaar overal naartoe kunnen verhuizen?*".

Deze vraagstelling komt voort uit de opdracht die voor Logica is uitgevoerd. De uitvoer hiervan heeft plaats gevonden bij het Working Tomorrow afstudeerprogramma van Logica.

Aan de hand van de originele opdracht definitie is er richting gegeven aan het onderzoeken van software agents. Deze software agents kunnen de eigenschap hebben om zich te verplaatsen tussen twee platformen die zich op aparte systemen bevinden, dit wordt migreren genoemd. Deze platformen kunnen worden gezien als leefomgeving voor een agent. Zodra een agent de capaciteit heeft om te migreren tussen de platformen wordt deze een "Mobile Agent" genoemd.

Een mobile agent migreert zich tussen de platformen doormiddel van netwerkcommunicatie, deze communicatie is te beveiligen met het gebruik van certificaten die aanwezig zijn op de systemen. Dankzij deze beveiliging kan een migratie veilig plaatsvinden en is het mogelijk om een veilige communicatie tussen een mobile agent en een andere agent te realiseren.

Een mobile agent kan onder controle worden gehouden door een andere agent en het platform waar de agent zich bevindt. Tijdens dit project is het "idee" ontstaan om een agent te ontwikkelen die deze controle voor zijn rekening neemt.

Het platform dat is gekozen tijdens het project is het Java Agent Development Framework. Dit platform biedt de benodigdheden voor het ontwikkelen van een agent georiënteerde oplossing die op Linux en Microsoft systemen kunnen werken.

De geleerde theorie is toegepast in een Business Case waar een hypothetische situatie waarin de vraag naar een systeem staat beschreven. Dit systeem moet in een omgeving met Linux en Microsoft systemen kunnen werken. Naast het opstellen van deze Business Case is er een voorbeeld Business Impact Analyse uitgevoerd om te onderzoeken wat de risico's van een dergelijk systeem zouden kunnen zijn en hoe dit in een financieel plaatje is weer te geven.

Het systeem is volgens een aangepaste ontwikkelmethode gerealiseerd. Voor deze aanpassing is gekozen door het formaat van het project en het ontbreken van een vitale stap. Het opstellen van de requirements voor het systeem. Het resultaat is een werkend systeem dat gebruik maakt van mobile agents zonder de controle te verliezen over de agents en een mogelijkheid om de status en resultaten van uitgegeven opdrachten in te zien.

Met de toepassing van de theorie en de ontwikkeling van een werkend systeem toont dit onderzoek aan dat het gebruik van mobile agents op een gecontroleerde en veilige manier kan plaatsvinden.

SUMMARY

This thesis is the result of the research done by Maikel Hoogeveen for the project *Agent Migration*. The question that answered by this research is: *"How can you prevent uncontrolled migration of agents?"*

This question is part of the project has taken place at the graduate program Working Tomorrow at Logica. The following definition led the research in the right direction to obtain an answer to the initial question:

"This research takes place during the graduate project at Logica and will investigate the use of software agents to collect information in a cross-platform environment and the impact and security aspects of this usage."

Software agents are pieces of software that can have different properties. Some software agents have the ability to move between different platforms. We call this migration. The platform is the environment where the agents live in. When an agent has the ability to migrate, it is being referred to as a mobile agent.

A mobile agent migrates between platforms by using communication over the network, it is possible to keep the communications secure by using certificates at each system. Using this secure connection makes it possible for a mobile agent to migrate securely between platforms.

It is possible to control a mobile agent by using another agent or platform. The idea to develop an agent that can accomplish this task is formed during this project. The platform Java Agent Development Framework is chosen for this project. This platform offers the necessities for the development of a system that works on both Linux and Microsoft systems.

The knowledge gained is applied to a Business Case, which describes a hypothetical situation where a company asks for a system that can operate in both Linux and Microsoft environments. A Business Case is made and a Business Impact Analyses took place to find out which risks would accompany the usage of such a system and how this can form a financial picture.

The system is developed using a modified development method. This modification had to take place because of the size of the project and the method is missing a crucial step, forming requirements for the system. The result is a working system that uses mobile agents without losing the control and the ability to check the status and results of given tasks.

This research proves that mobile agents can be used secure and controlled by applying the theory on a case and developing a working system.

VERKLARENDE WOORDENLIJST

Woord/Afkorting Betekenis	
Agent	Een stuk software met specifieke eigenschappen. (Zie 5.1)
AMS	Agent Management System
BIA	Business Impact Analyse
C&PS	Consulting en Professional Services
FDD	Feature Driven Development
FIPA	The Foundation of Intelligent Physical Agents
FO	Functioneel Ontwerp
GUI	Graphical User Interface
IMTP	Inter-container Message Transport Protocol
IPM	Inter Platform Mobility
ISA	Information, Security & Architecture
JADE	Java Agent Development framework
JRE	Java Runtime Environment
Platform	Een software matige omgeving waar agents uitgevoerd kunnen worden.
PLC	Public Limited Company
PM	Practice Manager
RMA	Remote Monitoring Agent
SL	Service Lines
SSL	Secure Sockets Layer
TO	Technisch Ontwerp
WT	Working Tomorrow

Tabel 1: Verklarende woordenlijst.

Toelichting literatuurverwijzingen:

De literatuurverwijzingen in dit document zijn als volgt opgezet. Tussen [] staat de eerste letter van de achternamen van de auteurs met erachter de datum van publicatie.

Voorbeeld 1: [FG97] staat voor: Franklin, Graesser, 1997.

1 INLEIDING

Hoe kan je op meerdere plaatsen tegelijk zijn zonder er daadwerkelijk te zijn? Hoe kan je meerdere taken tegelijk uitvoeren zonder iets te doen? Is het mogelijk om het antwoord op deze vragen te weten zonder dit document te lezen? Waarschijnlijk wel, maar de aandacht is er!

Deze afstudeerscriptie heeft betrekking op het onderzoek Agent Migratie dat is uitgevoerd bij het afstudeerprogramma van Logica genaamd Working Tomorrow.

De vraag die in de opdracht centraal staat heeft betrekking op de migratie van agents, maar wat zijn agents eigenlijk en wat zijn agent platformen? Wat houdt een migratie in, hoe veilig is het en hoe behoudt je de controle? Dat zijn de vragen waar we een antwoord op willen. Dit project geeft een antwoord op deze vragen, om ervoor te zorgen dat we een stap dichterbij het antwoord op de vragen die de aandacht trokken zijn.

De informatie die tijdens dit onderzoek naar boven is gekomen is als volgt ingedeeld. Hoofdstuk 2 beschrijft het bedrijf en de omgeving waar het project is uitgevoerd en hoofdstuk 3 legt het ontstaan van de opdracht uit. Voor de aanpak is in hoofdstuk 4 de opzet te vinden. De resultaten van deze aanpak worden in de hierop volgende hoofdstukken weergegeven, hoofdstuk 5 legt de theorie en keuzes die tijdens dit project zijn gemaakt uit. Deze kennis wordt toegepast op een hypothetische situatie in hoofdstuk 6. Hoofdstuk 7 licht het ontwikkelde systeem dat de in hoofdstuk 8 gevormde conclusie ondersteund toe.

2 HET BEDRIJF

Dit hoofdstuk beschrijft de organisatie en de omgeving waar de afstudeeropdracht is uitgevoerd.

2.1 Logica

Logica is een belangrijke internationale speler op het gebied van IT en business services met 39.000 mensen in dienst in 36 landen. Ze bieden oplossingen en diensten aan in het domein van consultancy, design, systeem integratie, bedrijfskritische applicaties en de outsourcing van bedrijfsprocessen. Logica richt zich op vier marktsectoren – Energy Utilities en Telecom, Finance, Public Sector, en Industrie Distribution en Transport. Door deze focus zijn ze in staat om op maat gemaakte oplossingen te bieden voor de uitdagingen waar klanten in hun specifieke markt mee te maken krijgen. In Nederland is deze focus verankerd in verschillende afdelingen die zich richten op een van de bovengenoemde segmenten.

Logica is gedreven op het helpen van klanten om leidende posities te behalen en te behouden in hun individuele markten. De kracht van Logica ligt daarbij op het gebied van industrie- en domeinkennis, het sterke bedrijfskundige en technologische inzicht, en de vele successen met het leveren op tijd en binnen budget.

Het huidige Logica Public Limited Company (plc.) is op 30 december 2002 ontstaan uit het voormalige Logica plc. (60%) en het voormalige CMG plc. (40%). CMG was in Nederland aanzienlijk groter dan Logica, dat vooral een focus had in Groot-Brittannië. In 2006 zijn daar nog WM-Data (zwaartepunt in Scandinavië) en Unilog (Zwaartepunt in Frankrijk) aan toegevoegd, zodat een gespreide dekking in Europa verkregen is. In 2008 is onder leiding van de nieuwe CEO, Andy Green, een programma opgestart om van alle onderdelen binnen het bedrijf een uniforme organisatie neer te zetten - One Logica. Hierbij zijn de waarden gedefinieerd, volgens welke iedere Logicaan zou moeten werken: Committed, Innovative en Open. De hierbij gestelde missie is als volgt, "To be the most trusted innovation partner".

2.2 Working Tomorrow

De opdracht wordt uitgevoerd binnen het programma Working Tomorrow (WT). Logica heeft dit programma opgestart om studenten de gelegenheid te geven af te studeren op een innovatieve opdracht met goede begeleiding. Innovatie staat centraal. Innovatief wat betreft technologie, concept of methodiek. Zo werken er studenten aan agent technologie, Multi touch technologieën, Augmented Reality, Smart Grids, slimme meters en mobile oplossingen. Working Tomorrow is nu op 5 vestigingen van Logica in Nederland aanwezig, en biedt jaarlijks plaats aan circa 120 studenten van HBO en WO. Op iedere vestiging is een projectleider van Working Tomorrow die de studenten begeleidt. Het Working Tomorrow programma heeft 4 hoofddoelen:

- Een centrale plaats bieden waar studenten uitdagende en innovatieve afstudeerprojecten kunnen uitvoeren.
- Het werven van toekomstige werknemers.
- Verhogen van de reputatie van Logica op het gebied van innovatie.
- Demo's en resultaten van projecten gebruiken voor het verkrijgen van betaalde opdrachten en vergaren van kennis.

Logica Nederland heeft naast de Industrie Sectoren (IS) ook Service Lines (SL). Consulting en Professional Services (C&PS) is een SL die er op gericht is om de IS zo goed mogelijk te voorzien van goed en gekwalificeerd personeel. C&PS is verder onder te verdelen in Consulting, SI Projects en Professional Skills. Professional Skills is weer verder onderverdeeld in practices, soort van afdelingen die gericht zijn op een bepaalde skillset, bijv Project Management, Java.Oracle of ISA. Figuur 1 geeft de huidige indeling van de Nederlandse Logica plc.

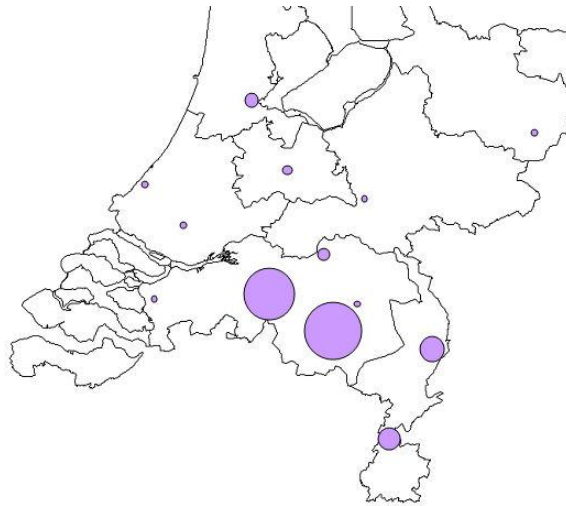


Figuur 1: Organogram Logica.

Alle studenten die bij WT afstuderen zijn daarnaast ook onder gebracht in een practice die nauw aansluit bij de opdracht. WT valt onder **Business Consulting**, **Professional Services** en **Outsourcing Services**.

2.3 Fontys Hogescholen

Fontys Hogescholen is met ongeveer 38.000 studenten, 4.000 personeelsleden en een onderwijsaanbod in bijna alle sectoren, een van de grootste hoger onderwijsinstellingen van Nederland. De 31 instituten verzorgen ruim 200 bacheloropleidingen in de varianten voltijd, deeltijd, duaal en in-service. Daarnaast worden 36 master varianten en zeven Associate degrees aangeboden. Cursussen en trainingen worden in stedelijke centra - vooral in Zuid-Nederland - aangeboden. Naast onderwijs legt Fontys zich ook - met 37 lectoraten - toe op onderzoek en kennisinnovatie en contractactiviteiten. In figuur 2 staat een schematisch overzicht van het aantal inschrijvingen per lesplaats.



Figuur 2: Aantal inschrijvingen van studenten per Fontys locatie.

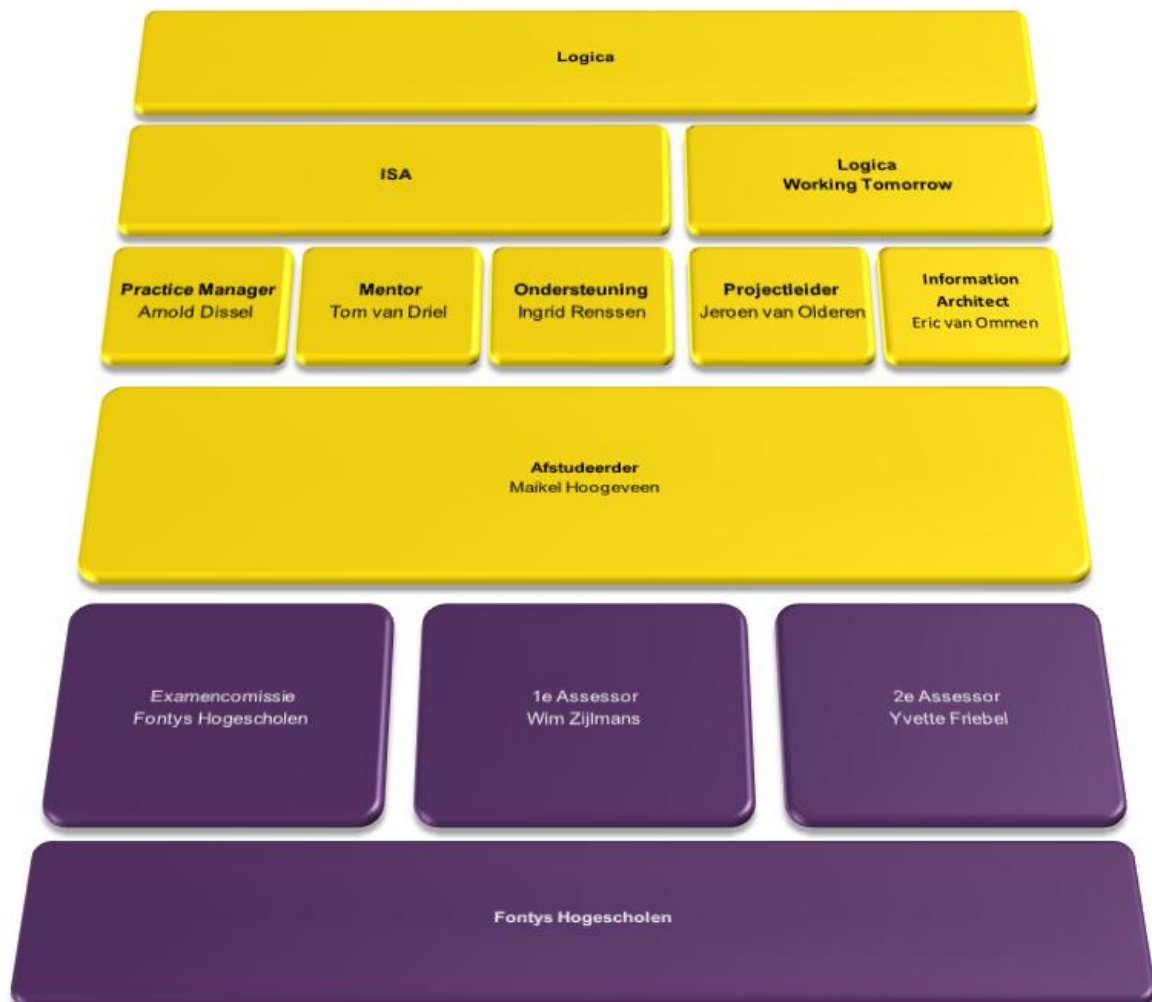
Fontys leidt haar studenten op tot toekomstige beroepsbeoefenaars die over vakinhoudelijke competenties beschikken en zich bewust zijn van de diversiteit aan kennis en cultuur die onze maatschappij vormgeven. Nadenken en praten over duurzaamheid, beroepsethiek en zingeving is daarbij vanzelfsprekend. Bovendien hecht Fontys grote waarde aan de persoonlijke vorming van de individuele student. Aandacht voor het ontdekken van de eigen talenten, dromen en ambities draagt bij aan dit proces. Ook het creëren van gedenkwaardige belevenissen tijdens de studietijd maakt deel uit van de persoonlijke vorming. Deze aandacht voor meer dan alleen onderwijs is terug te vinden in het brede assortiment van voorzieningen dat Fontys haar studenten biedt.

2.4 Begeleiding

Tijdens mijn afstuderen bij het Working Tomorrow programma, ben ik bij een practice geplaatst die past bij de richting van het project. Dit project valt onder de practice Information, Security & Architecture (ISA). Naast de procesmatige begeleiding vanuit Working Tomorrow verzorgd door Jeroen van Olderen wordt het project inhoudelijke begeleid vanuit de practice.

Deze begeleiding vanuit de practice ISA is in de vorm van een mentor welke functioneert als klankbord voor technische vragen, Tom van Driel is de mentor voor dit project. Tevens is de Practice Manager (PM) Arnold Dissel het contact voor kennis en ondersteuning vanuit de practice en eventueel noodzakelijke hard- en software. Voor de overige ISA gerelateerde afspraken en vragen kan ik terecht bij Ingrid Renssen, zij ondersteunt het ISA team en verzorgt interne communicatie.

De begeleiding vanuit Fontys wordt gedaan door Yvette Friebe, deze begeleiding is sturing van en controle op het proces en de inhoud van het gehele afstudeerproces. Dit om te voorkomen dat de afstudeeropdracht buiten mijn afstudeerrichting valt. In figuur 3 zijn alle partijen te zien met mijn positie als afstudeerder in het midden.



Figuur 3: Betrokken afstudeerstage.

3 DE OPDRACHT

De volgende opdrachtdomschrijving is het begin geweest van het project "Agent Migratie":

"Een agent kan fysiek reizen over een netwerk. Een agentplatform (de habitat van het agentteam) kan verspreid staan over verschillende machines bestaande uit verschillende typen hardware. Het is mogelijk om een agent te verplaatsen van de ene machine naar de andere. Dit kan handig zijn in een netwerk. Bijvoorbeeld om de rekenkracht dichterbij de data te krijgen of om een agent op een machine te zetten die je wilt monitoren. Naast de technische uitdaging van het migreren, kun je, je voorstellen dat je de migratie zelf in toom wilt houden. Hoe voorkom je dat agenten zomaar overal naartoe kunnen verhuizen?"

In deze opdrachtdomschrijving wordt duidelijk dat er "agents" zijn die binnen hun platform bepaalde activiteiten uitvoeren. Dit onderzoek geeft invulling aan het vraagstuk met betrekking tot de controle en het beheer van deze mobile agents. Om een specifieke richting aan het onderzoek te geven hebben er in het begin van het project naast het opstellen van het plan van aanpak welke is te vinden in bijlage A, gesprekken met experts en begeleiders plaatsgevonden. Tijdens deze gesprekken is naar voren gekomen dat er geen concrete "software agent" aanwezig was om te onderzoeken. Vanwege het gebrek aan een migrerende agent met een daadwerkelijk doel is de opdracht als volgt geïnterpreteerd:

"Tijdens het afstudeerproject bij Logica wordt er onderzoek gedaan naar het gebruik van software agents voor het verzamelen van informatie in een omgeving met verschillende platformen en de impact die het gebruik van deze software agents op de bedrijfsprocessen heeft. Naast de technische oplossing worden de security aspecten die op de software agents van toepassing zijn onderzocht."

Deze interpretatie van de opdracht geeft aan dat er naast het onderzoeken van de controle op agents tevens een systeem dat mobile agents gebruikt wordt ontwikkeld. Het ontwikkelde systeem is het resultaat van het toepassen van de kennis uit het onderzoek.

4 AANPAK ONDERZOEK

In dit hoofdstuk wordt de aanpak van het literatuuronderzoek en de toepassing van de geleerde kennis voor het ontwikkelen van een systeem uitgelegd. Er wordt een hoofdvraag opgezet waarmee een richting wordt gegeven aan de te onderzoeken bronnen. De hieruit voortvloeiende kennis wordt toegepast tijdens het ontwikkelen van een systeem.

4.1 Onderzoek

De Hoofdvraag die aan het begin van het literatuuronderzoek wordt onderzocht is als volgt:

“Hoe kan Logica software agents gebruiken voor het verzamelen van informatie in een omgeving met verschillende platformen waarbij de veiligheid van en de impact op de omgeving in acht wordt genomen?”

In deze hoofdvraag komen een aantal termen en doelen naar voren die moeten worden onderzocht. Om een richting aan het onderzoek te geven zijn de volgende deelvragen* opgezet.

- I. Wat zijn agents?
- II. Wat is een agent platform?
- III. Wat wordt er verstaan onder de migratie van een agent?
- IV. Hoe kan het doel van een migrerende agent worden bepaald?
- V. Hoe kan de impact van een agent georiënteerde oplossing bepaald worden?
- VI. Welke beveiliging is mogelijk voor het gekozen agent platform?
- VII. Hoe kan er controle op de migratie van een agent plaatsvinden?
- VIII. Via welke methode kan een systeem dat gebruik maakt van mobile agents worden ontwikkeld?

* Deze deelvragen verschillen van de deelvragen in het plan van aanpak. Deze wijziging is doorgevoerd tijdens de documentatie van het project om het verloop van het onderzoek op een gestructureerde wijze te laten verlopen en de leesbaarheid van dit document te verbeteren.

4.2 Systeemontwikkeling

Om het antwoord op de hoofdvraag te onderbouwen wordt er een systeem ontwikkeld dat gebruik maakt van de opgedane kennis uit het onderzoek. De ontwikkeling wordt uitgevoerd volgens de onderzochte ontwikkelmethode uit deelvraag VIII.

5 ONDERZOEK

Dit hoofdstuk bevat de theorie van het onderzoek dat is opgezet om een antwoord te kunnen geven op de hoofdvraag. Deze paragrafen behandelen de theorie achter de deelvragen.

5.1 Agents

In dit onderzoek wordt er gekeken naar het gebruik van software agents. Er zijn een aantal specifieke eigenschappen die aan een software agent kunnen worden toegekend. Deze eigenschappen worden in de gebruikte bronnen [FG97] en [MLL97] besproken en uitgelegd. Aan de hand van de informatie die uit deze bronnen is gehaald zijn er een aantal definities opgesteld die voor dit project worden gehandhaafd. De oorzaak voor het herdefiniëren van de software agents zijn de verschillen die tussen iedere informatiebron over software agents staan. In tabel 2 staan de verschillende eigenschappen die een software agent kan hebben gedefinieerd zoals deze voor dit onderzoek worden gebruikt.

Eigenschap	Definitie
Autonoom	Een stuk software dat staat is om op basis van eigen inzicht en waarnemingen te kunnen handelen.
Adaptief/Lerend	Een stuk software dat in staat is om te leren van ervaring opgedaan tijdens het uitvoeren van de functies en dit toepast op de manier waarop de functies worden uitgevoerd vanaf dat punt.
Mobiel	Een stuk software dat in staat is om zichzelf te migreren tussen platformen waarbij de software de status voor de migratie en eventuele data onthoudt en vervolgt.
Intelligent	Een stuk software dat in staat is om beslissingen te nemen bij het tegenkomen van bepaalde probleemstellingen.
Doel georiënteerd	Een stuk software dat bepaalde functies blijft herhalen tot het beoogde doel is behaald.
Communicatief/Samenwerkend	Een stuk software dat in staat is om samen te werken met andere stukken software en hier mee kan communiceren.
Flexibel	Een stuk software dat niet volgens een vast script handelt.
Passief	Een stuk software dat fysiek aanwezig is op een platform en wacht op een externe opdracht om bepaalde functies uit te voeren.
Karakter	Een stuk software met een eigen beeld van de omgeving, de doelen en de acties die het wil uitvoeren om die doelen te behalen.
Reactief	Een stuk software dat handelt op basis van een externe en/of interne trigger.

Tabel 2: Verschillende definities van een software agent.

Voor dit project wordt er onderzoek gedaan naar het gebruik van mobile agents, deze agents hebben de eigenschap dat ze kunnen migreren tussen agent platformen. Om tijdens dit project te kunnen refereren naar de definitie van een mobile agent is deze als volgt vormgegeven:

Een Mobile Agent is een stuk software dat in staat is om zichzelf te migreren tussen platformen waarbij de software de status voor de migratie en eventuele data onthoudt en vervolgt.

In de volgende twee paragrafen wordt uitgelegd wat een agent platform is en wat deze migratie nou eigenlijk inhoudt.

5.2 Agent platform

Aan de definitie van een Mobile Agent is te zien dat er agent platformen bestaan. Deze paragraaf legt uit wat dit zijn en waarom er voor een bepaald platform is gekozen.

Een agent platform is een software matige omgeving waar agents zich in kunnen bevinden. Er zijn momenteel verschillende platformen die ondersteuning bieden voor agent based oplossingen. De bron [L04] beschrijft een kort onderzoek naar de verschillende platformen in 2004. Een aantal van de onderzochte platformen zijn:

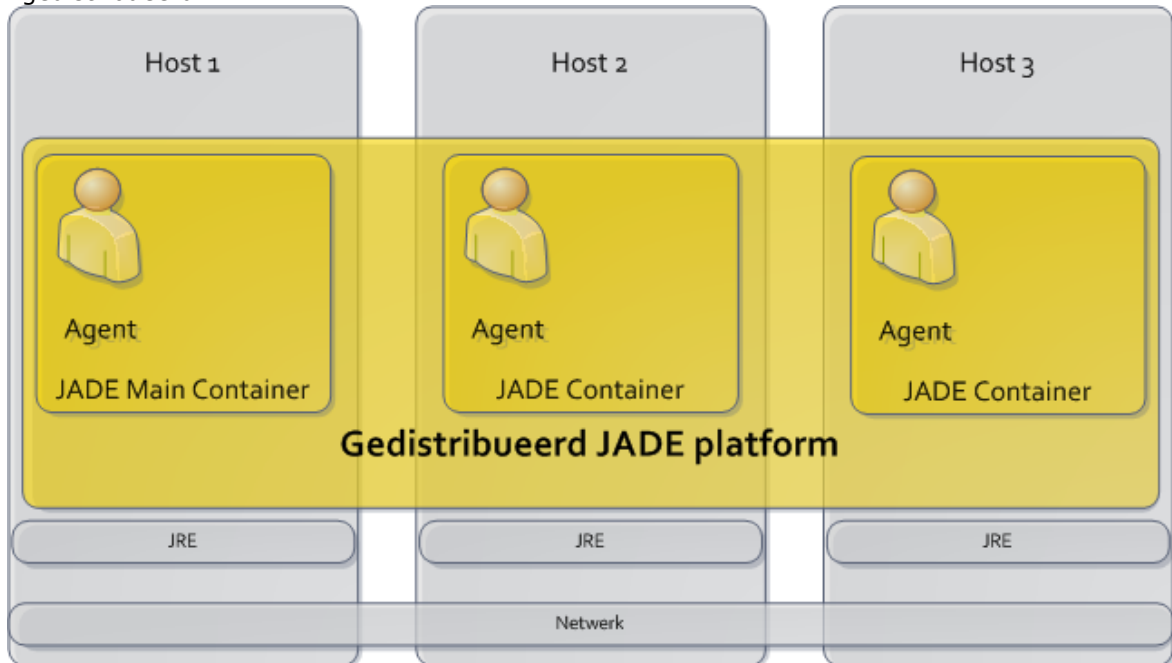
- Java Agent Development framework (JADE)
- Agent Development Kit (ADK)
- FIPA-OS
- JACK Intelligent Agents (JACK)
- Java Agent Services API (JAS)

Uit het onderzoek dat in 2004 is uitgevoerd was gebleken dat ADK, JADE en JACK aan de meeste criteria van de in het onderzoek opgestelde lijst voldeden. Van deze 3 platformen was JADE tevens het best gedocumenteerde en meest actieve platform.

Er zijn een paar redenen waarom er in dit onderzoek wordt gekozen voor het gebruik van JADE als agent platform.

- Bij eerder onderzoek binnen Logica is met dit platform gewerkt wat een voordeel kan zijn bij het tegenkomen van technische problemen.
- De volwassenheid van het JADE platform ligt erg hoog door de deelname van organisaties bij onderzoeken naar verbeteringen en uitbreidingen.
- De ondersteuning door deze groep van organisaties.
- Dankzij de ontwikkeling in JAVA is het platform op verschillende systemen inzetbaar.

Op Figuur 4 is een voorbeeld te zien van een JADE platform dat over meerdere systemen is gedistribueerd.



Figuur 4: Gedistribueerd JADE Platform.

Op de Figuur 4 is te zien hoe het platform concept werkt. Host 1, 2 en 3 zijn losse systemen waar een Java Runtime Environment (JRE) op staat geïnstalleerd. Dankzij deze omgeving kan het JADE platform worden gedraaid. Host 1 beschikt in dit voorbeeld over de main container en de overige hosts hebben een losse agent container. Doormiddel van communicatie over het netwerk kunnen agents met elkaar communiceren. Agents kunnen in deze opzet migreren tussen de verschillende containers gezien deze bij hetzelfde platform horen.

Toelichting:

De Main Container bevat de standaard agents die het platform beheren. Dit zijn de

- AMS agent: Het Agent Management System, verantwoordelijk voor platform management.
- DF agent: Directory Facilitator, deze voorziet het platform van een service lijst met aangeboden diensten van andere agents.
- RMA Agent: Remote Monitoring Agent, deze agent is aanwezig zodra het platform met een management venster wordt gestart.

De losse Container die aan een platform wordt gekoppeld bevat geen AMS, DF of RMA agent. De agents in deze containers worden centraal vanuit de Main Container beheerd.

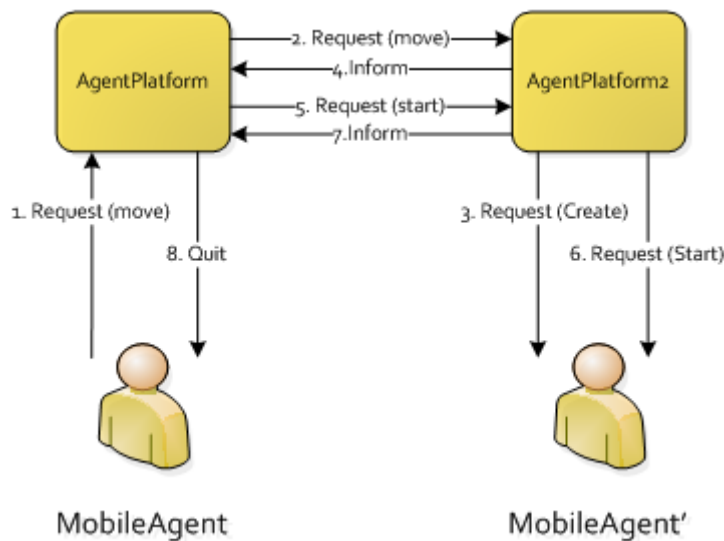
5.3 Migratie

Volgens de definitie van Mobile Agents hebben ze de eigenschap dat ze kunnen migreren tussen platformen. Om erachter te komen welke techniek hier achter zit en hoe dit werkt is gebruik gemaakt van de bronnen [BCTRM10] en [BCTR10].

Mobiliteitsprotocol conform FIPA

The Foundation of Intelligent Physical Agents (FIPA) is een organisatie die agent georiënteerde ontwikkelingen en standaarden hiervoor bespreekt en vaststelt. Services die worden ontwikkeld voor het JADE platform gebruiken doorgaans de standaarden die door de FIPA worden vastgesteld om ervoor te zorgen dat de ontwikkelingen op elkaar aansluiten.

Figuur 5 geeft het protocol weer van de Inter Platform Mobility (IPM) service die volgens deze standaarden is ontwikkeld. Dit protocol geeft aan hoe het migratieproces van een MobileAgent werkt.



Figuur 5: Mobiliteitsprotocol conform FIPA

MobileAgent' op figuur 5 is de tijdelijke kopie van de MobileAgent. De stappen die tijdens dit migratieproces worden gevolgd zijn in de onderstaande tabel toegelicht.

Stap	Toelichting
1	Er wordt door de MobileAgent een aanvraag gedaan om te migreren.
2	Dit wordt door het platform naar het remote platform gestuurd.
3	Het remote platform creëert een kopie van de agent.
4	Het remote platform geeft aan dat de agent is klaargezet.
5	Het platform meldt dat de kopie mag worden gestart.
6	Het remote platform start de kopie.
7	Het remote platform meldt dat de kopie is gestart.
8	Het platform stopt de originele agent.

Tabel 3: Toelichting Mobiliteitsprotocol.

5.4 Doel

De hoofdvraag die aan het begin van dit onderzoek wordt gesteld geeft een doel aan het onderzoek, echter niet aan het te onderzoeken onderwerp. Vanwege het toepassende karakter van dit onderzoek is het noodzakelijk dat de onderzochte theorie wordt toegepast op een realistische situatie.

Om dit te bewerkstelligen is de keuze gemaakt om het doel van de migrerende agents te bepalen in een Business Case. Deze Business Case wordt opgezet voor het hypothetische bedrijf X en moet invulling geven aan het doel van het systeem dat wordt ontwikkeld tijdens dit onderzoek. De inhoud van de Business Case wordt bepaald aan de hand van een aantal gesprekken met expert van Logica.

5.5 Impact

Het onderzoeken van de impact van het invoeren van een oplossing binnen een organisatie kan ervoor zorgen dat eventuele financiële risico's die de organisatie loopt inzichtelijk worden gemaakt. Tijdens het onderzoeken van informatie met betrekking tot het uitvoeren van een Business Impact Analyse (BIA) zijn er geen vaste methodes naar voren gekomen maar voorbeelden van of uitwerkingen van bestaande impact analyses. Om ervoor te zorgen dat er een aanpak wordt gebruikt waarbij eventuele vragen kunnen worden opgehelderd is er gekozen om een intern voorbeeld te hanteren vanuit Logica.

Aan de hand van dit voorbeeld worden de volgende stappen uitgevoerd:

- Het in kaart brengen van de afdelingen en processen binnen de organisatie.
- Het opstellen van risico's en het indelen van risico categorieën.
- Per proces de impact en frequentie van een risico bepalen.

Na het uitvoeren van deze stappen is bekend wat de totale impact van een risico is op een proces. De financiële impact is echter nog onbekend en er is geen bekende methode gevonden om de totalen om te zetten in financiële gegevens.

In de verschillende projecten tijdens de studie en stages is echter gebleken dat het management van een organisatie meer waarde hecht aan financieel inzicht dan aan een getal dat een "totaal" weergeeft. Tijdens het maken van de BIA wordt er gekeken op welke manier het mogelijk is om toch een financiële uitwerking te krijgen.

5.6 Security

Voor het JADE platform zijn er momenteel twee uitbreidingen die betrekking hebben op de beveiliging binnen JADE. Dit zijn TrustedAgents en de Security add-on.

De twee uitbreidingen zullen in de paragrafen 5.6.1 en 5.6.2 kort worden behandeld omdat ze niet worden gebruikt in het uiteindelijke concept. 5.6.3 beschrijft een onderzochte functie van het JADE platform die wel toepasbaar is voor de oplossing.

5.6.1 Trusted Agents Add-on

De TrustedAgents add-on is een beveiligingsservice die ervoor zorgt dat enkel agents met een token/sleutel toestemming krijgen om op het JADE platform te komen. Hiervoor is een TokenProvider en een TokenValidator nodig.

TokenProvider

De TokenProvider levert een token/sleutel aan een agent zodat deze zich kan identificeren. Deze sleutel kan worden gerealiseerd door het gebruik van een stuk software dat een sleutel kan generen op basis van vooraf bepaalde gegevens en een gekozen encryptie methode.

TokenValidator

De TokenValidator controleert de sleutel/token op geldigheid alvorens de agent toegang krijgt tot het platform.

Toepassing op het concept

Voor het ontwikkelen van het concept is geen gebruik gemaakt van de bovenstaande add-on door het ontbreken van een TokenProvider en TokenValidator. Deze onderdelen zouden in eventueel opvolgende projecten kunnen worden ontwikkeld om het concept systeem te verbeteren.

5.6.2 Security Add-on

De security add-on biedt een aantal mogelijkheden voor het JADE platform. Deze mogelijkheden zijn:

- Authenticiteit
- Toestemming
- Integriteit en confidentialiteit

Deze mogelijkheden bieden de volgende voordelen:

Authenticiteit

De authenticiteitcontrole zorgt ervoor dat een bepaalde gebruiker die containers en agents op een platform aanmaakt bekend is binnen het systeem.

Toestemming

Met gebruik van de authenticiteitcontrole kan men regels toepassen op de functies van agents op basis van de gebruiker die de agents heeft aangemaakt.

Integriteit en confidentialiteit

Met deze optie kan men ervoor zorgen dat de berichten die tussen agents binnen een platform of externe platformen voorzien worden van encryptie en handtekening. Momenteel zit in de module de aanname dat een bericht voorzien van een handtekening geldig is.

Toepassing op het concept

Zoals aangegeven in de JADE Security Guide [JB05] is de security module momenteel niet geschikt voor het gebruik van de mobiliteit van agents. Zodra deze module is aangepast voor het gebruik van mobile functies, kan er worden gekeken naar eventuele toepassingen.

5.6.3 SSL IMTP

Een van de nieuwe mogelijkheden van het JADE platform (versie 4.0) is het gebruik van een Secure Sockets Layer (SSL) verbinding om het inter-container message transport protocol (IMTP) tot stand te brengen. SSL is een encryptie-protocol dat zorgt voor een veilige verbinding op basis van certificeringen van 2 of meer partijen. IMTP is het protocol waar de agents binnen het JADE platform gebruik van maken om berichten naar externe containers of platformen te sturen. [V04] en [BCTRM10] gaan verder in op de werking en toepassing van deze functionaliteit.

Toepassing op het concept

Het in 5.3 besproken migratie protocol maakt gebruik van IMTP om de migratie van een agent mogelijk te maken. Dit houdt in dat een SSL-IMPT zorgt voor een beveiligde migratie.

Dit is mogelijk door de volgende stappen toe te voegen aan een agent based oplossing.

- Het aanmaken van een versleuteld certificaat in een opslagbestand.
- Het aanmaken van een bestand met de vertrouwde certificaten.
- Het aanmaken van de losse certificaten.
- De platformen opstarten met de juiste configuratie.
- Mobile Agents remote adressen met https in plaats van http laten gebruiken.
- Ieder vertrouwd platform voorzien van een certificaat en het bestand met alle voor het platform vertrouwde certificaten.
- Platformen forceren om enkel communicatie met vertrouwde platformen toe te staan.

Op deze manier is het mogelijk om een beveiligde migratie te creëren tussen twee platformen.

5.7 Controle

Om een mogelijke vorm van controle te verkrijgen op de Mobile Agents wordt er gebruik gemaakt van de bronnen [BCTRM10], [BCTR10] en de voorbeelden die voor het JADE platform beschikbaar zijn. In de bronnen wordt de werking van het platform uitgelegd en wordt er gerefereerd naar de voorbeelden van software agents die voor het platform beschikbaar zijn gesteld.

Tijdens het onderzoeken van de mogelijkheden van het JADE platform en de manier waarop agents kunnen worden ontwikkeld is een belangrijke functie naar voren gekomen. Het Agent Management Systeem (AMS). Dit systeem heeft de mogelijkheid om op aanvraag van agents andere agents binnen het JADE platform te activeren en deactiveren. Dit heeft het voordeel dat er een vorm van controle kan worden ontwikkeld om te voorkomen dat mobile agents zelf beslissen naar welk platform ze migreren en ze pas in te zetten zodra dit nodig is.

Tijdens het ontwikkelen van het systeem wordt er gekeken hoe deze functionaliteit toegepast kan worden.

5.8 Ontwikkelmethodes

Voor het ontwikkelen van het systeem is er aan het begin van het onderzoek gekeken welke ontwikkelmethode aansluit bij het te ontwikkelen systeem. De oplossing die gerealiseerd wordt is gebaseerd op het gebruik van verschillende software agents die een bepaalde functionaliteit toevoegen aan het systeem.

Voor het vergelijken van verschillende methodes is gekeken naar de volgende AGILE methodes:

- Feature Driven Development (FDD)
- Rational Unified Process (RUP)
- Extreme Programming

De redenen waarom er is gekozen voor één van de AGILE methodes zijn als volgt:

- Tijdens het verloop van het ontwikkelproces is het mogelijk dat er nieuwe ontwikkelingen ontstaan voor het gekozen platform waardoor er mogelijk aanpassingen komen in het ontwerp.
- Tijdens dit project wordt voor het eerst ontwikkeld in JAVA waardoor er mogelijk aanpassingen komen in de uitwerking van het systeem.

De AGILE methodes staan bekend om hun snelle en dynamische aanpak waardoor eventuele wijzigingen tijdens de ontwikkeling eenvoudig doorgevoerd kunnen worden.

De methode die het beste aansluit bij het te realiseren doel is Feature Driven Development (FDD). Deze methode is aangepast vanwege de grootte van het te ontwikkelen concept. Meer informatie over de originele methode is te vinden in het boek "a practical guide to feature-driven development" [PF02].

In de originele methode ontbreekt de noodzaak voor het noteren van de requirements van het te ontwikkelen systeem. Om ervoor te zorgen dat de wensen van de "klant" worden doorgevoerd in het systeem, worden de requirements opgenomen alvorens de methode wordt toegepast. De aangepaste versie van de FDD methode heeft hierdoor de volgende stappen:

1. Opnemen van de requirements

Tijdens de eerste stap worden de requirements van het systeem bekend gemaakt. Deze requirements worden ingedeeld in functionele requirements, data requirements, rapportage requirements en system requirements.

2. Ontwerp het globale model

Tijdens het uitvoeren van de tweede stap wordt de scope van het project en de context van het systeem onderzocht. Aan de hand van deze informatie wordt er een model gemaakt welke ter feedback aan de mentor en begeleiders wordt voorgelegd.

3. Ontwikkel een lijst met functionaliteiten

Uit het globale model zijn functionaliteiten van het systeem te herleiden, deze functionaliteiten worden verzameld in een lijst met prioriteiten.

4. Planning

In de originele methode wordt bij planning een verdeling gemaakt van de functionaliteiten en onderliggende klassen voor verschillende ontwikkel teams. Omdat er maar 1 persoon aan de ontwikkeling werkt wordt de planningsfase als volgt gebruikt: De lijst met functionaliteiten wordt conform de MoSCoW methode opgezet. Dit houdt in dat de functionaliteiten een prioriteit krijgen toebedeeld. Aan de hand van deze lijst worden de functionaliteiten verdeeld over de beschikbare ontwikkeltijd.

5. Ontwerp functionaliteit

Voor elke functionaliteit wordt er een onderzoek gedaan naar de werking en de opzet hiervan om ervoor te zorgen dat iedere functionaliteit waarde toevoegt aan het POC.

6. Bouw functionaliteit

Zodra de functionaliteit is ontworpen wordt er gestart met de bouw van de functionaliteit. Zodra dit is afgerond wordt de functionaliteit getest en indien noodzakelijk verbeterd.

6 TOEPASSING THEORIE

In dit hoofdstuk wordt de kennis die tijdens het onderzoek is opgedaan toegepast op de Business Case die in de eerste paragraaf wordt beschreven.

6.1 Doel

Om het doel te kunnen bepalen is er gebruik gemaakt van de kennis van experts binnen Logica. In het eerste gesprek met Tom van Driel (IT Consultant en mentor) zijn een aantal mogelijke doelen besproken, deze doelen zijn als voorbeeld gebruikt tijdens gesprekken met verschillende collega's binnen Logica. Het resultaat van deze gesprekken is een business case te vinden in bijlage B.

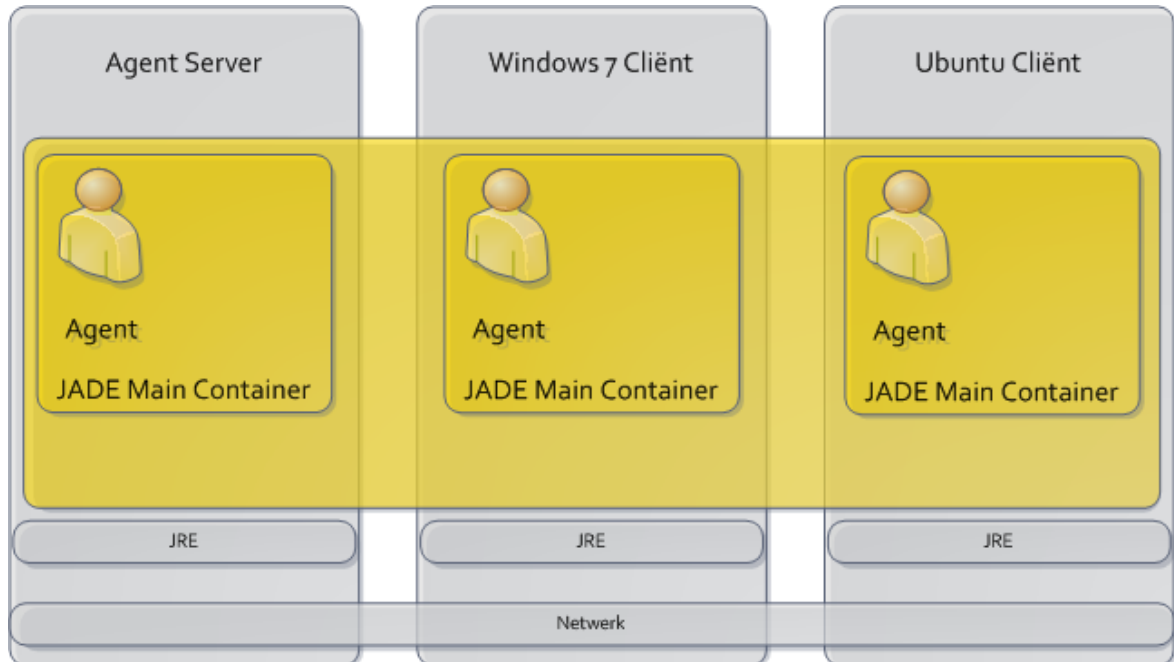
In de Business Case wordt het ontstaan van het project Agent Migratie bij het hypothetische bedrijf X beschreven. Het onderzoek dat plaats vindt is opgezet voor bedrijf X, dit bedrijf heeft door ervaring in de voorgaande jaren besloten om een opdracht aan Agent Based Services B.V. uit te geven. De opdracht is het in kaart brengen van het applicatielandschap van het bedrijf, echter ligt de manier waarop de oplossing wordt gerealiseerd vrij.

In de business case staat de informatie die door de software agents moet worden verzameld. Het management van bedrijf X wil namelijk weten wat de impact is van bepaalde applicaties, Tevens willen ze weten van welke systemen de applicaties afhankelijk zijn. Om deze informatie te verkrijgen zijn er een aantal gegevens nodig van de applicatie. De gegevens die moeten worden verzameld zijn in tabel 4 te vinden.

Gegeven	Beschrijving
Applicatiennaam	De naam van de applicatie.
Host naam	De naam van de host waar de applicatie op draait.
CPU	Het CPU gebruik van de applicatie
Geheugen	Het geheugen gebruik van de applicatie
Protocol	Het protocol waar de applicatie gebruik van maakt.
Port	De poort waar de applicatie gebruik van maakt.
IP	Het IP-adres waar de applicatie gebruik van maakt.
Interface	De interface waar de applicatie gebruik van maakt.
Status	De status van de verbinding.
Licentie	Licentie van de applicatie.

Tabel 4: Gegevens Applicatie.

Nu de noodzakelijke gegevens bekend zijn kan er worden gekeken naar de omgeving van de applicaties. De omgeving van de klant bestaat zoals in de Business Case vernoemd uit Microsoft en Linux systemen. Beide operating systemen bieden ondersteuning aan JAVA waardoor JADE kan worden gebruikt als platform. Zoals eerder in dit document uitgelegd kan JADE verspreid over meerdere systemen draaien. Echter is er voor deze situatie een andere opzet noodzakelijk. In figuur 6 is de opzet voor de platformen weergegeven zoals deze wordt gesimuleerd.



Figuur 6: JADE Platformen noodzakelijk voor het concept.

Zoals in figuur 6 is te zien bestaat de gesimuleerde omgeving uit een drietal systemen. De omgeving draait op de laptop die door Logica beschikbaar is gesteld. De Windows 7 en Ubuntu systemen worden virtueel op de laptop gehost met VMware player.

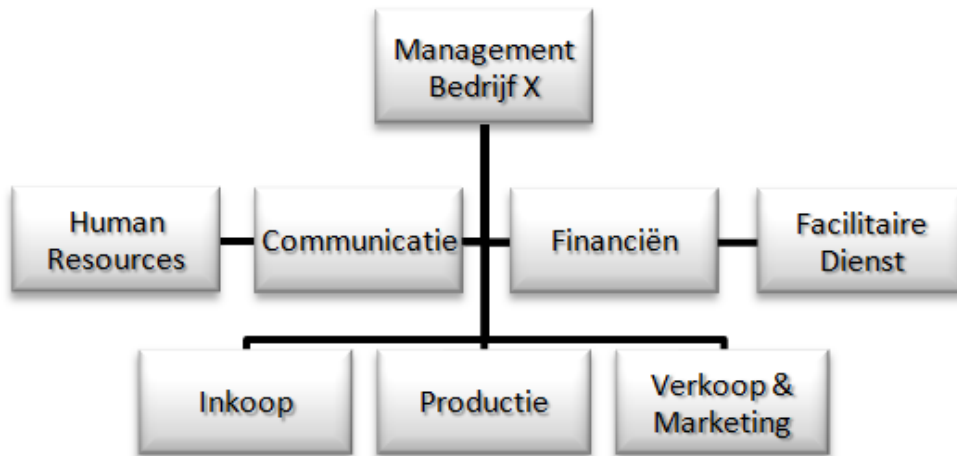
Het virtuele systeem "Test01" heeft de rol van Agent Server, Het virtuele systeem "Test02" dient als Ubuntu Cliënt en het virtuele systeem "Test03win" dient als Windows 7 cliënt. Alle systemen beschikken over een JRE en een installatie van JADE. De noodzakelijke communicatie tussen de platformen is mogelijk via het virtueel netwerk dat door VMware player wordt geboden.

Het verschil met het gedistribueerde platform in figuur 4 is als volgt uit te leggen. In figuur 4 is een gedistribueerd JADE platform te vinden waar losse containers verspreid staan over een aantal host systemen. In figuur 6 heeft iedere host systeem een eigen JADE omgeving met een Main Container. Deze opzet is gekozen vanwege de gewenste functionaliteit van de oplossing voor bedrijf X. Het concept moet namelijk op meerdere systemen binnen de cross platform omgeving draaien die via het netwerk met elkaar in verbinding staan. Mocht het voorkomen dat de verbinding tussen Host 2 of 3 en Host 1 wegvalt, kan het zijn dat de agent containers niet meer werken en de agents die in de container zitten verloren gaan.

Om dit te voorkomen is zoals in figuur 6 aangegeven ieder systeem uitgerust met het volledige platform met een eigen Main Container. Op deze manier kunnen de agents hun werkzaamheden ook zonder verbinding uit blijven voeren.

6.2 Impact

In de Business Impact analyse (bijlage C) is gekeken welke afdelingen er binnen de organisatie aanwezig zijn. Figuur 6 geeft het organigram van de organisatie weer.



Figuur 7: Organigram bedrijf X.

Zoals in bovenstaande Figuur is te zien bestaat bedrijf X uit 7 afdelingen en het management. Voor ieder van deze afdelingen is er gekeken welke bedrijfsprocessen aanwezig zijn en welke impact deze hebben op de afdeling. Hierna zijn de volgende risicocategorieën zijn opgezet met gebruik van de risico's opgesteld in Bijlage C:

Menselijke risico's:

Personeel dat tijdens of na de implementatie met de oplossing in contact komt kan door bijvoorbeeld onvoldoende kennis of met slechte bedoelingen zorgen dat er bepaalde fouten ontstaan.

Natuurlijke risico's:

De technische uitwerking van de oplossing is zoals bijna ieder systeem niet bestand tegen natuurlijke oorzaken zoals brand of waterschade. Dit is echter sterk afhankelijk van de opstelling van het systeem dat het platform van de oplossing host.

Hardware risico's:

De hardwarematige risico's die op het uiteindelijke concept van toepassing zijn vallen onder deze categorie, denk hierbij aan verstoringen van systemen en infrastructuur binnen een afdeling.

Software risico's:

De softwarematige risico's die de oplossing met zich mee kan brengen is gebruik gemaakt van [MBPK10]. Onder deze categorie vallen risico's die ontstaan bij het ontwerpen van de oplossing maar ook tijdens de ontwikkeling. In Bijlage C is een overzicht te vinden met de risico's per risicocategorie.

Nu de risico's voor bedrijf X bekend zijn kunnen ze worden toegepast op de processen per afdeling. Dit is gedaan door per risico per afdeling de verwachte frequentie en de verwachte impact in te vullen. Als een risico een Frequentie (F) van 4 heeft en een Impact (I) van 1 dan is het totaal (T) van het risico $4 * 1 = 4$. Op het moment dat een risico een lagere frequentie heeft maar een hogere impact kan dit risico schadelijker zijn voor de organisatie. Een F van 2 en een Impact van 5 geeft namelijk een totaalscore van 10. Op deze manier is het inzichtelijk welk risico of welke risico categorie de meeste impact heeft.

Tabel 5 geeft de totalen weer van frequentie, impact en totaal per risico categorie voor de afdeling productie. *Let op: De berekening Frequentie * Impact = Totaal is in deze tabel niet zichtbaar.* In bijlage C is de uitwerking per afdeling terug te vinden.

Risico	Frequentie	Impact	Totaal
Menselijke Risico's	8	8	16
Natuurlijke Risico's	3	9	9
Hardwarestoring	11	14	31
Softwarestoring	45	62	111
Totaal	67	93	167

Tabel 5: Impact risico's Afdeling productie.

Na het onderzoeken van de risico's per afdeling wordt de financiële impact van de risico's op Bedrijf X berekend. Voor deze berekening zijn een aantal rekencijfers vereist. Voor iedere core business afdeling wordt een percentage van de omzet gebruikt. De afdelingen dragen ieder 33% mee aan de jaaromzet. Voor de ondersteunende afdeling wordt gerekend met de kosten die bedrijf X jaarlijks aan de afdelingen heeft.

Afdeling	Rekencijfers
Productie	€ 305.382.000,00
Verkoop & Marketing	€ 305.382.000,00
Inkoop	€ 305.382.000,00
Human Resources	€ 9.254.000,00
Facilitaire Dienst	€ 9.254.000,00
Communicatie	€ 18.508.000,00
Financiën	€ 27.762.000,00

Tabel 6: Rekencijfers per afdeling.

De financiële impact van ieder risico per afdeling wordt berekend met behulp van de maximale score van de risico's. Hiervoor wordt bij ieder risico de frequentie en de impact op 5 gezet wat resulteert in de maximale totalen per risico. De bijdrage in de omzet wordt door dit totaal gedeeld zodat ieder punt een waarde in euro's krijgt. Deze waarde wordt gebruikt om per afdeling de kosten per risico te berekenen. De volgende waarden zijn gedefinieerd voor de berekening:

Waarde	Omschrijving
F	Financiële impact
O	Omzet/Kosten
T	Totale impact van alle processen
P	Impact van het proces
M	Maximaal risico totaal
R	Score risico categorie

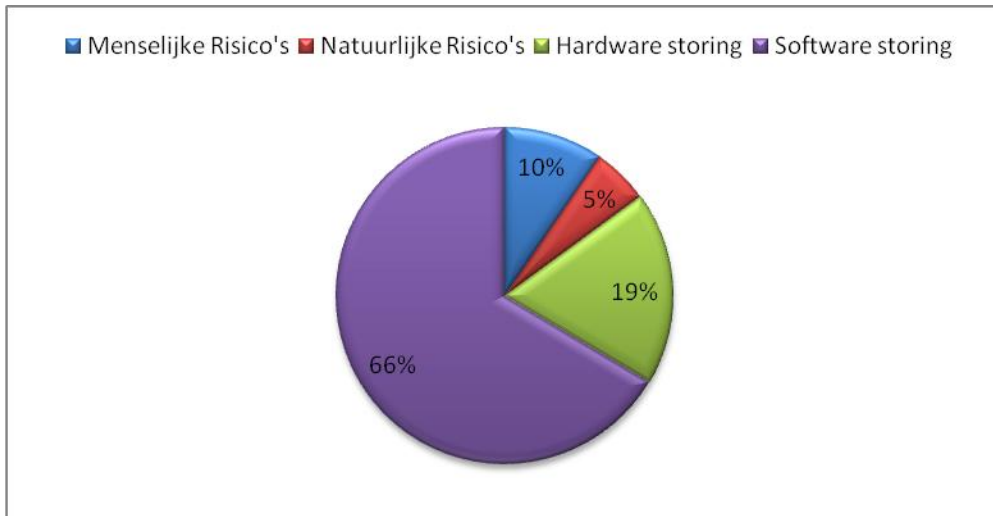
Tabel 7: Uitleg formule waarden.

De berekening is als volgt opgezet: $F = ((O / T) * P) / M * R$

Voor deze business impact analyse wordt gebruik gemaakt van de totale kosten per risico categorie.

De berekening die hiervoor wordt gebruikt is: $F = (O / M) * R$

Als we deze berekening toepassen op de afdeling productie resulteert dit in de volgende gegevens:



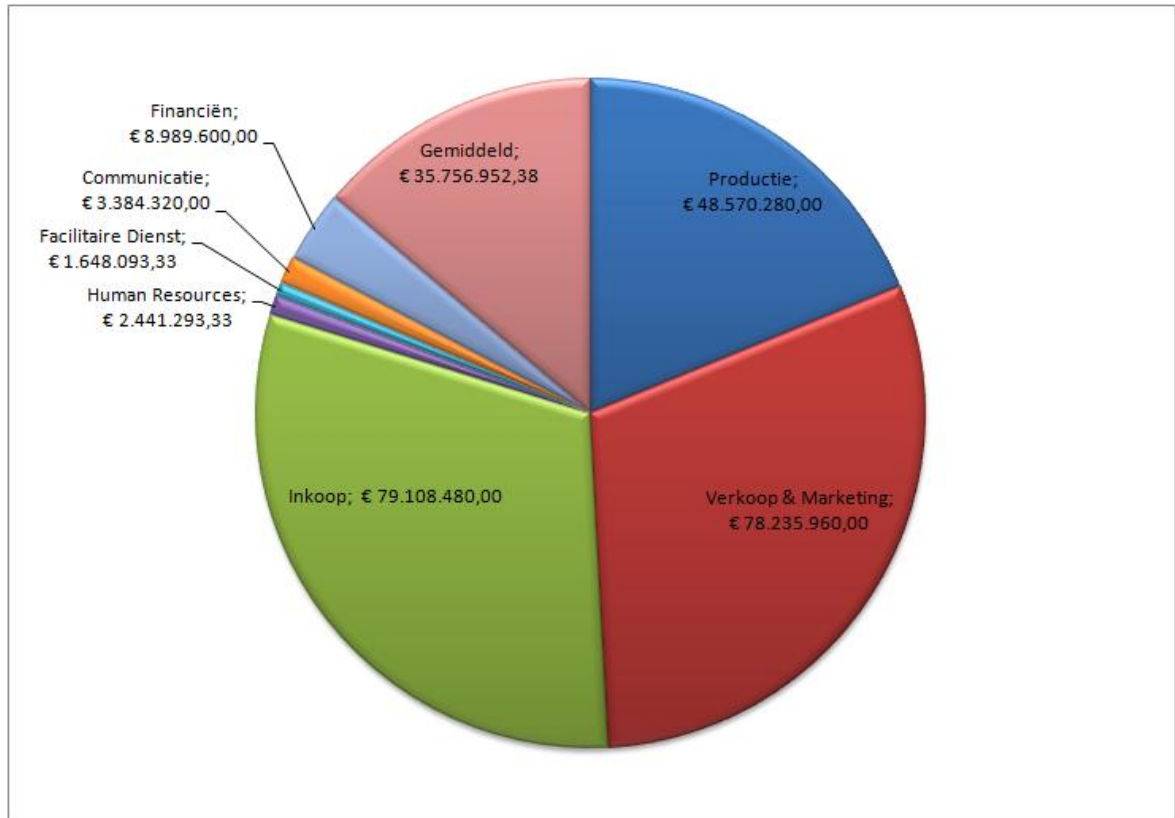
Figuur 8: Percentage Financiële impact risico's afdeling productie.

Risico	Score categorie	risico Percentage	Financiële impact
Menselijke risico's	16	10	€ 4.653.440,00
Natuurlijke risico's	9	5	€ 2.617.560,00
Hardware storing	31	19	€ 9.016.040,00
Software storing	111	66	€ 32.283.240,00
Totaal	167	100	€ 48.570.280,00

Tabel 8: Financiële impact risico's afdeling productie.

Na het uitvoeren van de impact analyse is naar voren gekomen dat de risico's de meeste impact hebben op de afdelingen Inkoop en Verkoop & Marketing. De Productieafdeling loopt minder risico omdat de vitale systemen binnen deze afdeling zijn afgeschermd van het bedrijfsnetwerk. De grootste risico's komen voort uit de software categorie, dit zijn risico's waar tijdens het ontwerpen van software rekening moet worden gehouden om te voorkomen dat ze tijdens de ontwikkeling ontstaan.

In figuur 9 wordt de impact op de verschillende afdelingen in euro's weergegeven zodat het inzichtelijk is voor welke afdeling de risico's de grootste impact hebben.



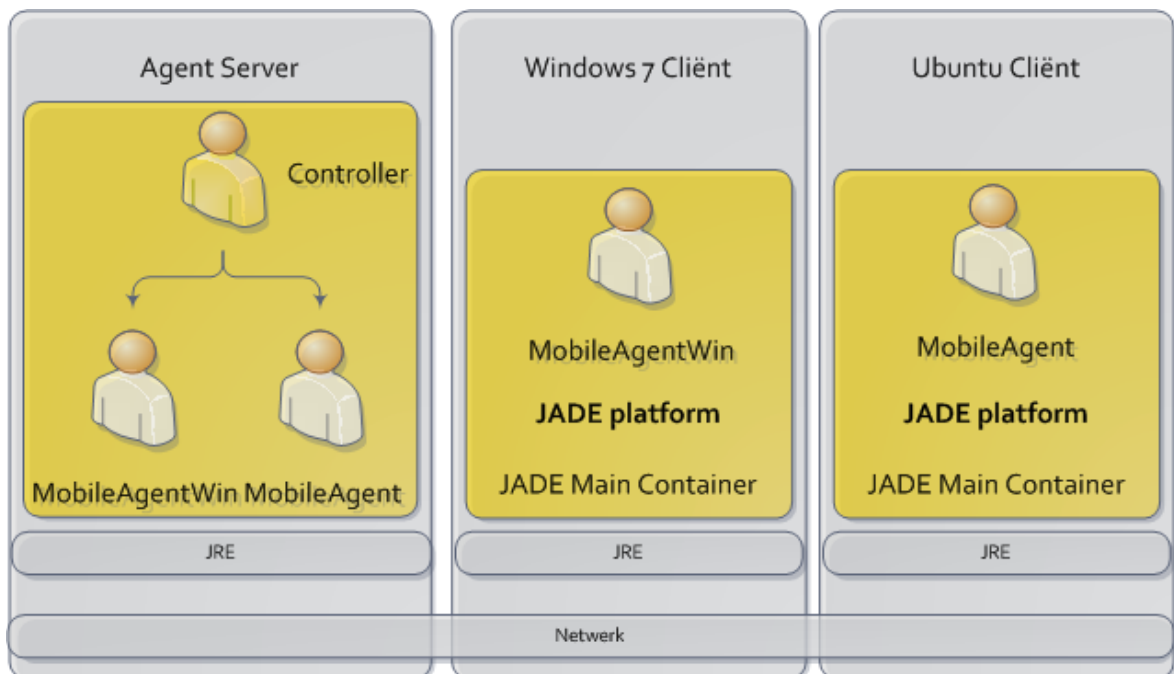
Figuur 9: Overzicht impact afdelingen bedrijf X.

Let op:

De in deze analyse gebruikte risico's zijn voorbeelden van mogelijke risico's, deze moeten per analyse gespecificeerd worden. De impact van een bepaald risico kan het beste met het management dat verantwoordelijk is voor bepaalde processen besproken worden, deze kunnen namelijk vertellen hoeveel geldt het uitvallen van een bepaald systeem kost.

6.3 Controle

Tijdens de realisatie van het systeem wordt een "Controller" agent ontwikkeld om aan te tonen dat het mogelijk is om een systeem te ontwikkelen dat gebruik maakt van mobile agents zonder de controle te verliezen. In het onderstaande figuur wordt het idee weergegeven zoals deze is ontstaan tijdens het bestuderen van de documentatie die voor het JADE platform beschikbaar is.



Figuur 9: Basis idee "Controller".

In dit concept worden de mobile agents "MobileAgentWin" en "MobileAgent" aangemaakt door de "Controller" zodra er een opdracht van de gebruiker binnenkomt die op de Windows 7 cliënt of de Ubuntu cliënt moet worden uitgevoerd.

Op het moment dat de gewenste mobile agent is geactiveerd dient de controller de opdracht aan de mobile agent toe te kennen.

De mobile agent migreert naar het opgegeven platform en voert de gewenste taak uit. Zodra de taak is afgerond migreert de mobile agent terug en wordt verwijderd. De Controller geeft de resultaten terug aan de gebruiker.

7 SYSTEEMONTWIKKELING

In dit hoofdstuk wordt het concept met de bijbehorende functionele en de technische uitwerkingen besproken.

De ontwikkeling van het systeem volgt de stappen van de aangepaste FDD methode. De uitwerkingen van deze stappen zijn te vinden in het Functioneel Ontwerp (FO) te vinden in Bijlage D en het Technisch Ontwerp (TO) te vinden in Bijlage E.

7.1 Toelichting concept

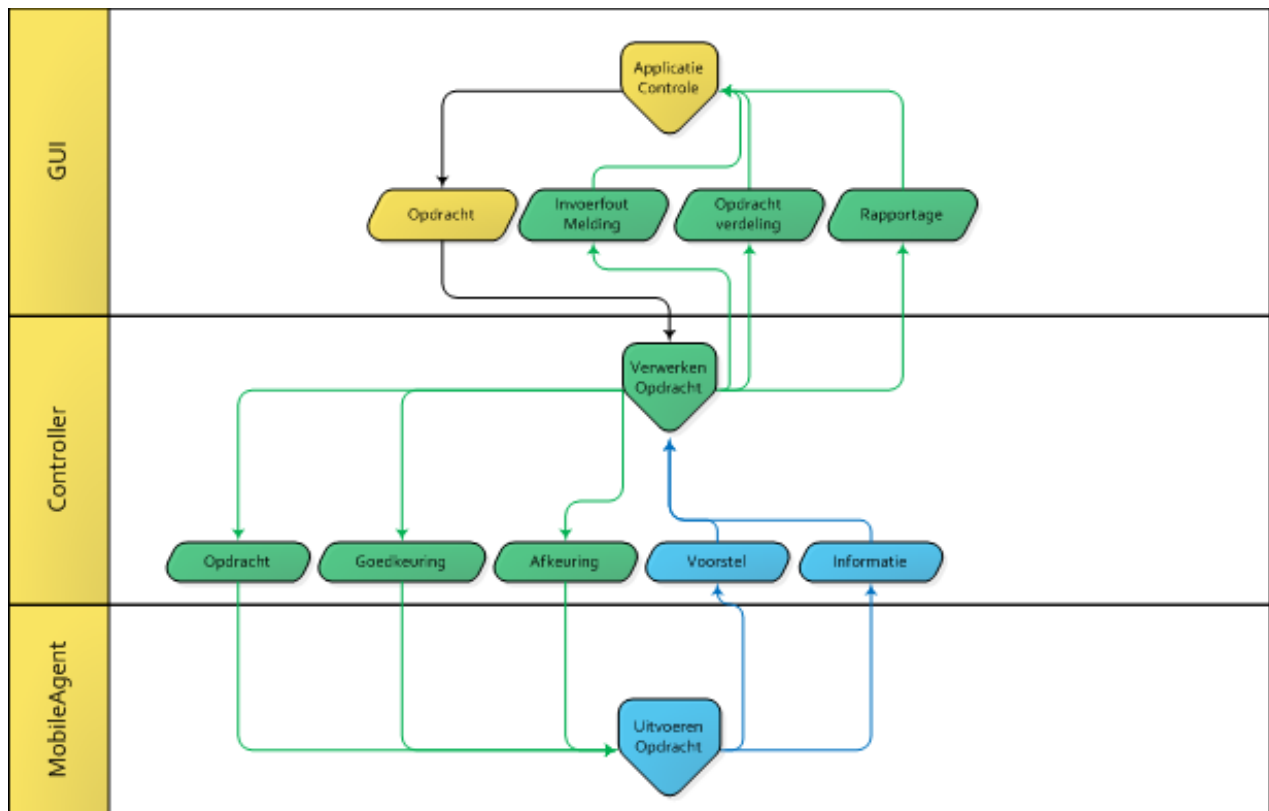
In de komende paragrafen worden een aantal termen gebruikt die terug zijn te vinden in het FO en TO. In de onderstaande tabel worden deze kort beschreven.

Term	Beschrijving
User	De user is de persoon die het systeem een opdracht geeft.
GUI	De Graphical User Interface (GUI) is het onderdeel van de toepassing dat ervoor zorgt dat de user kan communiceren met de controller.
Controller	De controller is de agent die ervoor zorgt dat alle opdrachten door de correcte mobile agent worden uitgevoerd. Alles wordt door de controller gestuurd en gecontroleerd.
MobileAgent	De Mobile Agent is een mobiele agent die in opdracht van de controller naar een extern Linux platform kan migreren om gegevens op te halen.
MobileAgentWin	De MobileAgentWin is een mobiele agent die in opdracht van de controller naar een extern Microsoft platform kan migreren om gegevens op te halen.
Platform	Het platform dat wordt genoemd is JADE.

Tabel 9: Termen concept.

7.2 Globaal model

Het globale model dat is opgezet voor het concept is in figuur 10 weergegeven. In het FO zijn de onderliggende niveaus weergegeven die richting hebben gegeven aan de opzet van het concept. In dit document worden de requirements van het systeem besproken die aan de hand van de Business Case zijn opgesteld.



Figuur 10: Globaal model functioneel ontwerp.

7.3 Functionaliteiten

In deze paragraaf worden de functionaliteiten van de GUI, Controller en de MobileAgents besproken, deze zijn ontwikkeld op basis van de functionele requirements uit het FO.

7.3.1 De GUI

De GUI beschikt over de volgende functionaliteiten.

- Invoermogelijkheid voor opdrachten
- Een startknop voor het systeem
- Mogelijkheid tot publiceren van meldingen
- Doorsturen van opdrachten aan de Controller
- Status van opdrachten weergeven
- Een overzicht met de resultaten van de opdrachten

7.3.2 De Controller

De Controller beschikt over de volgende functionaliteiten.

- Opdracht controle
- Meldingen genereren voor gebruiker
- MobileAgents selecteren, aanmaken en starten
- Opdracht versturen
- Voorstellen controleren
- Informatie en meldingen verwerken
- Opdrachtstatus beheren
- Rapportage aanmaken

7.3.3 Mobile Agents

De mobile agents die voor het concept zijn ontwikkeld hebben de volgende functionaliteiten:

De MobileAgent (Linux):

- Opdrachten aannemen
- Voorstellen/Weigeren opdracht (Op basis van type platform)
- Migratie (Conform het Mobiliteitsprotocol)
- Informatie ophalen uit een Linux systeem
- Meldingen versturen naar de Controller

De MobileAgentWin (Microsoft):

- Opdrachten aannemen
- Voorstellen/Weigeren opdracht (Op basis van type platform)
- Migratie (Conform het Mobiliteitsprotocol)
- Informatie ophalen uit een Microsoft systeem
- Meldingen versturen naar de Controller

7.4 Planning

Bij het ontwikkelen van de functionaliteiten is er gekeken welke noodzakelijk zijn en welke van elkaar afhankelijk zijn. De prioriteit is volgens de MoSCoW methode ingedeeld, dit houdt in dat de volgende prioriteiten zijn toegekend per functionaliteit.

- Prioriteit 1: Deze eis moet terugkomen in het eindresultaat.
- Prioriteit 2: Deze eis is zeer gewenst maar zonder is het product bruikbaar.
- Prioriteit 3: Deze eis mag alleen aan bod komen als er tijd genoeg is.
- Prioriteit 4: Deze eis zal in dit project niet aan bod komen.

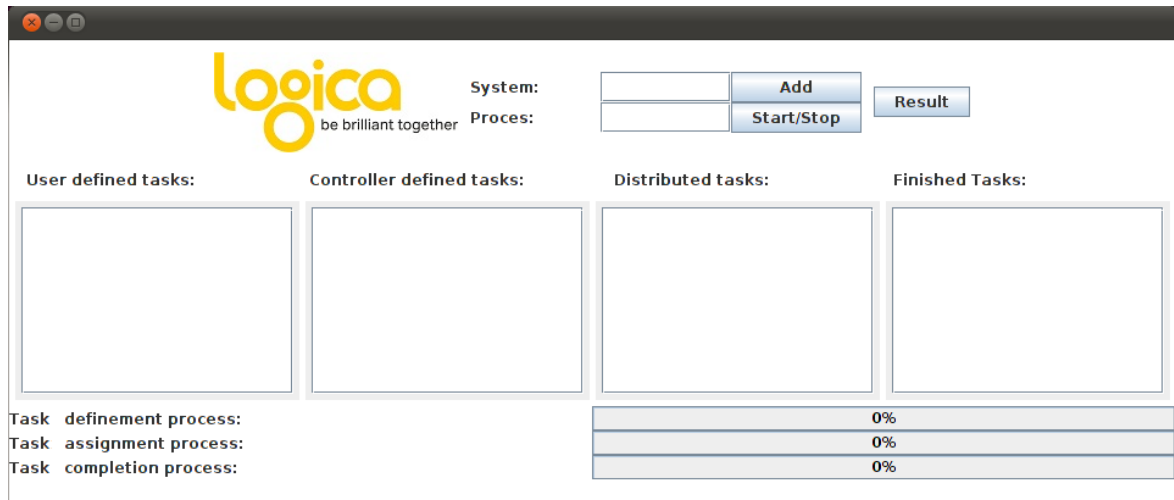
Op basis van deze prioriteiten is een planning gemaakt voor de ontwikkeling. De GUI, Controller, MobileAgentWin en MobileAgent zijn echter door de onderlinge afhankelijkheid parallel aan elkaar ontwikkeld waardoor de originele ontwikkelplanning niet is gehandhaafd.

7.5 Resultaat

Het resultaat van bovenstaande ontwikkeling is een werkend systeem dat gebruik maakt van een "Controller" agent om de MobileAgents te controleren en te beheren bij het uitvoeren van de aan hun toebedeelde taken. Dit systeem toont aan dat het mogelijk is om de migratie van migrerende agents te controleren en beheren.

7.5.1 GUI

Op figuur 11 is de GUI van het systeem te zien zoals deze tijdens het ontwikkelproces is vormgegeven.



Figuur 11: Graphical User Interface Concept.

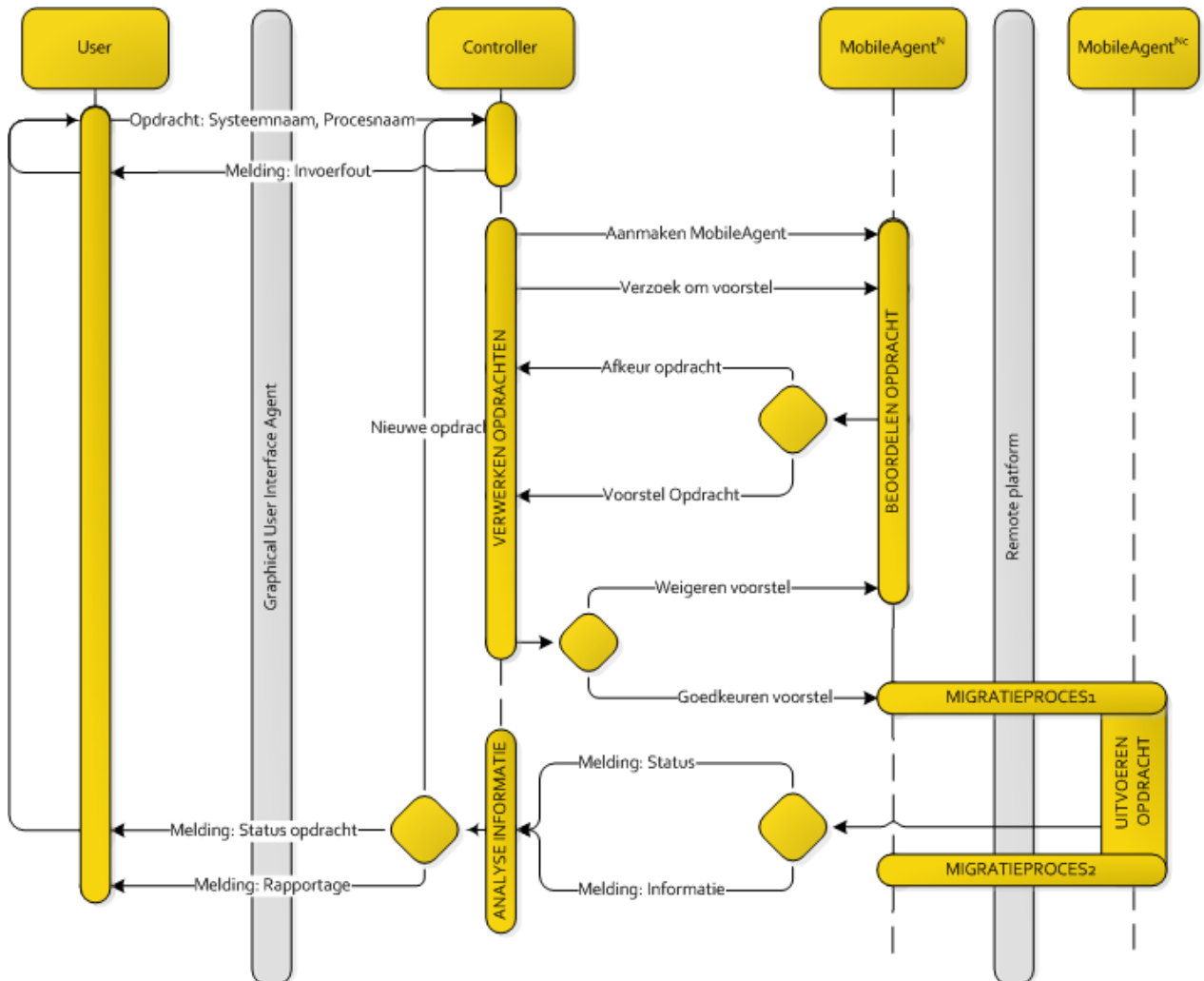
De onderstaande tabel geeft weer uit welke componenten de GUI is opgebouwd.

Component	Beschrijving
Input 1	Invoerveld voor een systeemnaam
Input 2	Invoerveld voor een Procesnaam
Add Button	Knop voor het toevoegen van de opdracht.
Start/Stop Button	Knop voor het starten met uitvoeren van de opdrachten.
Result Button	Knop voor het opvragen van de resultaten.
User Defined Tasks	Lijst met onverwerkte opdrachten.
Controller defined tasks	Lijst met opdrachten die aan de controller zijn doorgegeven.
Distributed tasks	Lijst met opdrachten die door de controller zijn uitgegeven.
Finished tasks	Lijst met afgesloten opdrachten.
Task definement process	Voortgang van de opdrachtuitgifte aan de controller.
Task assignment process	Voortgang van de opdrachtuitgifte van de controller aan MobileAgents.
Task completion process	Voortgang van alle opdrachten.

Tabel 10: Uitleg Graphical User Interface.

7.5.2 Systeem

Figuur 12 geeft de opzet voor de GUI, Controller en MobileAgent^N weer. Tabel 11 op de volgende pagina bevat de toelichting voor het ontwerp. Meer informatie over dit ontwerp is te vinden in het TO.



Figuur 12: Opzet systeem.

Begrip	Toelichting
MobileAgentN	De gewenste MobileAgent benodigd voor de opdracht. Keuze uit de MobileAgent of MobileAgentWin.
MobileAgentNc	MobileAgentN na het migratieproces.
Graphical User Interface Agent	Agent die de communicatie tussen User en Controller regelt.
Remote platform	Het JADE platform op een extern systeem
Verwerken opdrachten	Lineair proces waarin de opdrachten worden gecontroleerd, een MobileAgent wordt aangemaakt, de opdracht wordt uitgegeven en het voorstel wordt gekeurd.
Analyse informatie	Dit proces handelt binnenkomende meldingen af, die kunnen wijzigingen in de status zijn of de resultaten van de opdracht.
Beoordelen opdracht	Het controleren of de gegeven opdracht uitvoerbaar is voor de agent, en het zenden van een voorstel tot uitvoer.
Migratieproces1	Migreren van de MobileAgent naar het remote platform.
Uitvoeren opdracht	Het ophalen en verzenden van de informatie.
Migratieproces2	Migreren van de MobileAgent naar het server platform.

Tabel 11: Toelichting Systeem.

8 CONCLUSIE(S) EN AANBEVELING(EN)

Dit onderzoek heeft zich gericht op het onderzoeken van migrerende agents, tijdens het onderzoek is er gekeken naar een vorm van controle en beveiliging bij het gebruik van deze migrerende agent. De theorie achter de technologie is beschreven en toegepast op een fictieve situatie. Aan het einde van dit onderzoek is de theorie ondersteund door de ontwikkeling van een agent georiënteerd systeem.

Met het onderzoeken en definiëren van de term "Mobile Agent" en het onderzoeken van de mogelijkheden die het JADE platform kan bieden is de basis voor het onderzoek gelegd. Er is een fictieve Business Case ontwikkeld om de geleerde kennis op toe te passen. Aan de hand van deze business case is er een doel voor de Mobile Agents opgezet, namelijk het verzamelen van gegevens in een crossplatform omgeving.

Naast de Business case is er een Business Impact Analyse uitgevoerd om als voorbeeld te dienen voor een daadwerkelijke toepassing. Tijdens het uitvoeren van deze analyse is duidelijk geworden hoe risico's die aan het gebruik van een oplossing zitten, op een in dit geval fictieve situatie onderzocht kunnen worden. De gebruikte risico's zijn echter een kleine greep uit de mogelijke risico's die bij compleet proces van ontwikkeling tot in gebruik name van een oplossing aanwezig kunnen zijn. Het advies is het onderzoeken van de impact van de toepassing per situatie met gebruik van de kennis van het management van de afdelingen, het management heeft namelijk inzicht welke schade een bepaald risico met zich mee kan brengen.

Omdat security steeds belangrijker is bij de ontwikkeling van nieuwe oplossingen en technieken heeft dit onderzoek tevens aandacht besteed aan dit onderwerp. Er is gekeken naar de mogelijkheden binnen het JADE platform op het gebied van beveiliging, en welke noodzakelijke beveiliging te realiseren is bij het gebruik van gevoelige data. Voor het JADE platform zijn er twee uitbreidingen (gedeeltelijk) ontwikkeld, de TrustedAgents add-on en de Security add-on. Beide uitbreidingen zijn tijdens de uitwerking van het concept echter niet gebruikt. Dit in verband met het ontbreken van nog te ontwikkelen onderdelen voor de TrustedAgents add-on en de missende ondersteuning van de Security add-on die door de ontwikkelaars zelf al wordt aangegeven in hun documentatie. Voor de toekomst is het interessant om voor TrustedAgents de missende onderdelen te ontwikkelen en toe te passen. Op langer termijn zal de Security add-on echter meer functionaliteiten bieden vanwege de community die hier achter zit.

Op dit moment is een oplossing die gebruik maakt van Mobile Agents te beveiligen door de communicatie van en naar de platformen via een beveiligde verbinding (SSL) te forceren. Met het gebruik van deze functionaliteit maken de platformen enkel contact met de door hun vertrouwde platformen op basis van certificaten die van encryptie zijn voorzien.

De werking van de hierboven uitgelegde conclusie is bewezen met het proof of concept dat tijdens het onderzoek is gerealiseerd. Deze toepassing maakt gebruik van een beveiligde verbinding en een "Controller" agent om de migratie van Mobile Agents veilig en onder controle te houden.

Voor de toekomst is het van belang dat de ontwikkelingen op het gebied van agents en platformen in de gaten worden gehouden. Door de grote community achter het JADE platform en de overige onderzoeken naar agent georiënteerde oplossingen bestaat de kans dat het aantal toepassingen van agent technologie snel groeit. Het resultaat hiervan zullen nieuwe standaarden zijn die helpen bij het ontwikkelen van veilige agent georiënteerde oplossingen.

EVALUATIE

Tijdens dit project heb ik veel zaken geleerd met betrekking tot het uitvoeren van een onderzoek en het succesvol afronden van een zelfstandig project. Dankzij de verschillende methodes en invalshoeken die de opleiding me heeft aangeleerd heb ik de vraagstukken uit het onderzoek op de juiste manier weten te behandelen. Zonder de ervaring van het werken met en het toepassen van verschillende technieken en methodes was het een stuk moeilijker geweest om in het diepe van deze innovatieve opdracht te springen.

Toen ik aan de opdracht begon had ik al een besluit gemaakt over de technische uitwerking. Omdat er geen mobile agents aanwezig waren om mijn onderzoek op te richten, zou ik een eventueel concept met behulp van de programmeertaal Python realiseren. Dit omdat ik hier al enige ervaring mee had opgedaan in het verleden. Tijdens het onderzoeken van de werking van agents en het zoeken naar informatie ben ik erachter gekomen dat het JADE platform dat gebruik maakt van JAVA in menig onderzoek naar voren is gekomen. Net voor het opleveren van mijn plan van aanpak heb ik besloten om de beperking tot Python uit het plan te halen. Deze keuze heeft me extra werk opgeleverd omdat ik mijn onderzoek van richting moest veranderen, het heeft me echter ook extra kennis opgeleverd. Naast het onderzoek moest ik me namelijk bekend maken met het ontwikkelen met gebruik van JAVA, dit heeft als resultaat dat ik meer kennis in het project heb opgedaan dan ik in eerste instantie dacht.

Tijdens het vergaren van bronnen was ik in het begin van het onderzoek niet echt op een gestructureerde manier bezig, met als gevolg dat ik tijdens het verwerken van informatie af en toe op zoek ben geweest naar dat ene document uit de grote hoop waar ik de kennis uit heb opgedaan. Toen ik dit besepte heb ik iedere bron hernoemt en meteen toegevoegd in het document dat er gebruik van maakte.

De feedback momenten waar ik gebruik van heb gemaakt tijdens het project hebben mij geleerd dat ik meer na moest denken over de manier waarop de informatie wordt geïnterpreteerd door de lezer. Hierdoor heb ik mijn eigen schrijfstijl aangepast om ervoor te zorgen dat het bericht dat ik duidelijk wil maken ook aankomt op de manier die ik bedoel.

De grootste fout die ik heb gemaakt tijdens mijn project is het direct toepassen van de geleerde kennis zonder dit goed te documenteren. Hier heb ik een mooi leermoment mee gecreëerd tijdens de uren die ik heb gespendeerd om de documentatie aan te passen.

De scope die ik mezelf heb opgelegd door het willen betrekken van het bedrijfskundige aspect en het ontwikkelen van een werkend systeem heeft ervoor gezorgd dat ik veel verschillende dingen heb moeten onderzoeken en bekijken. Als ik echter terugkijk op de gehele stage ben ik blij met het resultaat en de kennis die ik heb opgedaan tijdens het hele proces.

De omgeving waar ik de opdracht in heb uitgevoerd bij Logica, tussen de andere Working Tomorrow studenten, is me erg bevallen. Dankzij de verschillende collega's en hun projecten om me heen had ik de mogelijkheid om de bekende vastloop momenten te doorbreken en snel weer verder te kunnen werken.

LITERATUURLIJST

[BCTR10] BELLIFEMINE F., CAIRE G., TRUCCO T., RIMASSA G., 2010, JADE PROGRAMMER'S GUIDE, TILAB S.P.A., CSELT S.P.A., TELECOM ITALIA S.P.A.

[BCTRM10] BELLIFEMINE F., CAIRE G., TRUCCO T., RIMASSA G., MUNGENAST R., 2010, JADE ADMINISTRATOR'S GUIDE, JADE BOARD, TILAB S.P.A., CSELT S.P.A.

[FG97] FRANKLIN P., GRAESSER A. 1997. IS IT AN AGENT, OR JUST A PROGRAM?: A TAXONOMY FOR AUTONOMOUS AGENTS. MEMPHIS: INSTITUTE FOR INTELLIGENT SYSTEMS, UNIVERSITY OF MEMPHIS.

[JB05] JADE BOARD, 2005, JADE SECURITY GUIDE, TILAB.

[L04] LESZCZYNA R, EVALUATION OF AGENT PLATFORMS, 2004, JOINT RESEARCH CENTRE ISPRA, ITALY.

[MBPK10] MARTIN B., BROWN M., PALLER A., KIRBY D., 2010, CWE/SANS, TOP 25: MOST DANGEROUS SOFTWARE ERRORS, MITRE, SANS.

[MLL97] MERZ. M., LIBERMAN B., LAMERSDORF W. 1997 USING MOBILE AGENTS TO SUPPORT INTERORGANIZATIONAL WORKFLOW MANAGEMENT - WORKFLOW MANAGEMENT WITH MOBILE AGENTS. HAMBURG: UNIVERSITY VOGT-KÖLLN.

[PF02] PALMER S., FELSING J., 2002, A PRACTICAL GUIDE TO FEATURE-DRIVEN DEVELOPMENT (THE COAD SERIES), ASIN: 0130676152, PRENTICE HALL PTR.

[V04] VITAGLIONE G, 2004, MUTUAL-AUTHENTICATED SSL IMTP CONNECTIONS: SETTING UP SECURE COMMUNICATION AMONGST JADE CONTAINERS, TELECOM ITALIA LAB.

BIJLAGEN

A Plan van aanpak

B Business Case

C Business Impact Analyse

D Functioneel ontwerp

E Technisch ontwerp

Adres:

Logica
Working Tomorrow
Flight Forum 3250
Eindhoven
The Netherlands

Contact:

M. Hoogeveen
Trainee Working Tomorrow
T: + 31 88 56 46181
M: +31 6 1151 6852
E: maikel.hoogeveen@logica.com
E2: maikel.hoogeveen@mobileemail.vodafone.nl
www.logica.com