# 1. Bedrijf & Probleemdomein: 200-250 woorden.

Het bedrijf Great Outdoors is een groothandel op het gebied van kampeer- en wandelartikelen. Dit betekent dat zij op grote schaal artikelen inkopen die gebruikt kunnen worden bij kamperen en wandelen. Zij verkopen deze artikelen in grote kwantiteiten door aan kleinere winkels die deze spullen weer direct verkopen aan consumenten. Great Outdoors haalt hun winst voornamelijk uit kleine winstmarges per product op grote schaal.

Binnen de afdeling data-analyse op het hoofdkantoor is Amsterdam houdt men zich voornamelijk bezig het verzamelen en analyseren van verscheidene datapunten binnen en bedrijf. En hoe deze informatie kan helpen bij het maximaliseren van de winst.

Bij de strategische besluitvorming binnen het bedrijf met betrekking op de inkoop/verkoop en het assortiment van artikelen speelt data een steeds meer leidende rol. Het is namelijk belangrijk om aan de hand van keiharde feiten en cijfers beter geïnformeerde keuzes te kunnen maken. Denk hierbij aan keuzes zoals het toevoegen of verwijderen van aan artikel uit het assortiment. Of een product eerder in het seizoen beschikbaar stellen. Deze keuzes kunnen enorme impact hebben op de winst die binnengehaald wordt en daarom is het essentieel dat ze goed geïnformeerd gemaakt worden.

# 2. Aanleiding & Probleemstelling: 200-250 woorden.

Binnen Great Outdoors wordt de data op grote schaal verzameld en beheerd, maar er ontbreekt essentieel inzicht. Dit gebrek aan inzicht belemmert niet alleen de dagelijkse operaties maar maakt het ook lastig strategische beslissingen te nemen. Het succes van Great Outdoors wordt bemoeilijkt doordat de invloed van maatschappelijke ontwikkelingen op de bedrijfsprestaties niet duidelijk is.

Momenteel is het voor menig medewerker niet mogelijk om effectief gebruik te maken van de beschikbare data. Dit omdat er wel data beschikbaar is, maar geen interactieve manier is om deze data te kunnen bekijken/filteren en relativeren.

Er is ook een kans om de analyses te verbeteren, inclusief het voorspellen van toekomstige data. Dit geeft het management een helder beeld voor strategische besluitvorming. Het benutten van deze kans zal niet alleen de afdeling, maar het hele bedrijf doeltreffender maken, met verbeterde besluitvorming en impact op bedrijfsresultaten en klanttevredenheid. In het licht van toenemende concurrentie en de behoefte aan snelle, accurate reacties op veranderingen, is het cruciaal om deze kans nu te grijpen.

# 3. Doelstelling: 200-250 woorden.

Great Outdoors streeft ernaar om haar besluitvorming te verbeteren door meer gebruik te maken van data. Dit willen ze realiseren door concrete inzichten te verkrijgen uit een dashboard, waarin grafieken en kaarten op verschillende detailniveaus worden gepresenteerd. Dit doel is waardevol omdat het de mogelijkheid biedt om gefundeerde beslissingen te nemen op basis van feitelijke gegevens en analyses, wat uiteindelijk kan leiden tot een efficiënter beheer en optimalisatie van bedrijfsprocessen.

Naast het raadplegen van het dashboard met historische en actuele gegevens, kan het systeem ook toekomstige data voorspellen. Door gebruik te maken van geavanceerde voorspellende modellering, zoals machine learning-algoritmen, kan het direct de verwachte invloed op winstcijfers voorspellen voor elk potentieel strategisch besluit. Deze benadering biedt een proactieve en inzichtelijke manier om strategische beslissingen te ondersteunen, waardoor het bedrijf beter is uitgerust om weloverwogen keuzes te maken met het oog op toekomstige resultaten.

# 4. Concrete werkzaamheden: 200-250 woorden.

Om het project succesvol te initiëren, beginnen we met een uitgebreid onderzoek binnen de afdeling om specifieke data-analysebehoeften te identificeren. Deze directe interactie met diverse afdelingen is bedoeld om een alomvattend begrip te krijgen van de vereisten en verwachtingen.

Parallel aan deze navraag voeren we een grondig onderzoek uit naar bestaande software- en infrastructuurnormen en -eisen binnen het bedrijf. We focussen op cruciale aspecten zoals gebruikte programmeertalen, frameworks, het potentieel van uitbreidbare applicaties, en de voorkeur voor web- of desktopapplicaties. Beveiligingseisen vormen eveneens een integraal onderdeel van dit onderzoek, waarbij specifieke aandacht wordt besteed aan de nodige veiligheidsmaatregelen.

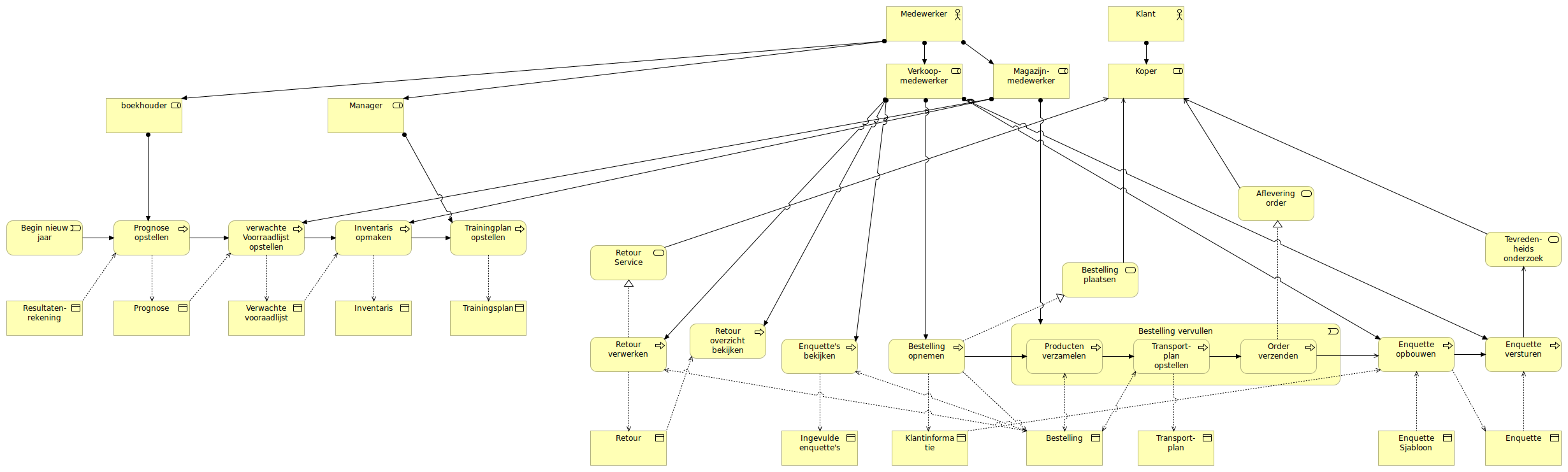
Na deze initiële verkenning gaan we dieper in op de integratiemogelijkheden om de nieuwe data-analysefunctionaliteiten te koppelen aan bestaande databronnen. Dit omvat een grondige analyse van hoe de nieuwe oplossing goed kan worden geïntegreerd, met speciale aandacht voor mogelijke uitdagingen en kansen.

Met alle verzamelde inzichten en een duidelijk beeld van de software stack en eisen, maken we tijdens de implementatiefase gebruik van de scrum agile-methodologie. Deze aanpak zorgt voor een iteratief proces waarin continue feedback mogelijk is, en waar veranderingen kunnen worden aangevraagd tijdens de ontwikkeling. Dit stelt ons in staat om flexibel in te spelen op evoluerende behoeften en optimaliseert het succes van het project.

# 5. Archimate Business layer

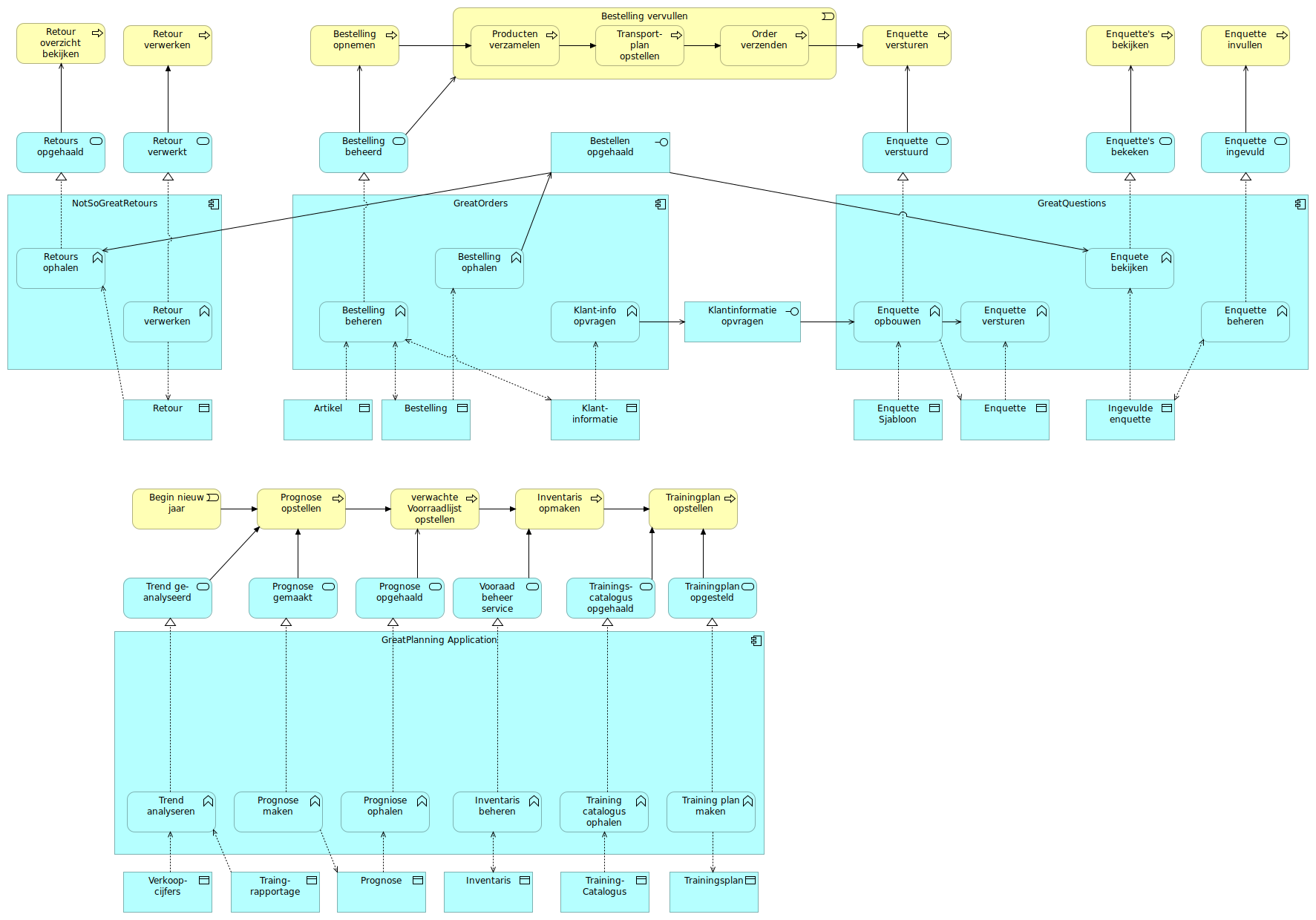
In het verkoopdeel van onze bedrijfsactiviteiten kunnen klanten diverse services benutten. Kopers hebben de mogelijkheid om bestellingen te plaatsen, tevredenheidsonderzoeken uit te voeren, retourservices aan te vragen en de aflevering van hun orders te regelen. Deze services zijn specifiek gericht op de behoeften van kopers, en de belangrijkste stakeholders hierbij zijn de magazijnmedewerker, de koper zelf en de verkoopmedewerker. Het proces dat volgt na het plaatsen van een bestelling omvat het vervullen van de order, waarbij activiteiten zoals het verzamelen van producten, het opstellen van een transportplan en het verzenden van de order worden uitgevoerd.

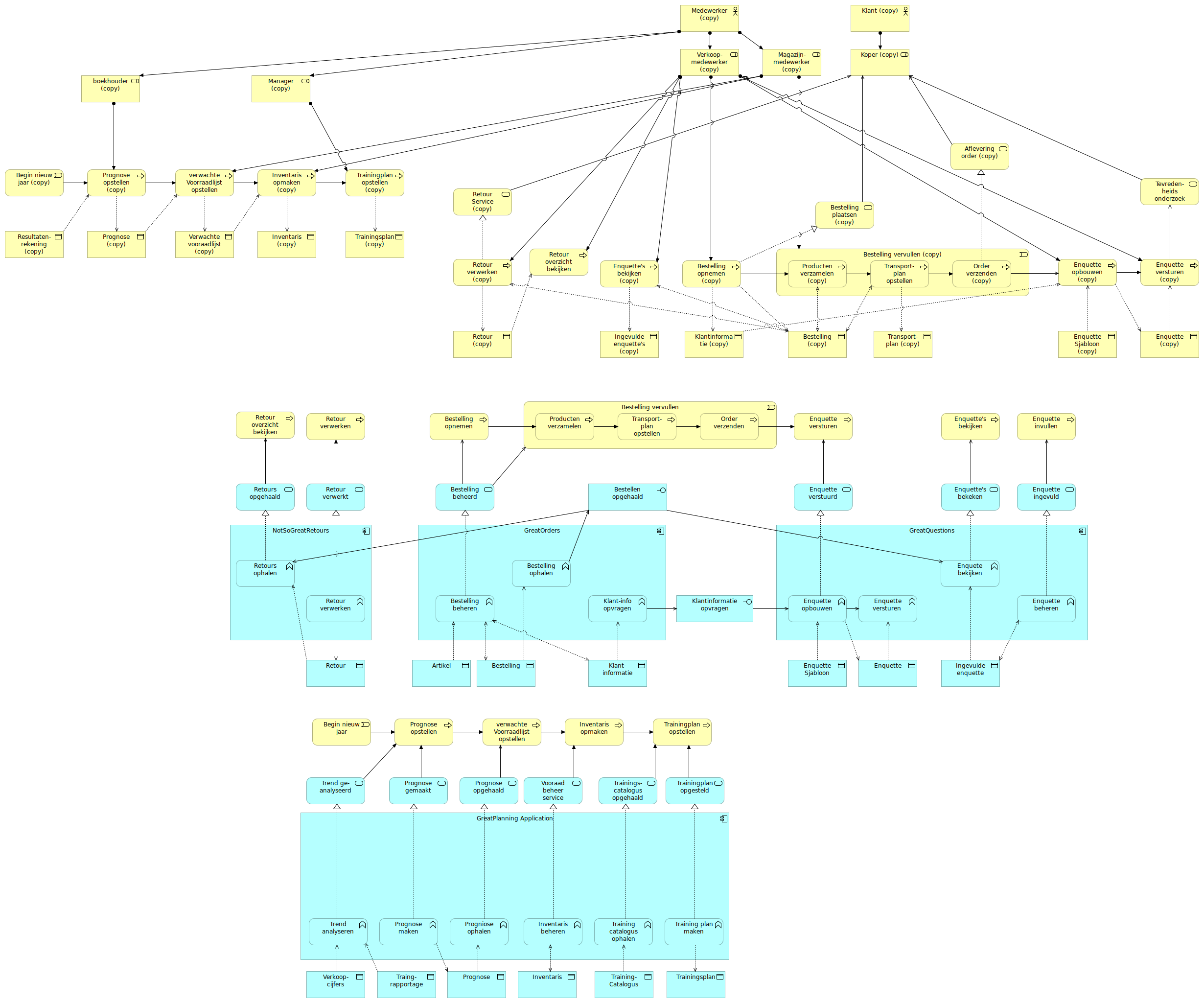
Aan de planningszijde van onze bedrijfsopzet staan verschillende services tot de beschikking van stakeholders zoals de boekhouder, manager en magazijnmedewerker. De boekhouder kan prognoses opstellen, de manager kan training plannen opzetten, terwijl de magazijnmedewerker verantwoordelijk is voor het opstellen van verwachte voorraadlijsten en het maken van inventarissen.



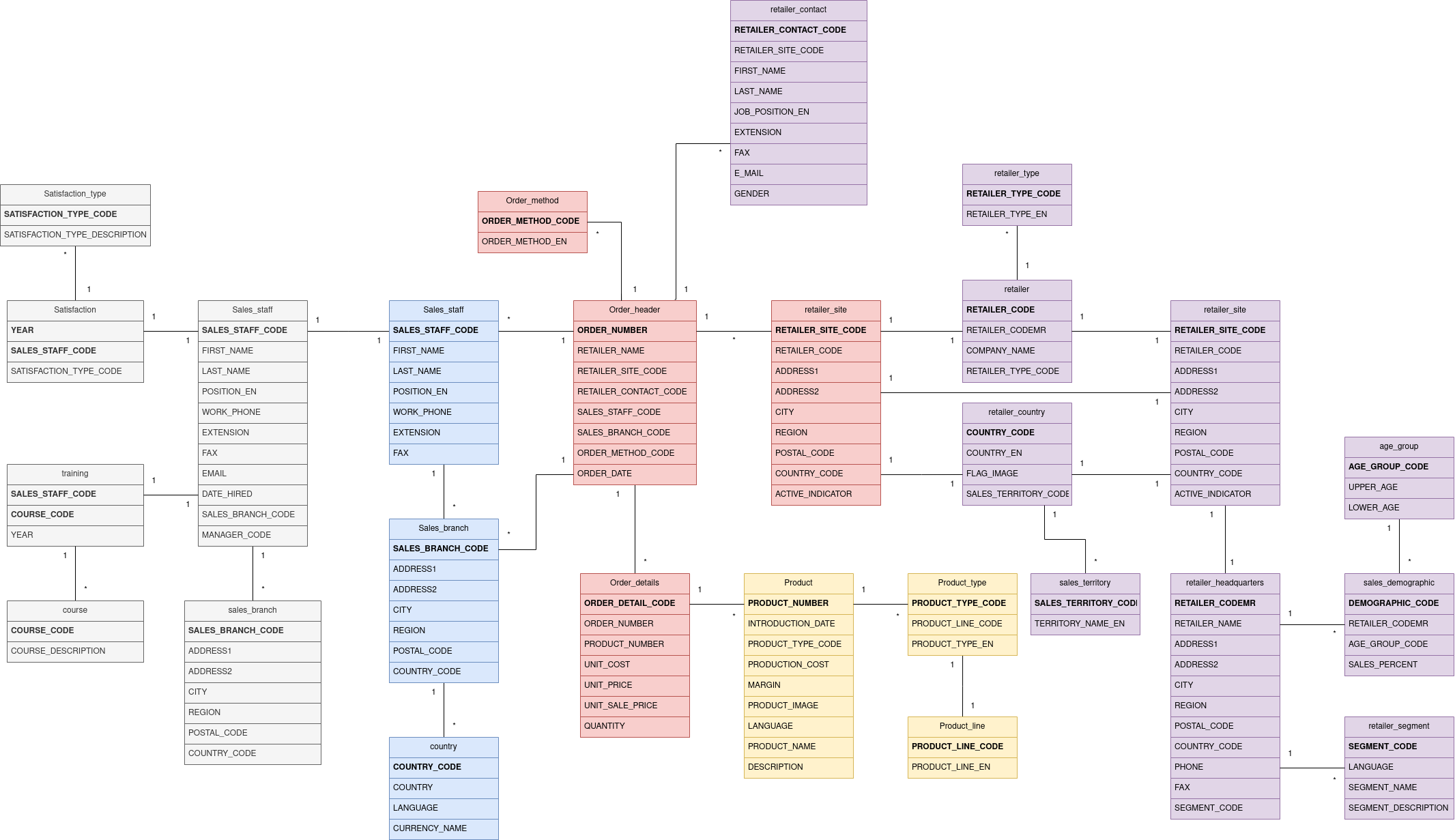
# 6. Archimate Application layer

Onze Application Layer is opgedeeld in diverse applicatiecategorieën. "NoSoGreatRetours" richt zich op retourzendingen van klanten, met services zoals Retours ophalen en Retour verwerken, beide afhankelijk van de business services. "GreatOrders" behandelt orderbeheer en -ophalen, gericht op het verwerken van uitgaande bestellingen. "GreatPlanning" omvat services zoals Trend-Analyse, Prognose, Voorraadbeheer en Training beheer, gebaseerd op financiële resultaten en trends, waardoor effectieve verkoopdoelen en trainingsplanning mogelijk zijn. "GreatQuestions" faciliteert klanttevredenheid door middel van enquêtes versturen, bekijken en invullen. Deze applicatiecategorieën spelen elk een specifieke rol in het stroomlijnen van bedrijfsprocessen en het verbeteren van de klantbeleving.

7. Combinatie Business en Applicatie layer



# 8. Brongegevensmodel

In de kern van ons brongegevensmodel staat het verkoopproces als het centrale feit. De essentiële dimensies die dit proces omvatten zijn artikel, Retail (klant) en salesperson. Elk van deze dimensies draagt bij aan een holistisch begrip van het verkoopproces (Orderheader & Orderdetail tabellen), waarbij de interacties tussen artikelen, klanten en verkoopmedewerkers in kaart worden gebracht. Deze dimensies vormen de bouwstenen waarmee we later de sub-feiten en sub-dimensies gaan definiëren.

**Indien de foto's moeilijk te zien zijn, download ze direct van onze OneDrive** [**hier**](https://dehaagsehogeschool-my.sharepoint.com/:f:/g/personal/22111131_student_hhs_nl/Epl1fSnPNVRHnhpO8H_MEJABRYlVeZIzmPE6INYpf-d5gQ?e=0KCemZ)**.**

# 9. Datastructuren in Python

Voor de opdracht 2 uit week 3, moeten we drie datastructuren in python gebruiken en onderzoeken. Wij hebben de volgende structuren gekozen:

* Dictionary
* List
* Deque

We hebben voorbeelden van code voor een algoritme dat gebruikmaakt van een specifieke datastructuur. Deze zijn te vinden onder kopje: **3.2/examples** in de github.

## Big Oh

Big O-notatie is een manier om te praten over hoelang het duurt voor een computerprogramma om iets te doen als de hoeveelheid werk groter wordt. Het is een soort snelle samenvatting om te begrijpen hoe efficient een programma is wanneer het met steeds grotere hoeveelheden informatie werkt. Een lage Big O-notatie betekent dat het programma snel blijft, zelfs als de hoeveelheid werk toeneemt. Bijvoorbeeld, O(1) betekent dat het ongeveer even snel blijft, terwijl O(n) betekent dat het langzamer wordt naarmate er meer werk is.

## Dictionary

Dictonaries of anders genoemd associative arrays of maps zijn essentielen datastructuren binnen Computer Science. Met een dictionary kan je key-value pairs efficient opslaan en ophalen.

Het ophalen van data uit een dictionary heeft de volgende Big O notatie: O(1), oftewel constant time.

Een voorbeeld van een algoritme dat gebruik maakt van dictionaries is **Frequency Counting.**  
Bij een frequency counting algoritme wordt geteld hoe vaak verschillende elementen voorkomen. Zoals woorden in een document, of karacters in een zin.

Zo kan je bijvoorbeeld het meest voorkomende element uit een array bepalen, door per verschillend element in een dictionary op te slaan hoe vaak die voorkomt.

Bij dit voorbeeld komt een Big O notatie van O(n) kijken, dit omdat je wel elk element van de array moet doorlopen. Vervolgens bepaal je het meest voorkomende element aan de hand van de dictionary die je hebt opgebouwd.

## List

Het list-structuur wordt gebruikt in het algoritme dat de tekst omkeert. De procedure begint met het omzetten van de invoertekst naar een lijst van karakters, een operatie die lineaire tijd in beslag neemt, met een tijdscomplexiteit van O(n), waarbij n de lengte van de invoertekst is.

Vervolgens wordt een iteratieve aanpak toegepast met twee pointers in een while-lus om de lijst in-place om te keren. In het slechtste geval, waarbij n even is, moeten alle n/2 karakters worden omgewisseld. Hoewel technisch gezien O(n/2), wordt deze complexiteit vereenvoudigd tot O(n) omdat constante factoren in de Big-O-notatie worden weggelaten.

Tot slot wordt de omgekeerde lijst van karakters weer omgezet naar een string, wat opnieuw O(n) tijd in beslag neemt. In totaal blijft de algehele tijdscomplexiteit van het algoritme O(n), wat aangeeft dat de prestaties evenredig zijn aan de lengte van de invoertekst. Deze efficiëntie maakt het algoritme geschikt voor het omkeren van teksten, zelfs voor langere strings.

## Deque

De `deque` in Python is een afkorting voor "doubly-ended queue" en is een datastructuur die elementen aan beide uiteinden kan toevoegen of verwijderen met constante tijdcomplexiteit. De `deque` is geïmplementeerd als een double linked list, waardoor zowel het toevoegen als verwijderen van elementen aan het begin en het einde van de lijst efficiënt is.

Een bekend algoritme dat gebruikmaakt van de `deque` in Python is het "sliding window" algoritme. Dit algoritme wordt vaak gebruikt bij problemen waarbij je een venster van een bepaalde grootte over een reeks gegevens moet verplaatsen, zoals bijvoorbeeld bij het vinden van het maximale of minimale element in elk venster.

De tijdscomplexiteit van het sliding window algoritme met gebruik van de `deque` is O(n), waarbij n het aantal elementen in de invoerlijst is. Dit komt omdat elk element slechts één keer aan het begin en één keer aan het einde van de `deque` wordt toegevoegd of verwijderd, wat resulteert in een constante tijdcomplexiteit voor elke bewerking. Daarom wordt het algehele aantal bewerkingen beperkt tot O(n).