Literatuurstudie:

# Inleiding

Artificiële intelligentie is in elk vakgebied een essentieel onderdeel om aan de top van je sector te blijven in het jaar 2024. Daarom is ook het bedrijf Itineris zich in dit onderwerp aan het verdiepen. Itineris is een bedrijf dat zich volledig inzet op het vergemakkelijken van de business processen aan de hand van digitale tools voor bedrijven die handelen met basisvoorzieningen zoals water en elektriciteit.

Itineris zet de nadruk op efficiëntie aan de hand van geautomatiseerde processen, een versimpelde UX (User Experience) en de mogelijkheid aan klanten om zichzelf te helpen. Dit resulteert voor de klanten dan in een gegarandeerde lagere service kost.

Itineris zorgt ook voor betere inzichten voor de klanten aan de hand van omni-channel features die je toelaten van proactief de tevredenheid van je klant in de goede richting te sturen. Je kan business processen vergemakkelijken aan de hand van een flexibele, snelle en super aanpasbare third-party integration.

De software die Itineris aanbied maakt gebruik van het MS-Dynamics 365 ERP-pakket. Microsoft Dynamics 365 is een portfolio met intelligente zakelijke applicaties die superieure operationele efficiëntie koppelt aan baanbrekende klantervaringen – waardoor jij meer agile te werk kunt gaan en de complexiteit kunt verminderen zonder stijgende kosten. Wat Itineris onderscheid van de competitie is dat ze bovenop dit pakket van Microsoft nog een eigen aanvullende software laag hebben gebouwd.

Itineris is heel veel bezig met customer service om de klanten zo goed mogelijk verder te helpen en ze doen dit op verschillende manier zoals bv: telefonische gesprekken, e-mails, chats en dergelijke. Ze doen dit al op een hoog niveau maar ze blijven naar verbetering zoeken. Dit is dan ook de reden dat ze nu naar manieren kijken om artificiële intelligentie te gebruiken om deze processen te vergemakkelijken en efficiënter te laten verlopen. Zo zouden ze bijvoorbeeld een chatbot willen die vanzelf klanten kan helpen aan de hand van bestaande documentatie zonder dat de medewerkers van Itineris er zelf zouden tussen moeten komen. Maar ze kijken niet enkel naar de client-side, ze willen ook een co-pilot die de medewerkers kan helpen met het uitvoeren van simpele business opdrachten. Deze co-pilot zou dan moeten kunnen helpen met het verwerken van tickets of snel medewerkers moeten kunnen helpen als ze iets zouden moeten opzoeken.

Momenteel zijn ze dit al aan het bekijken hoe ze dit kunnen implementeren op andere vlakken binnen het bedrijf maar nog niet bij de customer service. Ze werken hiervoor met de co-pilot die Microsoft aanbiedt, Microsoft co-pilot 365, maar hiermee komen ze nog bij enkele dingen in de knoop. Sommige dingen kan het standaard pakket van de Microsoft co-pilot al maar de functionaliteiten die gebruik zouden maken van de extra laag die bovenop het MS Dynamics pakket is gebouwd werken nog niet. Dit zouden we dus verder onderzoeken hoe we deze co-pilot kunnen trainen en implementeren zodat deze op de correcte manier de relevante informatie zou kunnen voorleggen.

Op basis hiervan heb ik de volgende onderzoeksvraag geformuleerd: *onderzoeksvraag: Hoe kunnen we de efficiënte ontwikkeling van een co-pilot bevorderen door het identificeren van verschillende valkuilen en trainingsmethoden met specifieke aandacht voor de vereisten van de onderneming?*

Om op deze onderzoeksvraag een kwalitatief antwoord te kunnen geven moeten we uiteraard op een gestructureerde manier onderzoek doen in de relevante onderwerpen. De beste manier om dit te voltooien is van op voorhand de nodige doelstellingen op te stellen in overleg met het bedrijf en een de hand daarvan aan de slag te gaan.

Ik heb voor de casus van Itineris de volgende doelstellingen opgesteld:

* Identificeren van de belangrijkste vereisten voor de co-pilot
  + Wat moet de co-pilot kunnen?
  + Wat zijn de criteria om te beslissen of een handeling succesvol is?
* Identificeren van de verschillende valkuilen die we kunnen tegenkomen bij het ontwikkelen van de co-pilot.
  + Wat zijn de gekende valkuilen bij het ontwikkelen van een co-pilot?
  + Hoe kunnen we deze valkuilen het beste omzeilen?
* Trainen van de co-pilot.
  + Hoe train je een co-pilot?
  + Waar zit de informatie die de co-pilot moet verwerken?
  + Hoe gaat de co-pilot trainen met deze informatie?
* Ontwikkelen van een prototype van een co-pilot die voldoet aan de criteria van punt 1.

# Stand van zaken

Het onderzoeksdomein van artificiële intelligentie is al een stuk langer aanwezig dan wat de meeste mensen denken. Volgens een artikel van The Council of Europe is het nog steeds een relatief “jong” onderzoeksdomein van 60 jaar oud terwijl de meeste mensen er maar in contact mee zijn gekomen de laatste 10 jaren. Het onderzoek naar artificiële intelligentie, wat een set is van wetenschappen, theorieën en technieken (inclusief wiskundige logica, statistiek, waarschijnlijkheden, computationele neurobiologie, informatica), richt zich op het imiteren van cognitieve vaardigheden van de mens. Geïnstantieerd in de tweede wereldoorlog, is de ontwikkeling sterk gelinkt aan die van computers en heeft geleid naar de mogelijk om computers steeds ingewikkeldere taken te laten uitvoeren, wat we hiervoor enkel aan mensen konden overlaten.

Desondanks het nog steeds ver van menselijke intelligentie staat in de strikte vorm is artificiële intelligentie toch al sterk ontwikkeld. Het ultieme doel is van een “sterke” artificiële intelligentie te ontwikkelen die sterk verschillende gespecialiseerde problemen op een autonome manier kan oplossen. Momenteel zitten we hier nog helemaal niet aan aangezien de artificiële intelligentie die nu op de markt te vinden is enkel nog maar in staat is van efficiënt te helpen in de specifieke gebieden waarbinnen ze getraind zijn. Sinds 2010 heeft de discipline een sterke verbetering ondervonden door de grote verbetering in kracht van computers en toegang tot grote hoeveelheden data.

## Machine learning

Machine learning is een vorm van artificiële intelligentie dat gericht is op het bouwen van systemen die van verwerkte data kunnen leren of data gebruiken om prestaties te verbeteren. De termen artificiële intelligentie en machine learning worden vaak door elkaar gebruikt terwijl dit twee verschillende dingen zijn. Artificiële intelligentie is de overkoepelende term waaronder machine learning valt.

Er bestaan verschillende soorten machine learning algoritmes waarvan de meest prominente supervised machine learning , unsupervised machine learning, semi-supervised machine learning en reinforcement machine learning zijn.

### Supervised machine learning algoritmes

Supervise machine learning algoritmes zijn afhankelijk van correct gelabelde data. Het algoritme wordt de data gegeven tezamen met de verwachte uitkomst van de data. Het model leert dan ui de relaties van de input en output data om een model te bouwen. Het model mapt de input data naar de gewenste output en wordt getraind tot het model het gewenste niveau van accuraatheid bekomt. Dit wordt altijd getraind onder de supervisie van een developer.

Supervised machine learning wordt gebruikt voor:

* Het classificeren van nieuwe data in bestaande groepen en categorieën
* Voorspellen van ontwikkelende en toekomstige trends gebaseerd op voorspellende modellen
* Bedrijven helpen in campaign en acquisitie projecten door het voorspellen van veranderingen van variabelen

### Unsupervised machine learning algoritmes

Unsupervised machine learning algoritmes worden niet gecontroleerd door een developer en worden getraind op datasets zonder labels. Unsupervised machine learning algoritmes worden gebruikt voor het herkennen van patronen, trends of groepen van datasets waarvan de elementen onbekend zijn. Deze soort machine learning kan gebruikt worden om relaties te leggen tussen verschillende data punten en het segmenteren van gelijkaardige data.

Unsupervised machine learning algoritmes worden gebruikt voor:

* Het segmenteren van data voor gepersonaliseerde marketing campaigns
* In het licht zetten van onbekend trends bij de klanten
* Gepersonaliseerde content presenteren gebaseerd op interesses van online muziek en streaming services.

### Semi-supervised machine learning algoritmes

Zoals de naam suggereert is dit een mengeling van supervised en unsupervised machine learning algoritmes. Het wordt gebruikt voor datasets waarbij slechts een deel van de data correct is gelabeld. Semi-supervised machine learning algoritmes worden getraind op de subset van correct galebelde data. Het model gebruikt dan deze training om de rest van de data te labelen.

Semi-supervised machine learning algoritmes worden gebruikt om:

* Groeperen van grote tekstdocumenten zoals gescande boeken of documenten.
* Categoriseren van data dat tijdsintensief zou zijn voor een menselijke specialist.

### Reinforcement machine learning

Reinforcement machine learning laat een systeem toe van te leren en verbeteren van een functie door trial and error. Het model start vanaf nul en zal de beste oplossing proberen vinden in een specifieke omgeving door te leren van de acties die het in vorige fases heeft genomen. Dit model zit dus eigenlijk in een feedback loop waarbij elke actie ofwel wordt beloond ofwel afgestraft. Deze iteraties worden complexer als de complexiteit van de omgeving en acties die het model kan ondernemen toenemen. Reinforcement machine learning algoritmes worden vooral gebruikt als een actie te complex is voor een statische functie die een developer heeft geschreven.

Reinforcement machine learning wordt gebruikt bij:

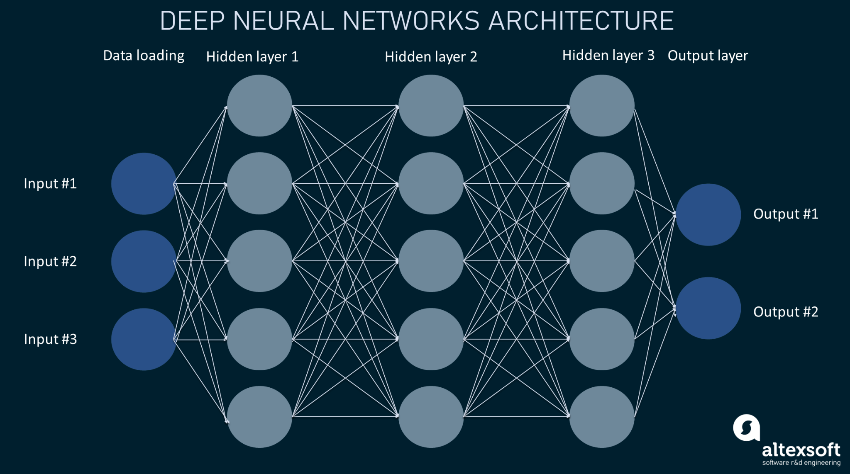
* Ontwikkeling van artificiële intelligentie
* Systemen in zelfrijdende auto’s
* Een model een spel te leren zoals schaken tegen een menselijke tegenstander

## Deep learning

Deep learning is een subset van machine learning, waarbij kunstmatige neurale netwerken (algoritmen die zijn ontworpen om te werken zoals het menselijke brein) leren van grote hoeveelheden gegevens. Er wordt van deep learning gesproken als een neuraal netwerk bestaat uit meer dan 3 lagen.

## Neurale netwerken

Een neuraal netwerk is een machine learning programma, of model, dat beslissingen maakt op een manier die vergelijkbaar is met het menselijke brein. Door het nabootsen van processen die de biologische neuronen gebruiken om samen te werken voor dingen te identificeren en de opties af te wegen komen deze netwerken aan conclusies.

Elk neuraal netwerk bestaat uit lagen van nodes, of artificiële neuronen – een input laag, een of meerdere verborgen lagen en een uitput laag. Elke node staat in connectie met andere nodes en heeft zijn eigen “gewicht” en “drempelwaarde”. Als de output van een node boven de gespecifieerde drempelwaarde is, dan is de node geactiveerd, stuurt de node data door naar de volgende laag van het netwerk.

Eens de input laag is vastgesteld wordt er voor elke node een gewicht vastgesteld. Dit gewicht bepaalt het belang van de node, waarbij de nodes met een groter belang meer invloed hebben op het eindresultaat in vergelijking met de nodes met een kleiner gewicht. Voor elke node wordt de input door een activatie-functie gestuurd om de output waarde te berekenen.

Indien deze output waarde groter is dan de drempelwaarde wordt data doorgestuurd naar de volgende laag in het neurale netwerk (bijvoorbeeld van input node #1 naar alle nodes in de verborgen laag 1 die in verbinding staan met input #1). Zo wordt de output van de ene node de input van de andere.

Terwijl we het model trainen zullen we de accuraatheid van het model willen evalueren en dit doen we aan de hand van een kost functie. Dit wordt ook vaak de Mean Squared Error (MSE) genoemd. Het doel is van deze kost functie zo klein mogelijk te krijgen om de correctheid van elke observatie te verzekeren. Het model past het gewicht en de drempelwaardes aan door verschillende iteraties aan de hand van de kost functie en reinforcement learning om zo tot een punt te komen dat we het lokale minimum kunnen noemen.

## Large Language Models

Een Large Language Model (LLM) is een deep learning algoritme dat een reeks natuurlijke taalverwerkingstaken kan uitvoeren. De modellen maken gebruik van transformator modellen en zijn getraind op grote datasets. Zo kunnen ze teksten of andere content herkennen, vertalen, voorspellen of genereren. Naast het aanleren van menselijke talen aan toepassingen van artificiële intelligentie, kunnen large language models ook getraind worden om verschillende taken uit te voeren, zoals het begrijpen van eiwitstructuren, het schrijven van softwarecode en nog veel meer.

### Wat is een transformatormodel?

Een transformatormodel is de meest voorkomende architectuur van een groot taalmodel. Deze bestaat uit twee kerncomponenten een encoder en een decoder. Een transformatormodel verwerkt gegevens door invoer in tokens om te zetten en vervolgens gelijktijdige wiskundige vergelijkingen uit te voeren om de relaties tussen de tokens te ontdekken. De computer kan dus de patronen zien die een mens zou vaststellen als dezelfde vraag aan hem zou worden voorgelegd.

Transformer-modellen vertrouwen op zelfaandachtsmechanismen, waardoor ze sneller kunnen leren dan traditionele modellen, zoals modellen voor het lange kortetermijngeheugen (LSTM). Een zelfaandachtig transformatormodel kan naar verschillende delen van een reeks, of de gehele context van een zin, kijken om voorspellingen te genereren.

#### Zelfaandachtsmechanisme

De kern van het transformer-model ligt in het zelfaandachtsmechanisme. Hierdoor kan het model het belang van een token in een reeks afwegen op basis van de relevantie ervan voor andere tokens. Hiermee kan het model aandachtscores berekenen en op basis hiervan zal het model hogere gewichten toekennen aan tokens die semantisch gerelateerd zijn (een gelijkaardige/ gelinkte betekenis hebben).

#### Encoder

De encoder is verantwoordelijk voor het verwerken van de invoerreeks en de correcte representaties eruit te halen. Het bestaat uit een stapel identieke lagen, elke laag bestaat uit een zelfaandachtsmechanisme gevolgd door een feed-forward neuraal netwerk. Het zelfaandachtsmechanisme laat het model concentreren op verschillende delen van de invoerreeks en het feed-forward neuraal netwerk verwerkt de informatie lokaal. Door deze samenwerking worden niet-lineaire transformaties mogelijk gemaakt.

#### Decoder

De decoder accepteert de gecodeerde representaties van de encoder en genereert een uitvoerreeks. Het bevat ook zelfaandachtsmechanismen om afhankelijkheden binnen de uitvoerreeks vast te leggen en aandacht te besteden aan de relevante delen van de invoerreeks. Hierdoor kan het model contextueel bewuste uitput genereren voor taken zoals automatische vertaling of samenvatting van teksten.

## Natural Language Processing (NLP)

Natural Language Processing is een algoritme dat probeert om tekst of spraak data te verstaan. NLP combineert bestaande taalregels met statistische, machine learning modellen. Als deze modellen samenwerken laat dit een computer tekst of spraak data behandelen en hier dan een response op genereren.

## Chatbots

Om het simpel uit te leggen is een chatbot een programma dat menselijke conversaties simuleert en behandelt, dit laat mensen toe van te communiceren met digitale apparaten alsof ze met een echte persoon waren aan het communiceren. Dit kunnen zeer simpele programma’s zijn die slechts een simpele query beantwoorden met een antwoord bestaande uit één lijn tekst, of dit kunnen complexe digitale assistenten zijn die leren en evolueren om steeds betere antwoorden en resultaten op te leveren.

Er zijn twee types chatbots die voornamelijk gebruikt worden namelijk:

* Task-oriented (declarative) chatbots
* Data-driven and predictive (conversational) chatbots

### Task-oriented (declarative) chatbots

Task-oriented (declarative) chatbots of taak-georienteerde (declaratieve) chatbots zijn programmas die focussen op het uitvoeren van maar een enkele functie. Door gebruik te maken van regels, NLP en heel weinig machine learning genereren ze geautomatiseerde maar conversationele antwoorden op vragen van gebruikers. Interacties met deze chatbots zijn zeer specifiek en gestructureerd, denk aan FAQs. Deze chatbots kunnen de meest normale vragen afhandelen zoals bijvoorbeeld de openingsuren van het bedrijf of de locatie van de vestiging. Deze chatbots zijn momenteel het meest voorkomende.

### Data-driven and predictive (conversational) chatbots

Data-driven and predictive (conversational) of data-gedreven en voorspellende (conversationele) chatbots worden vaak vergeleken met virtuele assistenten. Deze chatbots zijn veel interactiever, gepersonalizeerd en zijn in staat van een gesofisticeerd antwoord te geven op vragen. Deze chatbots zijn zich bewust van context en gebruiken natural language understanding (NLU), NLP en machine learning om bij te leren. Ze gebruiken voorspellende intelligenite en statistieken van voorlopende interacties om personalisatie te kunnen toepassen. Voorbeelden hiervan zijn Siri van Apple en Alexa van Amazon, deze chatbots zijn in staat van de voorkeuren van gebruikers te leren, gepersonaliseerde voorstellen te geven en zelf te voorspellen dat de gebruiker iets nodig zou hebben.

## Co-pilots

Een AI co-pilot is een interface die gebruik maakt van Large Language Models (LLMs) om gebruikers te ondersteunen in verschillende handelingen en beslissende processen. Door gebruik te maken van de LLMs kunnen AI co-pilots grote hoeveelheden data begrijpen, analyseren en behandelen.

Ai co-pilots spelen een grote rol in het verbeteren van productiviteit en efficiëntie van handelingen door:

* Het bieden van context gevoelige assistentie:
  + Ai co-pilots kunnen proactief reageren op de handelingen van de gebruiker aan de hand van een bepaalde context.
  + Het verzekerd relevantie en snelle reactietijden voor de nodige processen.
* Automatiseren van alledaagse taken:
  + Door repetitieve en tijds consumerende taken in handen te nemen laat de co-pilot toe van de gebruikers hun tijd te steken in strategische en creatieve taken dat de productiviteit doet stijgen.
* Analyseren van data:
  + AI co-pilots kunnen snel grote hoeveelheden data verwerken, patronen en trends herkennen, en correcte inzichten op deze data weergeven voor beslissingen te maken.
* Verbeteren van communicatie:
  + AI-co-piloten faciliteren effectieve interacties met verschillende belanghebbenden, waaronder werknemers, klanten en leveranciers, waardoor communicatieprocessen worden gestroomlijnd en vertragingen of misverstanden worden verminderd.
* Ongelijksoortige systemen te laten samenwerken:
  + Co-pilots kunnen de “tussenpersoon” zijn voor verschillende platformen, tools of software te laten samenwerken onder een geheel met de zekerheid van het behoud van data, toegang en compatibiliteit over het geheel.

Samengevat staat een een co-pilot in om handelingen te versimpelen en geldige begeleiding te bieden, met als doel de user experience te verbeteren en de business goals effectief en efficiënt te voltooien.

## Microsoft co-pilot

De co-pilot die Itineris zou willen implementeren is de co-pilot die Microsoft in 2023 heeft uitgebracht namelijk de Microsoft co-pilot. Het is gebaseerd op de Large Language Models en is daardoor in staat van verschillende dingen uit te voeren zoals bronnen citeren, liedjes schrijven, etc.. .

Het Maakt gebruik van het Microsoft Prometheus model, gebouwd bovenop GPT-4 van OpenAI. Dit is een fundamenteel Large Language Model dat is gefinetuned aan de hand van supervised en reinforcement learning technieken. De chatbot is dan ook in staat van te communiceren in verschillende talen en dialecten zoals ChatGPT van OpenAI.

### Microsoft co-pilot with Graph-grounded chat (M365 chat)

Microsoft co-pilot with Graph-grounded chat is een chatbot die vergelijkbaar is met de beter bekende chatGPT van OpenAi. De co-pilot combineert de kracht van artificiële intelligentie met uw werk data en applicaties om te helpen met creativiteit, productiviteit en uw andere vaardigheden.

Net zoals de andere AI chat apps kan Microsoft co-pilot informatie vinden op het internet en gedichten schrijven, maar het kan ook werk gerelateerde data zoals mails, chats en bestanden incorporeren om werk gerelateerde vragen te beantwoorden en verwerken.

Je kan de Microsoft co-pilot with Graph-Grounded chat op verschillende manieren aanhalen maar de meest prominente en meest aangeraden methode is via de desktop en mobiele versie van microsoft teams. De chat verschilt van de andere co-pilots in microsoft 365 apps doordat het een tool is die over verschillende applicaties werkt, om de kracht van artificiële intelligentie te presenteren in samenwerking met uw beveiligde werk data.

De meest voorkomende use cases van de M365 chat zijn:

* Een beeld te scheppen van de huidige situaties. De co-pilot kan opsommingen maken van grote hoeveelheden data in simpele samenvattingen.
* Creëren van content en brainstormen. Het kan helpen voor over nieuwe ideeën te brainstormen en nieuwe content te schetsen gebaseerd op verschillende bronnen.
* Snelle antwoorden verkrijgen. Microsoft co-pilot staat in als je eigen persoonlijke search engine. Je kan vragen stellen over specifieke bestanden en berichten of je kan informatie opvragen die je zelf niet meer terugvind.

### Microsoft co-pilot in customer service

Co-pilot in customer service voorziet real-time assistentie voor medewerkers tijds-consumerende te automatiseren en tickets efficiënter af te handelen om klanten beter te kunnen helpen. Als je de co-pilot features in de customer service workspace app activeert kun je de volgende acties door de co-pilot laten uitvoeren:

1. Antwoorden op vragen
2. Een email opstellen
3. Schetsen opstellen voor chat antwoorden
4. Samenvatten van een case/ticket en conversatie

## Microsoft Dynamics 365 Customer Service

Microsoft Dynamics 365 Customer Service is een business applicatie dat teams toelaat van snelle, consistente en kwalitatieve customer service te leveren en inzichten te verkrijgen over customer trends en interacties. Dynamics 365 Customer Service staat in voor de customer service en het ondersteunende operaties, dit doet het aan de hand van verschillende tools zoals multi-channel engagement en tracking over verschillende platformen en apparaten.

## Wat wordt er verwacht van de co-pilot?

In de eerste instantie is het de bedoeling dat de co-pilot de work-load verlicht en de productiviteit doet stijgen door de mensen meer tijd te geven voor de meer creatieve en tijdsintensieve handelingen. Of dat we dit effect dan ook effectief bekomen zullen we waarnemen aan de hand van enkele situaties of handelingen die we de co-pilot zouden laten doorlopen. De resultaten van deze handelingen vergelijken we dan met criteria die we op voorhand zullen opstellen om waar te nemen of een handeling succesvol is afgehandeld.

### Wat moet de co-pilot kunnen?

Er zijn verschillende handelingen die de co-pilot zal moeten kunnen uitvoeren dus we zullen de voornaamste proberen opstellen. Eerst en vooral zal de co-pilot moeten kunnen samenwerken met de data, eens dit gebeurd is gaat het voornaamste zijn van de data op een goede, context gevoelige manier te kunnen representeren. De eerste situatie zou dus zijn dat we een vraag moeten kunnen stellen aan de co-pilot via een chatbot en dat deze een correct antwoord moet kunnen geven aan de hand van de bestaande documentatie. Een tweede situatie zou zijn dat een werknemer een ticket opent van een klant en dat de co-pilot de case history van de klant voorlegt. Dit betekent dus dat de co-pilot ook aan de informatie van de klanten zou moeten kunnen en voor een specifieke klant de nodige dingen op een presentabele manier kan weergeven. Een laatste situatie zou zijn dat een medewerker een ticket van een klant heeft openstaan en dat de medewerker het ticket wilt behandelen. De medewerker kiest ervoor van een mail te versturen met de nodige info voor de bestaande context en de co-pilot zal hierbij helpen. Dit kan door de mail volledig of deels op te stellen met de bevestiging van de medewerker om het te versturen. De mail zal de relevante informatie bevatten in mensentaal die volgens de co-pilot nodig zijn om de context van het ticket op te lossen.

De korte opsomming van de handelingen zijn dus:

1. Er kan een vraag gesteld worden aan een chatbot waarbij je een correct en context-gevoelig antwoord krijgt aan de hand van de bestaande documentatie.
2. Indien een medewerker een ticket opent voor een situatie van een klant te behandelen, wordt de relevante informatie voor dit ticket op een presentabele manier weergegeven.
3. Een medewerker verkiest ervoor een ticket af te handelen aan de hand van een mail naar de klant te versturen. De co-pilot helpt bij het opstellen van de mail door het presenteren in mensentaal van de informatie die relevant is voor de context van het ticket.

### Wat zijn de criteria om te beslissen of een handeling succesvol is?

Om te weten of een handeling effectief goed is afgehandeld moeten we voor elk scenario bekijken wat de belangrijkste punten zijn die we willen testen. In grote lijnen kan je meestal een ja of nee antwoord geven om te zeggen of een handeling goed is afgelopen, maar soms kan het goed afgelopen zijn maar niet succesvol doordat het bijvoorbeeld te traag of maar een half antwoord gaf. We zullen dus voor de scenario’s die we hebben opgesteld de nodige criteria moeten vaststellen en een basis waaraan we deze criteria gaan evalueren.

Voor elk scenario zijn er bepaalde dingen die we altijd als een deel van het proces kunnen beschouwen, namelijk:

* De Initiële toestand
* De doel toestand
* Bepaalde acties gedefinieerd in een domein

De initiële toestand is belangrijk om altijd correct te definiëren vanuit wat we beginnen te meten. Om verschillende metingen correct op elkaar te laten afstemmen zal deze dus altijd hetzelfde moeten blijven.

De doel toestand zou over de verschillende testen ook hetzelfde moeten zijn. De manier waarop deze toestand is bekomen kan verschillen waardoor we dus zeker naar de genomen acties moeten kijken voor de beste acties te bekomen. Hierbij wordt er dus telkens compleet gedefinieerd wat er bekomen moet worden.

De bepaalde acties die gedefinieerd zijn in een domein verschillen van scenario tot scenario, zo kan het ene scenario stappen zoals ’10 meter naar voor’ hebben en het andere ‘schrijf: wat zijn de belangrijkste punten in mijn inbox’. Deze acties moeten ook goed gedefinieerd zijn om sterk verschillende testen te kunnen uitwerken.