



Agentic AI voor het MKB

Toepassen van Agentic AI in het midden- en klein bedrijf

Wat is een Agent? Hoe bouw je een agent? Iedereen praat over AI, Agents. Grote bedrijven en Consultancy firma's laten medewerkers een agent bouwen. De cursus en het bouwen van een Agent is van belang om daadwerkelijk te doorleven wat een agent is, kan betekenen, hoe een medewerker een agent ervaart. Maar waar begint het grotere publiek, het MKB? Hoe kun je Agentic AI toepassen in het Midden- en Kleinbedrijf waar weinig tijd, geld, kennis aanwezig is. We beginnen bij de basis met een uitleg van het speelveld.

Agentic AI leunt volledig op een aantal elementen: transitie, structuur, processen, data en de technologie maakt het vervolgens mogelijk. Een agent bouwen is niet alleen technologie implementeren, maar ook het proces verbeteren. Op www.act2vision.com zijn tal van artikelen geschreven over gerelateerde onderwerpen. Op plaatsen in dit document zijn verwijzingen opgenomen naar de artikelen voor verdere verdieping op het gebied van Operations & Supply Chain management, Data Driven, Digitalisering, Projectmanagement, Processen, Operational Excellence.

Act2Vision | Amsterdam, Nederland 2026.

Operational Excellence, Building process clarity today for the value chain of tomorrow.

Inhoud

Inleiding.....	4
Eerste verkenning; hoe bouw je een agent en waar begint het MKB?.....	4
Wat doet een agent concreet?	5
Waarom is dit relevant voor het MKB?.....	5
Voorbeelden van haalbare MKB-agents	5
Waar moet het MKB niet mee beginnen?.....	6
Kansrijke MKB-sectoren	6
Welke medewerkerstaak kan AI veilig voorbereiden.....	7
De startvraag	7
De veilige Agent, duidelijke grenzen.....	7
Een agent heeft informatie nodig.....	7
Praktische aanpak.....	10
Minimale agent-architectuur voor het MKB	10
Bouwen in 6 stappen.....	10
Stap 1: kies één proces met pijn	11
Stap 2: beschrijf de huidige werkwijze	14
Stap 3: bepaal de agent-taak	15
Stap 4: verzamel de kennisbasis	17
Stap 5: ontwerp de controles.....	20
Stap 6: meet resultaat.....	23
Mijn eerste Agent.....	26
Casus Fulfilment Centers.	26
Processeselectie en validatie	26
Business Case en projectvoorbereiding	26
Waarom is dit proces interessant.....	27
Mogelijke eerste agent-use-case	28
Vorbereiding van de besluitvorming	29
Alternatieve VAL/VAS Agents	30
De bouw van de agent	31
Functioneel Ontwerp	31

Bouw van de Agent (Python).....	57
Python Introductie	57
Aanpak	57
MVP VAS Readiness Agent in Python	61
Werkt de Agent?.....	80
Bijlage 1. Template: Agent-charter	91
Bijlage 2. Template: gecontroleerde kennisbasis.....	95
Bijlage 3. Template: controlemodel voor een MKB-agent	104

Inleiding

Gaat een Agent mijn werk overnemen? Voor grote bedrijven is agentic AI vaak onderdeel van digital twins, digital brains en control towers. Voor het MKB begint het veel kleiner: een agent die één terugkerend probleem oplost, werkt met bestaande documenten en systemen, en de mens laat goedkeuren.

Het grotere publiek moet niet beginnen met de vraag “hoe bouw ik een agent?”, maar met “welke medewerkerstaak kan AI veilig voorbereiden?”.

De eerste volwassen MKB-toepassing is waarschijnlijk geen autonome digitale collega, maar een betrouwbare procesassistent die werk voorbereidt, afwijkingen signaleert en standaardtaken versnelt. Dat is praktisch, betaalbaar en leerzaam.

Eerste verkenning; hoe bouw je een agent en waar begint het MKB?

Een AI-agent is geen gewone chatbot. Een chatbot geeft vooral antwoord. Een agent krijgt een doel, verzamelt informatie, redeneert over de volgende stap, gebruikt tools en levert een uitkomst op. Denk aan: “controleer welke klanten te laat betalen, stel prioriteiten, maak conceptmails en zet een actielijst klaar.” De mens blijft voorlopig beslisser, maar de agent doet het voorbereidende werk.

Het speelveld bestaat grofweg uit vier niveaus.

Niveau 1 is de gewone AI-assistent. Die helpt met schrijven, samenvatten, analyseren en structureren. Dit is waar de meeste bedrijven nu beginnen.

Niveau 2 is de taakgerichte agent. Die voert één duidelijk proces uit, bijvoorbeeld offertes controleren, klantvragen classificeren, rapporten maken of voorraadverschillen analyseren.

Niveau 3 is de procesagent. Die kijkt naar een hele workflow, bijvoorbeeld order-to-cash, inkoop, planning, customer service of recruitment. De agent bewaakt stappen, signaleert uitzonderingen en adviseert acties.

Niveau 4 is de digitale operatie: control tower, digital brain of digital twin. Grote bedrijven bouwen dit voor supply chain, productie en logistiek.

In AbcSupplyChain-rapport (AbcSupplyChain, 2025) worden tal van toepassingen beschreven: digitale twins spiegelen de echte supply chain voor scenarioanalyse, een digital brain geeft voorspellingen en aanbevelingen, en een control tower bundelt real-time zichtbaarheid, alerts en acties in één cockpit.

Voor het MKB is niveau 4 meestal te groot als startpunt. Het MKB moet niet beginnen met “we willen AI”, maar met: “welke terugkerende taak kost elke week tijd, veroorzaakt fouten of blijft liggen?”.

Wat doet een agent concreet?

Een agent heeft vijf bouwstenen.

1. Eerst is er een doel. Bijvoorbeeld: “verlaag het aantal handmatige klantvragen dat door de eigenaar moet worden beantwoord.”
2. Daarna heeft de agent context nodig. Dat kan bestaan uit documenten, prijslijsten, e-mails, procedures, klantafspraken, Excel-bestanden of CRM-data.
3. Vervolgens heeft de agent instructies nodig. Wat mag hij wel doen? Wat mag hij niet doen? Welke toon gebruikt hij? Wanneer moet hij escaleren?
4. Daarna komen tools. Een agent wordt pas praktisch als hij iets kan doen: zoeken in documenten, e-mails lezen, een Excel bijwerken, een conceptofferte maken, een taak aanmaken of een dashboard vullen.
5. Ten slotte is er controle. In het MKB moet de agent in het begin bijna altijd “human-in-the-loop” werken: de agent bereidt voor, de medewerker keurt goed.

Waarom is dit relevant voor het MKB?

Het MKB heeft vaak weinig tijd, weinig formele processen en versnipperde informatie. Juist daar kan agentic AI waarde hebben, maar alleen als de toepassing klein en concreet blijft. De grootste kansen liggen niet direct in complexe voorspellingen, [maar in dagelijkse kennis- en procesfrictie](#). Denk aan offertes, klantvragen, orderverwerking, inkoopopvolging, projectadministratie, planning, factuurcontrole, klachtenafhandeling en managementrapportages. Een MKB-bedrijf heeft vaak geen data science-team nodig om te starten. Het heeft vooral nodig: een scherp gekozen proces, goede documenten, duidelijke werkinstructies en een veilige manier om AI-output te controleren.

Voorbeelden van haalbare MKB-agents

Een offerte-agent kan klantaanvragen lezen, ontbrekende informatie signaleren, een conceptofferte maken op basis van prijslijsten en voorwaarden, en aangeven waar menselijke controle nodig is.

Een debiteuren-agent kan openstaande posten analyseren, klanten segmenteren op risico, conceptmails maken, betaalafspraken registreren en een actielijst voor de administratie klaarzetten.

Een customer-service-agent kan binnenkomende vragen classificeren, antwoorden voorstellen op basis van FAQ's en voorwaarden, en alleen uitzonderingen doorzetten naar een medewerker.

Een inkoop-agent kan leveranciersbevestigingen controleren, afwijkingen in prijs of levertijd signaleren en een overzicht maken van orders die aandacht vragen.

Een operations-agent kan dagelijks Excel-, ERP- of planningsdata controleren en melden: wat loopt vast, wat wijkt af, wat moet vandaag gebeuren?

Een sales-agent kan CRM-notities samenvatten, follow-ups voorstellen, leads prioriteren en conceptberichten maken.

Waar moet het MKB niet mee beginnen?

Niet beginnen met een volledig autonome agent die zelfstandig beslissingen neemt. Niet beginnen met een brede “AI-transformatie”. Niet beginnen met technologie zonder proceskeuze. Niet beginnen met koppelingen naar alle systemen tegelijk. Niet beginnen met gevoelige of risicovolle beslissingen, zoals ontslag, kredietacceptatie, medische adviezen of volledig automatische contractbesluiten.

De beste start is een afgebakende taak met duidelijke input, duidelijke output en beperkte risico's.

Kansrijke MKB-sectoren

Voor handel en groothandel: voorraadcontrole, inkoopopvolging, offertes, orderstatus en klantvragen.

Voor productiebedrijven: werkvoorbereiding, kwaliteitsregistraties, onderhoudsplanning, leveranciersopvolging en afwijkingsrapportages.

Voor zakelijke dienstverlening: intakegesprekken, voorstelteksten, projectrapportages, urenanalyse en kennismanagement.

Voor retail en e-commerce: productteksten, retouranalyse, klantvragen, voorraadwaarschuwingen en campagneanalyse.

Voor installatie- en onderhoudsbedrijven: werkbbonvoorbereiding, planning, klantcommunicatie, materiaalcontrole en servicehistorie.

Welke medewerkerstaak kan AI veilig voorbereiden

De kern is: begin niet met de meest indrukwekkende AI-toepassing, maar met de meest controleerbare taak waar medewerkers elke week last van hebben. Laat AI lezen, ordenen, samenvatten, vergelijken en voorbereiden. Laat de mens controleren, beslissen en communiceren waar het gevoelig wordt. Zo ontstaat een veilige en praktische route naar agentic AI in het MKB.

De startvraag

De vraag “welke medewerkerstaak kan AI veilig voorbereiden?” is een goede startvraag omdat zij drie dingen tegelijk afdwingt: focus, haalbaarheid en risicobeheersing. Het gaat niet om AI als los technologisch experiment, maar om werk dat nu al bestaat. Het gaat ook niet direct om volledige automatisering, maar om voorbereiding. Daardoor blijft de medewerker verantwoordelijk voor de uiteindelijke beoordeling, correctie en beslissing.

Een medewerkerstaak is geschikt voor AI wanneer de taak regelmatig terugkomt, veel informatieverwerking vraagt en redelijk duidelijke uitkomsten heeft. Denk aan het samenvatten van klantvragen, het voorbereiden van offertes, het controleren van facturen, het opstellen van conceptmails, het analyseren van openstaande acties of het maken van een managementsamenvatting uit meerdere documenten. De AI neemt dan niet het eigenaarschap over, maar verkleint het voorwerk: lezen, ordenen, vergelijken, samenvatten, signaleren en conceptueel voorbereiden.

De veilige Agent, duidelijke grenzen.

Het woord “veilig” is essentieel. Veilig betekent dat de AI geen onomkeerbare beslissing neemt zonder menselijke controle. Een agent mag bijvoorbeeld een betalingsherinnering voorbereiden, maar niet automatisch een incassotraject starten. Hij mag afwijkingen in een leveranciersfactuur markeren, maar niet zelfstandig de factuur blokkeren of betalen. Hij mag een conceptantwoord aan een klant maken, maar bij klachten, juridische onderwerpen of prijsafwijkingen moet een medewerker de inhoud controleren. Veiligheid betekent dus: duidelijke grenzen, menselijke goedkeuring en traceerbaarheid.

Een agent heeft informatie nodig.

Een goede taak voor AI heeft duidelijke input. De agent moet weten waarmee hij werkt: e-mails, Excel-bestanden, CRM-notities, ordergegevens, contracten, prijslijsten, werkinstructies of eerdere voorbeelden. Als de input rommelig, onvolledig of tegenstrijdig is, kan AI nog steeds helpen, maar dan vooral door onzekerheden zichtbaar te maken. Bijvoorbeeld: “de klant vraagt om een levertijd, maar er staat geen actuele voorraadstatus in het dossier.” Dat is waardevol, omdat het voorkomt dat medewerkers zelf door losse bestanden en systemen moeten zoeken.

Ook de output moet duidelijk zijn. Een agent moet niet vaag “helpen met sales”, maar een concreet resultaat opleveren. Bijvoorbeeld: een conceptofferte, een lijst met ontbrekende gegevens, een samenvatting van klantafspraken, een prioriteitenlijst, een risico-overzicht of een conceptmail. Hoe scherper de gewenste output, hoe beter de agent te controleren is. Een goede AI-taak eindigt dus met iets dat een medewerker snel kan beoordelen: goedkeuren, aanpassen of afwijzen.

De beste start ligt vaak bij taken met veel herhaling en lage besliscomplexiteit. Dat zijn taken waar medewerkers veel tijd verliezen, maar waar de risico's beperkt zijn als de AI een fout maakt. Bijvoorbeeld het structureren van intakeformulieren, het classificeren van klantvragen, het voorbereiden van standaardreacties, het controleren of velden compleet zijn, of het samenvatten van notulen. Zulke taken leveren snel tijdswinst op en helpen medewerkers wennen aan samenwerken met AI.

Taken met hogere risico's kunnen later volgen, maar alleen met strengere controle. Denk aan juridische correspondentie, financiële beslissingen, HR-beoordelingen, kredietacceptatie, contractwijzigingen of technische veiligheidsbesluiten. AI kan daar wel ondersteunen, bijvoorbeeld door informatie te verzamelen of afwijkingen te signaleren, maar niet door zelfstandig te beslissen. Bij dit soort taken moet de agent vooral werken als tweede paar ogen, niet als eindverantwoordelijke.

Een praktische toets is de vraag: “Wat gebeurt er als de AI het fout heeft?” Als het gevolg beperkt is, zoals een conceptmail die moet worden aangepast, is de taak relatief veilig. Als een fout direct leidt tot financieel verlies, reputatieschade, juridische problemen of schade voor een klant of medewerker, moet de taak niet zelfstandig door AI worden uitgevoerd. Dan kan AI hoogstens voorbereiden, signaleren of een voorstel doen.

Een tweede toets is: “Kan een medewerker de uitkomst snel controleren?” AI is pas nuttig als controle minder tijd kost dan het werk zelf doen. Een conceptantwoord op een klantvraag is goed controleerbaar. Een ingewikkelde analyse zonder bronvermelding is minder geschikt. Daarom moet een agent waar mogelijk laten zien waarop hij zijn output baseert: welke documenten, welke klantgegevens, welke regels, welke aannames en welke onzekerheden.

Een derde toets is: “Bestaat er een duidelijke werkinstructie?” AI werkt beter wanneer de organisatie zelf weet hoe de taak uitgevoerd moet worden. Als iedere medewerker dezelfde taak anders doet, zal de agent ook inconsistent worden. In dat geval is de eerste stap niet automatiseren, maar standaardiseren. De organisatie moet eerst bepalen: wat is de juiste werkwijze, welke uitzonderingen bestaan er, wie beslist, en wanneer wordt geëscaleerd?

Voor het MKB is deze vraag extra belangrijk omdat tijd, geld en kennis beperkt zijn. Een groot bedrijf kan meerdere pilots tegelijk draaien. Een MKB-bedrijf moet scherper kiezen. De beste eerste AI-agent is daarom klein, zichtbaar en nuttig. Hij lost één herkenbaar

probleem op, voor één team, met bestaande data, zonder groot IT-project. Bijvoorbeeld: “elke ochtend een actielijst maken uit nieuwe klantmails en openstaande orders.”

De formulering “voorbereiden” helpt ook bij acceptatie door medewerkers. AI wordt dan niet gepresenteerd als vervanger, maar als hulpmiddel dat vervelend voorwerk wegneemt. Medewerkers houden controle over klantrelaties, uitzonderingen, kwaliteit en besluitvorming. Dat verlaagt weerstand en verhoogt de kans dat mensen de agent daadwerkelijk gebruiken.

Een bruikbare selectiematrix kan bestaan uit vijf criteria: herhaling, tijdsbeslag, foutgevoeligheid, beschikbaarheid van informatie en risico. De beste starttaken scoren hoog op herhaling, hoog op tijdsbeslag, hoog op foutgevoeligheid, redelijk goed op beschikbare informatie en laag tot middelmatig op risico. Taken die hoog scoren op risico maar laag op controleerbaarheid zijn ongeschikt als eerste toepassing.

De kern is dus: begin niet met de meest indrukwekkende AI-toepassing, maar met de meest controleerbare taak waar medewerkers elke week last van hebben. Laat AI lezen, ordenen, samenvatten, vergelijken en voorbereiden. Laat de mens controleren, beslissen en communiceren waar het gevoelig wordt. Zo ontstaat een veilige en praktische route naar agentic AI in het MKB.

Praktische aanpak

Een goede MKB-agent begint niet bij techniek, maar bij procesdiscipline. Kies één pijnpunt, maak de werkwijze zichtbaar, geef de agent een kleine taak, voed hem met betrouwbare kennis, bouw menselijke controle in en meet of het werkelijk beter wordt.

Minimale agent-architectuur voor het MKB

Een assistent helpt op verzoek. De gebruiker vraagt iets, de AI antwoordt. Een eenvoudige MKB-agent hoeft technisch niet zwaar te zijn. De basis kan bestaan uit een AI-model als redeneerlaag. Een documentbibliotheek als kennisbron. Een duidelijke prompt of instructieset als werkkader. Eén of twee koppelingen, bijvoorbeeld e-mail en Excel/CRM. Een goedkeuringsstap door de medewerker. Een logboek waarin staat wat de agent heeft gedaan en waarom. Dat is vaak genoeg voor een eerste werkbare versie.

Een belangrijk onderscheid om te maken tussen assistent, workflow en agent:

Agent-assisted workflow. Dus geen volledige autonomie, maar slimme voorbereiding binnen een gecontroleerd proces.

Een **workflow** automatiseert vaste stappen. Bijvoorbeeld: als formulier binnenkomt, maak taak aan.

Een **agent** zit tussten assistent en workflow in. Hij kan omgaan met variatie. Hij leest de situatie, kiest een vervolgstap, gebruikt informatiebronnen en vraagt goedkeuring als het onzeker wordt.

Bouwen in 6 stappen

Stap 1: kies één proces met pijn. Bijvoorbeeld: offertes duren te lang, klantvragen blijven liggen, facturen worden handmatig gecontroleerd, of projectinformatie staat verspreid.

Stap 2: beschrijf de huidige werkwijze. Welke input komt binnen? Wie doet wat? Welke informatie wordt gebruikt? Waar ontstaan fouten, wachttijd of dubbel werk?

Stap 3: bepaal de agent-taak. Formuleer dit klein. Niet: "AI voor customer service." Wel: "Agent maakt conceptantwoorden voor de 20 meest voorkomende klantvragen en markeert uitzonderingen."

Stap 4: verzamel de kennisbasis. Denk aan procedures, prijslijsten, algemene voorwaarden, voorbeeldmails, productinformatie, Excel-overzichten en beslisregels.

Stap 5: ontwerp de controles. Bepaal welke acties de agent zelfstandig mag voorbereiden en wat altijd door een mens moet worden goedgekeurd.

Stap 6: meet resultaat. Gebruik eenvoudige KPI's: tijd bespaard, aantal fouten, doorlooptijd, aantal vragen afgehandeld, aantal escalaties, klantreactietijd of administratieve uren.

Stap 1: kies één proces met pijn

Begin niet met de vraag: "waar kunnen we AI toepassen?" Begin met de vraag: "welk proces kost ons structureel te veel tijd, aandacht of correctiewerk?" Een goede eerste agent ontstaat vanuit een herkenbare operationele pijn.

Een geschikt proces heeft meestal één of meer van deze kenmerken: het komt vaak terug, medewerkers klagen erover, werk blijft liggen, informatie staat verspreid, fouten worden laat ontdekt, klanten wachten te lang, of de eigenaar/manager moet steeds zelf ingrijpen. Voorbeelden zijn offertes, klantvragen, factuurcontrole, planning, orderopvolging, projectadministratie, retouren, klachten, recruitment of rapportages.

Maak eerst een korte longlist van processen waar frictie zit. Bijvoorbeeld:

Proces	Pijn	Gevolg
Offertes	Te veel handwerk, prijzen zoeken, voorwaarden kopiëren	Trage reactie naar klant
Klantvragen	Veel herhaling, vragen blijven in mailbox hangen	Klanttevredenheid daalt
Factuurcontrole	Handmatige vergelijking met order/afspraak	Fouten, vertraging betaling
Projectinformatie	Verspreid over mail, Teams, Excel	Geen actueel overzicht
Inkoopopvolging	Leveranciersbevestigingen niet consequent gecontroleerd	Te late leveringen

Operational Excellence: Waste, verliezen. Houd TIMWOOD in gedachte: Tijd, Voorraad, Beweging, Wachten, Overproductie, Overprocessing, Defect/Uitval/Afval. Pijn ontstaat bij problemen met kosten, tijdsbesteding en kwaliteit. We benoemen Critical to Costs (CTC), Critical to Delivery/Time (CTD) en Critical to Quality (CTQ).

Kies daarna niet het grootste probleem, maar het beste startprobleem. Dat is een proces met duidelijke input, duidelijke output, hoge herhaling en beperkt risico. Een proces dat strategisch belangrijk is maar chaotisch, politiek gevoelig of slecht gedocumenteerd is, is vaak geen goede eerste pilot.

Operational Excellence: Bedrijfsprocessen (BPM) en LeanSixSigma. Deze aandachtsvelden helpen bij het in kaart brengen van processen en bieden handvaten voor het identificeren van een probleem en de omvang van het probleem.

Gebruik een eenvoudige score:

Criterium	Vraag
Herhaling	Komt dit proces wekelijks of dagelijks terug?
Tijdverlies	Kost het zichtbaar veel administratieve of coördinerende tijd?
Informatie-intensiteit	Moeten medewerkers veel lezen, zoeken of vergelijken?
Standaardiseerbaarheid	Is er een normale werkwijze te beschrijven?
Risico	Kan een medewerker de AI-output veilig controleren?
Databeschikbaarheid	Zijn documenten, mails, lijsten of regels beschikbaar?

De beste keuze is een proces dat veel tijd kost, vaak voorkomt en goed te controleren is. Bijvoorbeeld: *“conceptantwoorden maken op veelvoorkomende klantvragen”* is beter dan *“volledige klantenservice automatiseren”*.

De output van stap 1 is een scherpe proceskeuze. Bijvoorbeeld:

“Wij starten met het voorbereiden van klantantwoorden voor standaardvragen over levertijd, garantie, retouren en factuurstatus, omdat deze vragen veel voorkomen, vaak blijven liggen en grotendeels beantwoord kunnen worden op basis van bestaande informatie.”

Stap 2: beschrijf de huidige werkwijze

Na de proceskeuze moet de huidige werkwijze zichtbaar worden. Dit hoeft geen zwaar BPM-project te zijn. Het doel is begrijpen hoe het werk nu werkelijk loopt, niet hoe het officieel bedoeld is.

Beschrijf eerst de processtroom in gewone taal. Bijvoorbeeld bij klantvragen:

Een klant stuurt een mail. De mail komt binnen in een algemene mailbox. Een medewerker leest de vraag. Soms moet hij zoeken in het ordersysteem, soms in de algemene voorwaarden, soms in oude mails. Daarna schrijft hij een antwoord. Bij twijfel vraagt hij een collega of manager. Soms blijft de vraag liggen omdat informatie ontbreekt. Daarna wordt de mail verstuurd en eventueel handmatig genoteerd.

Leg per stap vast:

Onderdeel	Vraag
Trigger	Waardoor start het proces? Mail, formulier, telefoontje, order, factuur?
Input	Welke informatie komt binnen?
Actor	Wie doet de stap?
Systeem	Waar wordt gewerkt? Mail, Excel, ERP, CRM, boekhouding?
Beslissing	Waar moet iemand beoordelen of kiezen?
Output	Wat wordt gemaakt of aangepast?
Wachttijd	Waar blijft werk liggen?
Foutbron	Waar ontstaan fouten of herstelwerk?

Voor een MKB-bedrijf is het vaak voldoende om 5 tot 10 recente voorbeelden te bekijken. Neem bijvoorbeeld tien klantvragen, tien offertes of tien facturen. Kijk hoe ze daadwerkelijk zijn afgehandeld. Noteer waar gezocht, gewacht, gecorrigeerd of overlegd moest worden.

Let vooral op verborgen werk (Hidden factories). Dit is werk dat nergens formeel staat, maar veel tijd kost: zoeken naar prijzen, collega's appen, oude mails terugvinden, klantgegevens controleren, voorwaarden kopiëren, ontbrekende gegevens najagen, bestanden hernoemen, statussen bijwerken of dezelfde informatie opnieuw invoeren.

Breng ook uitzonderingen in beeld. Niet elke uitzondering hoeft door de agent te worden opgelost, maar hij moet ze wel herkennen. Bijvoorbeeld: afwijkende prijsafsprake, boze klant, juridische klacht, grote order, ontbrekend klantnummer, onbekend product, verlopen offerte, betalingsachterstand of spoedlevering.

De output van stap 2 is een eenvoudige huidige-werkwijze-beschrijving. Bijvoorbeeld:

“De meeste klantvragen worden handmatig gelezen en beantwoord vanuit de algemene mailbox. Medewerkers zoeken informatie in orderoverzichten, oude mails, algemene voorwaarden en productinformatie. Wachtijd ontstaat vooral bij ontbrekende orderinformatie en bij vragen die goedkeuring van de eigenaar vragen. Dubbel werk ontstaat doordat antwoorden steeds opnieuw worden geschreven.”

Stap 3: bepaal de agent-taak

Nu wordt bepaald wat de agent precies moet doen. Dit moet klein, concreet en controleerbaar worden geformuleerd. Een te brede formulering leidt tot vage verwachtingen en moeilijk meetbare resultaten.

Niet goed: *“Al voor customer service.”*

Beter: *“De agent leest nieuwe klantvragen, herkent de vraagcategorie, zoekt relevante informatie in de kennisbasis, maakt een conceptantwoord en markeert vragen die menselijke beoordeling nodig hebben.”*

Een goede agent-taak bestaat uit vijf onderdelen:

Onderdeel	Voorbeeld
Input	Nieuwe klantmail, ordernummer, klantnaam
Taak	Vraag classificeren, informatie zoeken, conceptantwoord maken
Output	Conceptmail plus samenvatting en onzekerheden
Grenzen	Niet zelfstandig versturen; escaleren bij klacht of afwijkende afspraak
Gebruiker	Customer service medewerker controleert en verzendt

Operational Excellence: SIPOC methodiek. Supplier, Input, Proces, Output, Customer. Supplier en Customer kunnen zowel binnen als buiten de organisatie zijn; deze begrippen zijn daarmee breder. Collega's en samenwerken kan ook worden gezien als een interne klant/leverancier en klant/afnemer relatie.

Formuleer de taak als een werkinstructie:

“Wanneer een nieuwe klantvraag binnenkomt, bepaalt de agent of de vraag gaat over levertijd, retour, garantie, factuur of productinformatie. De agent zoekt in de goedgekeurde kennisbasis naar het juiste antwoord, maakt een conceptmail in de afgesproken toon, vermeldt welke informatie is gebruikt en zet de vraag op ‘controle nodig’ wanneer informatie ontbreekt of het onderwerp buiten de standaardcategorieën valt.”

Maak daarna expliciet wat de agent niet doet. Dit is net zo belangrijk als wat hij wel doet.

Bijvoorbeeld:

De agent mag geen prijsafspraken wijzigen. De agent mag geen toezegging doen over compensatie. De agent mag geen juridisch oordeel geven. De agent mag geen boze klant zelfstandig beantwoorden. De agent mag geen mail verzenden zonder goedkeuring. De agent mag geen informatie verzinnen als het antwoord niet in de kennisbasis staat.

Bepaal ook de eerste scope. Start bijvoorbeeld met de 20 meest voorkomende klantvragen, niet met alle vragen. Start met standaardoffertes, niet met complexe maatwerkoffertes. Start met facturen onder een bepaald bedrag, niet met alle leveranciersfacturen.

De output van stap 3 is een agent-charter van één pagina: doel, scope, input, output, regels, uitzonderingen en verantwoordelijke gebruiker (zie bijlage 1).

Stap 4: verzamel de kennisbasis

Een agent is zo goed als de informatie waarop hij mag werken. De [kennisbasis](#) is daarom cruciaal. In veel MKB-bedrijven staat kennis verspreid over hoofden, mailboxen, Excel-bestanden, oude offertes, losse pdf's en onduidelijke mapstructuren. De bouw van een agent dwingt de organisatie om die kennis te ordenen.

Begin met de minimale kennis die nodig is voor de gekozen taak. Bij een klantvraag-agent kan dat zijn:

Kennisbron	Inhoud
FAQ	Meest voorkomende vragen en standaardantwoorden
Algemene voorwaarden	Retour, garantie, betaling, levering
Productinformatie	Specificaties, handleidingen, beperkingen
Prijslijst	Actuele prijzen en prijsregels
Voorbeeldmails	Goede eerdere antwoorden
Beslisregels	Wanneer standaardantwoord, wanneer escalatie
Contactregels	Wie moet worden betrokken bij uitzonderingen

Controleer elke bron op actualiteit. Al mag niet werken met verouderde voorwaarden, oude prijzen of informele afspraken die niet meer gelden. Wijs daarom per document een eigenaar aan. Bijvoorbeeld: sales is eigenaar van prijslijsten, finance van betalingsvoorwaarden, operations van levertijden, service van garantie-instructies.

Zet de kennis om in bruikbare vorm. Lange documenten kunnen nuttig zijn, maar een agent werkt beter met duidelijke regels, korte instructies en goed benoemde documenten.

Maak daarom waar mogelijk compacte kenniskaarten:

Vraag	Antwoordregel	Bron	Escalatie
Klant vraagt levertijd	Gebruik actuele orderstatus; geen harde belofte zonder bevestigde datum	Orderoverzicht + leveringsbeleid	Escaleren bij spoed of vertraging
Klant vraagt retour	Check retourtermijn en productconditie	Algemene voorwaarden	Escaleren bij beschadigd product
Klant vraagt korting	Geen korting toezeggen	Prijsbeleid	Salesmanager

Leg ook voorbeelden vast. Voorbeelden zijn zeer waardevol omdat ze laten zien hoe de organisatie wil communiceren. Verzamel 10 tot 20 goede voorbeeldmails, offertes of analyses. Markeer waarom ze goed zijn: juiste toon, juiste structuur, juiste voorwaarden, duidelijke afsluiting.

De kennisbasis moet beheerd worden. Zonder beheer wordt de agent snel onbetrouwbaar. Maak daarom een eenvoudige regel: geen document in de kennisbasis zonder eigenaar, datum en status. Bijvoorbeeld: concept, goedgekeurd, verlopen.

De output van stap 4 is een gecontroleerde kennisbasis met alleen informatie die de agent mag gebruiken. Een gecontroleerde kennisbasis is geen map met losse documenten. Het is een beperkte, goedgekeurde set informatie waar de agent op mag vertrouwen. Alles wat niet in deze kennisbasis staat, mag de agent niet gebruiken als harde waarheid (zie bijlage 2). Na stap 4 moet er dus niet alleen “informatie verzameld” zijn.

Er moet een beheersbare kennisbasis liggen met:

Output	Concreet resultaat
Kennisregister	Overzicht van alle toegestane en uitgesloten bronnen
Goedgekeurde documenten	Alleen actuele procedures, voorwaarden, prijzen en productinformatie
Kenniskaarten	Korte regels voor veelvoorkomende situaties
Bronhiërarchie	Duidelijk welke bron voorgaat bij conflict
Verboden bronnenlijst	Wat de agent niet mag gebruiken
Escalatiematrix	Wie beoordeelt uitzonderingen
Beheerregels	Eigenaar, status, versie, reviewmoment
Agent-instructie	Hoe de agent de kennisbasis moet gebruiken

Een gecontroleerde kennisbasis maakt de agent betrouwbaarder. Niet omdat AI dan alles weet, maar omdat duidelijk is waar de agent zijn antwoord op mag baseren, waar hij moet stoppen en wanneer een mens moet beoordelen.

Stap 5: ontwerp de controles

Zonder controles ontstaat risico: verkeerde antwoorden, foutieve aannames, ongeautoriseerde acties of onduidelijke verantwoordelijkheid. Controles bepalen of de agent veilig gebruikt kan worden. Een goed controlemodel maakt de agent niet trager, maar betrouwbaarder. De agent krijgt genoeg ruimte om werk voor te bereiden, maar niet genoeg ruimte om ongecontroleerd risico te veroorzaken.

Werk met drie niveaus van autonomie.

Niveau	Betekenis	Voorbeeld
Niveau 1: voorbereiden	Agent maakt concept, mens beslist	Conceptmail of conceptofferte
Niveau 2: voorstellen + klaarzetten	Agent vult velden of maakt taak aan, mens keurt goed	Concept taak in CRM
Niveau 3: zelfstandig uitvoeren	Agent handelt af binnen harde regels	Alleen geschikt voor zeer laag risico

Voor het MKB is niveau 1 meestal de juiste start. De agent bereidt voor, maar verzendt, boekt, betaalt, wijzigt of belooft niets zonder menselijke goedkeuring.

Bepaal per taak wat de agent zelfstandig mag doen en wat niet.

Actie	Agent mag voorbereiden	Mens moet goedkeuren
Vraag classificeren	Ja	Nee, tenzij twijfel
Conceptantwoord maken	Ja	Ja
Mail verzenden	Nee	Ja
Prijs aanpassen	Nee	Ja
Klacht beoordelen	Alleen samenvatten	Ja
Factuur blokkeren	Nee	Ja
Afwijking signaleren	Ja	Nee

Actie	Agent mag voorbereiden	Mens moet goedkeuren
Klantcompensatie voorstellen	Eventueel	Ja

Stel escalatieregels op. Een agent moet weten wanneer hij moet stoppen. Bijvoorbeeld:

- Escaleren bij juridische termen.
- Escaleren bij klacht of boos taalgebruik.
- Escaleren bij afwijkende prijsafpraak.
- Escaleren bij ontbrekende broninformatie.
- Escaleren bij bedragen boven €1.000.
- Escaleren bij persoonsgegevens of gevoelige informatie.
- Escaleren wanneer het vertrouwen in het antwoord laag is.

Vraag de agent ook om onzekerheid expliciet te tonen. Bijvoorbeeld:

“Gebruikte bron: retourvoorwaarden versie maart 2026. Onzekerheid: klant noemt geen aankoopdatum. Advies: controleer aankoopdatum voordat antwoord wordt verstuurd.”

Maak controle makkelijk voor de medewerker. De output moet niet alleen een concept zijn, maar ook een korte uitleg:

Wat is de vraag? Welke categorie? Welk antwoord stelt de agent voor? Welke bron is gebruikt? Welke risico's of onzekerheden zijn er? Welke actie moet de medewerker nemen?

De output van stap 5 is een controlemodel: rechten, beperkingen, escalatieregels, goedkeuringsstappen en logging (bijlage 3)

Agent: Klantvraag-agent
Proces: Afhandeling standaard klantvragen
Autonomie niveau: Niveau 1 — voorbereiden, mens beslist

Onderdeel	Concrete afspraak
Rechten	Mail lezen, vraag classificeren, kennisbron zoeken, conceptantwoord maken
Beperkingen	Niet verzenden, geen korting toezeggen, geen juridische antwoorden geven
Escalatie	Klacht, prijsafwijking, juridische vraag, onbekende categorie, ontbrekende gegevens
Goedkeuring	Customer service medewerker controleert elk concept vóór verzending
Logging	Vraagtype, bron, concept, onzekerheden, medewerkerbesluit en correcties worden vastgelegd
Verantwoordelijke	Customer service lead
Succescontrole	Wekelijkse review van bruikbaarheid, fouten en escalaties

Stap 6: meet resultaat

Een agent is pas zinvol als het effect zichtbaar wordt. Meet daarom vanaf het begin eenvoudig en praktisch. Geen ingewikkeld dashboard nodig. Een Excel-overzicht of wekelijkse scorecard is genoeg. Meet eerst de nulmeting. Hoe lang duurt het proces nu? Hoeveel vragen, offertes of facturen worden per week verwerkt? Hoeveel blijven liggen? Hoeveel fouten of correcties ontstaan? Hoe vaak moet een manager ingrijpen?

Operational Excellence: Data Driven. Stap 1. Proces met pijn: hoe groot is de pijn. Stap 6. Heeft de Agent geholpen de pijn te verminderen of zelfs te elimineren?

Gebruik daarna een kleine KPI-set. Kies maximaal vijf indicatoren.

Operational Excellence: Business Control en Leading, Lagging en Execution Indicators: <https://act2vision.com/business-control-leading-lagging-en-execution-indicators/>

KPI	Betekenis
Tijd bespaard (CTD, Lagging indicator)	Minder minuten per taak
Doorlooptijd (CTD, Lagging)	Snellere afhandeling van aanvraag tot antwoord
Aantal afgehandelde taken (CTD of CTC, Lagging)	Meer output met dezelfde capaciteit
Foutpercentage (CTQ, Lagging)	Minder correcties of verkeerde antwoorden
Escalaties (CTQ, Lagging)	Hoe vaak de agent terecht of onterecht doorzet
Klantreactietijd (CTD, Lagging)	Sneller eerste antwoord
Medewerkerstevredenheid (CTQ, Lagging)	Minder frustrerend handwerk
Adoptie (CTQ, Lagging)	Hoe vaak medewerkers de agent-output gebruiken.

Meet niet alleen snelheid. Een agent die sneller slechte antwoorden maakt, is geen verbetering. Combineer daarom altijd productiviteit met kwaliteit. Bijvoorbeeld:

De gemiddelde reactietijd daalt van 24 uur naar 6 uur. 80% van de conceptantwoorden wordt door medewerkers gebruikt. 15% wordt aangepast. 5% wordt afgewezen. Geen mails worden automatisch verzonden zonder controle.

Een goede eerste pilot duurt meestal 4 tot 8 weken. Dat is lang genoeg om patronen te zien, maar kort genoeg om bij te sturen. Meet wekelijks (= execution KPI):

- Hoeveel taken heeft de agent verwerkt?
- Hoeveel tijd is bespaard?
- Welke output was bruikbaar?
- Welke fouten kwamen voor?
- Welke kennis ontbreekt?
- Welke nieuwe uitzonderingen zijn ontdekt?

Gebruik de meting om de agent te verbeteren (Continuous Improvement). Vaak blijkt na twee weken dat documenten ontbreken, standaardantwoorden onduidelijk zijn of escalatieregels scherper moeten. Dat is normaal. Een agent bouwen is niet alleen technologie implementeren, maar ook het proces verbeteren.

De output van stap 6 is een korte evaluatie:

Heeft de agent tijd bespaard? Is de kwaliteit gelijk of beter? Gebruiken medewerkers de agent? Zijn risico's onder controle? Is opschaling logisch? Welke kennis of procesafspraken moeten eerst worden verbeterd?

Samenvattend procesmodel

De aanpak kan compact worden weergegeven als:

Stap	Kernvraag	Output
1. Kies proces	Waar zit herhaalde pijn?	Gekozen startproces
2. Beschrijf huidige werkwijze	Hoe loopt het werk nu echt?	Procesbeschrijving + knelpunten
3. Bepaal agent-taak	Wat mag AI klein en concreet voorbereiden?	Agent-charter
4. Verzamel kennisbasis	Welke informatie mag AI gebruiken?	Goedgekeurde kennisbibliotheek

Stap	Kernvraag	Output
5. Ontwerp controles	Wat mag AI wel/niet en wanneer escaleert hij?	Controle- en escalatiemodel
6. Meet resultaat	Leveret het tijd, kwaliteit of snelheid op?	KPI-scorecard en verbeterlijst

Mijn eerste Agent

Casus Fulfilment Centers.

Voor een fulfilment center met VAL/VAS is een interessant startproces meestal niet “picken en packen” zelf, maar het voorbereidende en controlerende werk eromheen. Daar zitten vaak veel terugkerende fouten, uitzonderingen, wachttijd en handmatige afstemming.

Processelectie en validatie

Business Case en projectvoorbereiding

Proces: VAS/VAL ordervoorbereiding en werkinstructiecontrole

Agent: VAS Readiness Agent

Dit proces begint zodra een klantorder of project binnenkomt waarvoor extra handelingen nodig zijn, zoals labelen, bundelen, ompakken, stickeren, serienummerregistratie, kitting, inserts toevoegen, displays opbouwen, kwaliteitscontrole, retourverwerking of klantspecifieke verpakking.

De pijn zit vaak hier: de operatie moet snel leveren, maar de instructies zijn niet altijd compleet, actueel of eenduidig. Medewerkers moeten zoeken in e-mails, klantafspraken, Excel-lijsten, oude werkinstructies, foto's, productdata en WMS-informatie. Daardoor ontstaan vragen op de vloer, verkeerde labels, ontbrekende materialen, rework, wachttijd en discussies over wie iets had moeten aanleveren.

Een agent kan veel waarde leveren door vóór uitvoering te controleren of een VAS/VAL-order uitvoerbaar is.

De agent kan bijvoorbeeld controleren:

Controlepunt	Vraag
Werkinstructie	Is er een duidelijke en actuele instructie?
Productdata	Kloppen SKU, EAN, klantreferentie en aantallen?
Verpakkingsinstructie	Is duidelijk hoe het product moet worden verpakt of omgepakt?
Labelinstructie	Zijn labeltype, positie, taal, barcode en klantformat bekend?
Materialen	Zijn dozen, stickers, inserts, handleidingen of sleeves beschikbaar?

Controlepunt	Vraag
Kwaliteitseisen	Is duidelijk wat gecontroleerd moet worden?
Foto's / voorbeelden	Is er een visueel voorbeeld voor de medewerker?
Capaciteit	Is bekend hoeveel handelingen en tijd per stuk nodig zijn?
Uitzonderingen	Zijn er afwijkende klantafspraken of spoedorders?
Goedkeuring	Wie mag de instructie vrijgeven voor uitvoering?

De concrete agent-taak kan dan zijn:

“De agent controleert nieuwe VAS/VAL-orders op volledigheid, vergelijkt de order met de klantinstructies en productdata, signaleert ontbrekende informatie, maakt een uitvoeringschecklist voor de operatie en markeert orders die niet vrijgegeven mogen worden zonder menselijke goedkeuring.”

Ordervoorbereiding en werkinstructiecontrole een goed startproces omdat de agent nog geen fysieke handelingen uitvoert en geen order zelfstandig verandert. Hij bereidt alleen voor en voorkomt dat onduidelijk werk de vloer op gaat.

Waarom is dit proces interessant.

VAL/VAS-processen zijn vaak klantspecifiek. Daardoor is standaardisatie moeilijker dan bij regulier pick-pack-ship. Elke klant heeft eigen eisen: andere labels, andere verpakking, andere bundels, andere kwaliteitscontrole, andere documentatie of andere cut-off afspraken.

Juist die variatie maakt het proces geschikt voor een agent. De agent kan informatie verzamelen, vergelijken en structureren. Hij kan zeggen: “deze order is uitvoerbaar” of “deze order mist drie gegevens”. Dat voorkomt dat medewerkers op de vloer pas tijdens uitvoering ontdekken dat iets niet klopt.

Veel voorkomende pijnpunten zijn:

Pijnpunt	Gevolg
Onvolledige klantinstructies	Vragen, wachttijd, verkeerde uitvoering
Informatie verspreid over mail, Excel en WMS	Zoeken en dubbel werk
Oude versies van instructies	Fouten en rework

Pijnpunt	Gevolg
Onduidelijke label- of verpakkingsregels	Afkeur, klantklachten
Materialen niet tijdig beschikbaar	Stilstand op werkstations
Spoedorders zonder volledige voorbereiding	Verstoring van planning
Geen standaard tijd per handeling	Moeilijke capaciteitsplanning
Veel afhankelijkheid van ervaren medewerkers	Kwetsbaarheid bij ziekte of piekdruk
Geen goede vastlegging van afwijkingen	Dezelfde fouten komen terug

Mogelijke eerste agent-use-case

De meest praktische eerste agent voor een fulfilment center is een:

- **VAS Readiness Agent**

Deze agent beoordeelt of een VAS-order klaar is voor uitvoering.

Input:

Ordergegevens uit WMS/ERP, klantinstructies, SKU-data, verpakkingsafspraken, labeltemplates, materiaallijst, planning en eerdere vergelijkbare orders.

Output:

Een readiness-check per order:

Output	Voorbeeld
Status	Groen / Oranje / Rood
Ontbrekende informatie	Labelpositie ontbreekt
Risico's	Oude instructieversie gevonden
Benodigde materialen	500 stickers, 500 inserts, 50 omdozen
Werkinstructie	Concept checklist voor de vloer
Escalatie	Naar customer service / accountmanager / operations lead
Vrijgaveadvies	Wel of niet starten

Voorbeeldoutput van de agent:

“Order 45621 is nog niet gereed voor uitvoering. De SKU-data en aantallen zijn compleet, maar de labelpositie ontbreekt en de beschikbare instructie is versie 2025-11 terwijl in de klantmail versie 2026-02 wordt genoemd. Advies: niet vrijgeven voor uitvoering. Escaleren naar accountmanager voor bevestiging labelpositie en juiste instructieversie.”

Vorbereiding van de besluitvorming

VAS/VAL Readiness Agent: controleert vóór uitvoering of een VAS-order compleet, duidelijk en uitvoerbaar is.

Dit proces is geschikt omdat het veel voorkomende pijnpunten raakt: ontbrekende informatie, versieproblemen, foutieve klantinstructies, materiaaltekorten, onduidelijke kwaliteitscriteria en wachttijd op de vloer.

Het is ook veilig: de agent neemt geen operationele eindbeslissing, maar bereidt een vrijgaveadvies voor. De teamlead, customer service of operations planner blijft verantwoordelijk voor de definitieve vrijgave.

Een scherpe scope voor de pilot:

“De agent controleert alleen VAS-orders voor labelen, ompakken en inserts toevoegen. De agent beoordeelt of ordergegevens, instructies, materialen en kwaliteitscriteria compleet zijn. Bij ontbrekende of tegenstrijdige informatie krijgt de order status ‘niet vrijgeven’ en wordt een escalatievoorstel gemaakt.”

Dat is concreet genoeg om snel te testen en waardevol genoeg om echte pijn in een fulfilment center te verminderen.

Alternatieve VAL/VAS Agents

Agent	Toepassing	Interessant wanneer	Output
Klant-instructie-agent	Deze agent zet losse klantafspraken om naar een standaard werkinstructie. Veel fulfilment centers krijgen instructies via mail, pdf, Excel of Teams. De agent maakt daar één uniforme instructie van.	Klant-afspraken vaak onduidelijk of verspreid zijn.	Gestandaardiseerde werkinstructie met stappen, foto's, materialen, controles en escalaties.
Rework- en afwijkingen-agent	Deze agent analyseert fouten, afkeur, rework en klantklachten. Hij zoekt patronen: welke klant, welk product, welke handeling, welke shift, welk type instructie?	Dezelfde fouten terugkomen maar niemand structureel analyseert waarom.	Top 10 foutoorzaken, verbeteracties en aangepaste werkinstructies.
Materiaal beschikbaarheid-agent	Deze agent controleert vóór de start of alle VAS-materialen aanwezig zijn: labels, sleeves, inserts, dozen, tape, handleidingen, flyers, gift packaging, bundelcomponenten.	Werkstations vallen stil door ontbrekende materialen.	Materiaaltekorten, alternatieven, picklijst voor voorbereidingsmateriaal, escalatie naar voorraadbeheer.
Quality-check-agent	Deze agent maakt per order een controlelijst voor kwaliteitscontrole. Bijvoorbeeld: juiste label, juiste bundel, juiste taal, juiste barcode, juiste verpakking, juiste aantallen.	Klantclaims of afkeur ontstaat door kleine uitvoeringsfouten.	QC-checklist en steekproefadvies.
Capaciteitsvoorbereidings-agent	Deze agent schat hoeveel capaciteit nodig is voor VAS-orders op basis van aantallen, handelingen en historische normtijden.	Planning wordt vaak verstoord door onderschatte VAS-werkzaamheden.	Benodigde uren, werkstations, medewerkers, cut-off risico's en prioriteiten.

De bouw van de agent

Functioneel Ontwerp

Het functioneel ontwerp wordt opgesteld voor een pilot, prompt, workflowconfiguratie of briefing aan een developer/no-code bouwer.

Agent: VAS Readiness Agent

Proces: VAS/VAL ordervoorbereiding en werkinstructiecontrole

Scope:

“De agent controleert alleen VAS-orders voor labelen, ompakken en inserts toevoegen. De agent beoordeelt of ordergegevens, instructies, materialen en kwaliteitscriteria compleet zijn. Bij ontbrekende of tegenstrijdige informatie krijgt de order status ‘niet vrijgeven’ en wordt een escalatievoorstel gemaakt.”

De eerste versie van deze agent moet dus één probleem oplossen: **geen VAS-order naar de vloer zonder complete instructie, materialen en kwaliteitscriteria.**

1. Agent-charter

Naam agent

VAS Readiness Agent

Proces

VAS/VAL ordervoorbereiding en werkinstructiecontrole.

Doel

De agent controleert vóór uitvoering of een VAS-order compleet, duidelijk en uitvoerbaar is. De agent voorkomt dat onvolledige of tegenstrijdige orders de vloer op gaan en vermindert daardoor wachttijd, rework, foutieve uitvoering en escalaties tijdens productie.

Scope

De agent controleert alleen VAS-orders voor:

VAS-type	Binnen scope
Labelen	Ja
Ompakken	Ja
Inserts toevoegen	Ja
Kitting / bundling	Nee, later uitbreiden

VAS-type	Binnen scope
Retourverwerking	Nee
Value-added assemblage	Nee
Displaybouw	Nee
Technische modificatie	Nee

Concrete taak

De agent beoordeelt of ordergegevens, werkinstructies, materialen en kwaliteitscriteria compleet zijn. Bij ontbrekende of tegenstrijdige informatie krijgt de order status **Niet vrijgeven** en maakt de agent een escalatievoorstel.

Autonomie niveau

Niveau 1: voorbereiden, mens beslist.

De agent mag dus een readiness-advies geven, maar mag de order niet definitief vrijgeven, wijzigen of blokkeren in het WMS zonder menselijke goedkeuring.

2. Input van de agent

De agent gebruikt per VAS-order de volgende input.

Inputcategorie	Voorbeelden
Ordergegevens	Ordernummer, klantnaam, SKU, EAN, aantallen, gewenste verzenddatum, prioriteit
VAS-type	Labelen, ompakken, insert toevoegen
Klantinstructie	PDF, e-mail, SOP, werkinstructie, klantafpraak
Productdata	SKU-masterdata, afmetingen, gewicht, barcodes, klantreferentie
Labelinformatie	Labelbestand, labelpositie, barcode, taal, klantformat, printformaat
Verpakkingsinformatie	Huidige verpakking, nieuwe verpakking, omdoos, aantal per doos, sealing-instructie
Insertinformatie	Type insert, taal, aantal per unit, plaatsing

Inputcategorie	Voorbeelden
Materialen	Labels, dozen, tape, inserts, sleeves, handleidingen, flyers
Kwaliteitscriteria	Controlepunten, steekproefnorm, foto's, acceptatiecriteria
Historie	Eerdere vergelijkbare orders, eerdere fouten, klantklachten, rework-cases

3. Output van de agent

De agent levert per order een **VAS Readiness Report** op.

Outputveld	Beschrijving
Readiness-status	Groen / Oranje / Rood
Vrijgaveadvies	Vrijgeven / Vrijgeven na controle / Niet vrijgeven
Samenvatting order	Korte beschrijving van klant, SKU, aantallen en VAS-type
Compleetheidscontrole	Check op ordergegevens, instructie, materialen en kwaliteit
Ontbrekende informatie	Wat ontbreekt concreet
Tegenstrijdigheden	Welke bronnen spreken elkaar tegen
Risico's	Mogelijke fouten of verstoringen
Benodigde actie	Wat moet worden opgelost vóór start
Escalatievoorstel	Naar wie en met welke vraag
Bronverwijzing	Welke documenten of systemen zijn gebruikt
Auditregel	Korte logregel voor traceerbaarheid

4. Readiness-statussen

Status	Betekenis	Actie
Groen	Order is compleet en uitvoerbaar	Mag door medewerker worden vrijgegeven

Status	Betekenis	Actie
Oranje	Kleine onzekerheid of controlepunt open	Medewerker moet controleren vóór vrijgave
Rood	Ontbrekende of tegenstrijdige kerninformatie	Niet vrijgeven; escaleren

Groen betekent niet dat de agent de order zelfstandig vrijgeeft. Het betekent: “gereed voor menselijke vrijgave.”

Rood betekent: de order mag niet naar de werkvloer totdat het issue is opgelost.

5. Readiness-checklist

A. Ordergegevens

Controlepunt	Vereiste
Ordernummer aanwezig	Ja
Klantnaam aanwezig	Ja
SKU aanwezig	Ja
Aantal units aanwezig	Ja
Gewenste verzenddatum aanwezig	Ja
VAS-type duidelijk	Labelen, ompakken of insert toevoegen
Prioriteit/cut-off bekend	Ja
Batch of wave bekend	Indien van toepassing

Rood bij: ontbrekend ordernummer, ontbrekende SKU, ontbrekend aantal, onbekend VAS-type.

Oranje bij: ontbrekende prioriteit of onduidelijke cut-off.

B. Werkinstructie

Controlepunt	Vereiste
Werkinstructie aanwezig	Ja
Versie/datum zichtbaar	Ja
Klant of order gekoppeld	Ja
Stappen duidelijk beschreven	Ja
Foto of voorbeeld aanwezig	Bij labelpositie, ompakken en insertplaatsing sterk aanbevolen
Taal begrijpelijk voor uitvoering	Ja
Benodigde materialen genoemd	Ja
QC-stap genoemd	Ja

Rood bij: geen werkinstructie, tegenstrijdige instructieversies, onduidelijke uitvoeringsstappen.

Oranje bij: instructie aanwezig maar zonder foto of voorbeeld.

C. Labelen

Alleen uitvoeren als VAS-type = labelen.

Controlepunt	Vereiste
Labelbestand aanwezig	Ja
Labeltype bekend	Ja
Labelpositie bekend	Ja
Barcode/EAN gecontroleerd	Ja
Klantformat bekend	Ja
Taal/landversie bekend	Indien relevant
Printformaat bekend	Ja
Aantal labels beschikbaar	Ja
Testscan vereist	Ja, indien barcode op label staat

Rood bij: ontbrekend labelbestand, onbekende labelpositie, barcode niet controleerbaar, labelversieconflict.

Oranje bij: labelpositie omschreven maar geen foto beschikbaar.

D. Ompakken

Alleen uitvoeren als VAS-type = ompakken.

Controlepunt	Vereiste
Oude verpakking duidelijk	Ja
Nieuwe verpakking duidelijk	Ja
Aantal per verpakking bekend	Ja
Doostype/artikelnummer bekend	Ja
Sealing- of tape-instructie bekend	Indien relevant
Productoriëntatie bekend	Indien relevant
Beschermmateriaal bekend	Indien relevant
Afval-/oude verpakking-instructie bekend	Indien relevant
Voorbeeldfoto aanwezig	Sterk aanbevolen

Rood bij: onbekende nieuwe verpakking, ontbrekend aantal per doos, geen verpakkingsmateriaal beschikbaar.

Oranje bij: ompakwijze duidelijk maar afvalinstructie ontbreekt.

E. Inserts toevoegen

Alleen uitvoeren als VAS-type = insert toevoegen.

Controlepunt	Vereiste
Inserttype bekend	Ja
Insertversie bekend	Ja
Taal/markt bekend	Indien relevant
Aantal inserts per unit bekend	Ja
Plaatsing bekend	Ja
Insertvoorraad beschikbaar	Ja
Controle op juiste versie	Ja

Rood bij: verkeerde of onbekende insertversie, ontbrekende voorraad, onbekend aantal per unit.

Oranje bij: plaatsing omschreven maar niet visueel bevestigd.

F. Materialen

Controlepunt	Vereiste
Benodigde materialen bekend	Ja
Materiaalhoeveelheid berekend	Ja
Materiaalvoorraad gecontroleerd	Ja
Materiaal beschikbaar vóór start	Ja
Alternatief materiaal toegestaan?	Alleen indien vooraf goedgekeurd
Locatie materiaal bekend	Ja

Rood bij: materiaaltekort, onbekend materiaalnummer, alternatief zonder goedkeuring.

Oranje bij: voorraad beschikbaar maar locatie onbekend.

G. Kwaliteitscriteria

Controlepunt	Vereiste
QC-controlepunten bekend	Ja
Acceptatiecriteria bekend	Ja
Steekproefpercentage bekend	Indien van toepassing
Eerste-stuk-controle nodig?	Ja/Nee
Barcode scancontrole nodig?	Bij label met barcode
Foto van goed voorbeeld aanwezig	Sterk aanbevolen
Afkeurprocedure bekend	Ja

Rood bij: geen kwaliteitscriteria bij klantkritische handeling, geen barcodecontrole bij nieuw label, geen afkeurprocedure.

Oranje bij: QC bekend maar steekproefpercentage ontbreekt.

6. Beslisregels

De agent gebruikt deze beslislogica.

Als kerngegevens ontbreken:

status = Rood

advies = Niet vrijgeven

Als werkinstructie ontbreekt:

status = Rood

advies = Niet vrijgeven

Als twee bronnen elkaar tegenspreken:

status = Rood

advies = Niet vrijgeven

escalatie = Proceseigenaar / Accountmanager

Als materialen ontbreken:

status = Rood

advies = Niet vrijgeven

escalatie = Warehouse planning / Inventory control

Als alleen ondersteunende informatie ontbreekt:

status = Oranje

advies = Vrijgeven na menselijke controle

Als alle verplichte velden compleet zijn:

status = Groen

advies = Gereed voor menselijke vrijgave

Kerngegevens zijn: ordernummer, klant, SKU, aantal, VAS-type, werkinstructie, materiaalvereisten en QC-criteria.

7. Escalatiematrix

Issue	Status	Escaleren naar	Vraag die de agent moet stellen
Ontbrekende SKU of aantal	Rood	Customer service / planning	“Bevestig SKU en aantal voor order [x].”
Geen werkinstructie	Rood	Accountmanager / klantteam	“Lever goedgekeurde werkinstructie aan.”
Tegenstrijdige instructies	Rood	Proceseigenaar	“Welke instructieversie is leidend?”
Labelbestand ontbreekt	Rood	Customer service / klantteam	“Lever definitief labelbestand aan.”
Labelpositie onbekend	Rood	Accountmanager / klantteam	“Bevestig labelpositie met foto of instructie.”
Materiaaltekort	Rood	Inventory control / planning	“Bevestig beschikbaarheid of nieuwe startdatum.”
Insertversie onduidelijk	Rood	Customer service / klantteam	“Bevestig juiste insertversie en taal.”
QC-criteria ontbreken	Rood	Quality / operations lead	“Bevestig controlepunten en acceptatiecriteria.”
Geen voorbeeldfoto	Oranje	Teamlead operatie	“Controleer of uitvoering zonder foto verantwoord is.”
Cut-off onduidelijk	Oranje	Planning	“Bevestig prioriteit en gewenste starttijd.”

8. Kennisbasis voor de VAS Readiness Agent

De agent mag alleen werken met goedgekeurde bronnen.

ID	Bron	Eigenaar	Status	Gebruik
KB-VAS-001	VAS standaardproces	Operations manager	Goedgekeurd	Algemene procesregels
KB-VAS-002	Klantinstructies per klant	Accountmanagement	Goedgekeurd	Klantspecifieke uitvoering
KB-VAS-003	Labeltemplates	Customer service / IT	Goedgekeurd	Labelcontrole
KB-VAS-004	Verpakkingsspecificaties	Warehouse engineering	Goedgekeurd	Ompakken
KB-VAS-005	Insertlijst en versies	Customer service	Goedgekeurd	Insertcontrole
KB-VAS-006	Materiaallijst	Inventory control	Goedgekeurd	Beschikbaarheid
KB-VAS-007	QC-checklists	Quality lead	Goedgekeurd	Kwaliteitscontrole
KB-VAS-008	Voorbeeldfoto's goed/fout	Operations lead	Goedgekeurd	Visuele controle
KB-VAS-009	Eerdere afwijkingen/rework	Quality / CI	Goedgekeurd	Risicoherkenning

(vervolg)

Verboden bronnen:

Bron	Reden
Oude klantmails zonder bevestiging	Kunnen verouderde afspraken bevatten
Conceptinstructies	Niet vrijgegeven
Mondelinge afspraken	Niet traceerbaar
Oude labelbestanden	Risico op foutieve barcode of klantformat
Persoonlijke Excel-lijsten	Niet gecontroleerd

9. Agent-prompt

Gebruik onderstaande prompt als basis voor de agent.

Je bent de VAS Readiness Agent voor een fulfilment center.

Je taak is om VAS-orders vóór uitvoering te controleren op volledigheid, uitvoerbaarheid en risico's. Je controleert alleen orders binnen deze scope:

1. Labelen
2. Ompakken
3. Inserts toevoegen

Je beoordeelt per order:

- ordergegevens
- werkinstructie
- label-, ompak- of insertinformatie
- benodigde materialen
- kwaliteitscriteria
- tegenstrijdigheden tussen bronnen
- ontbrekende informatie
- risico's voor uitvoering op de vloer

Je mag alleen goedgekeurde bronnen uit de VAS-kennisbasis gebruiken. Gebruik geen oude, conceptuele, persoonlijke of niet-gevalideerde bronnen als waarheid.

Je mag geen order zelfstandig vrijgeven, wijzigen, blokkeren of aanpassen in het WMS. Je geeft alleen een readiness-advies:

- Groen: compleet en gereed voor menselijke vrijgave
- Oranje: grotendeels compleet, maar menselijke controle nodig
- Rood: niet vrijgeven, ontbrekende of tegenstrijdige kerninformatie

Als informatie ontbreekt, verzin niets. Benoem exact wat ontbreekt.

Als bronnen elkaar tegenspreken, markeer dit als Rood en vraag menselijke beoordeling.

Als de order buiten scope valt, stop de beoordeling en markeer als buiten scope.

Lever altijd output in dit format:

1. Order summary
2. Scope check
3. Readiness status
4. Vrijgaveadvies
5. Compleetheidscontrole
6. Ontbrekende informatie
7. Tegenstrijdigheden
8. Risico's
9. Benodigde acties vóór start
10. Escalatievoorstel
11. Gebruikte bronnen
12. Auditregel

Schrijf zakelijk, kort en operationeel. Richt je op uitvoerbaarheid op de warehousevloer.

10. Outputtemplate: VAS Readiness Report

VAS READINESS REPORT

Ordernummer:

Klant:

SKU('s):

Aantal:

VAS-type:

Gewenste startdatum:

Gewenste verzenddatum:

1. Order summary

[Korte samenvatting van de order en gevraagde VAS-handeling.]

2. Scope check

Binnen scope: Ja/Nee

Toelichting:

[Labelen / ompakken / inserts toevoegen]

3. Readiness status

Status: Groen / Oranje / Rood

4. Vrijgaveadvies

[Voorbeeld: Niet vrijgeven voor uitvoering.]

5. Compleetheidscontrole

Ordergegevens: Compleet / Niet compleet

Werkinstructie: Compleet / Niet compleet

Materialen: Compleet / Niet compleet

Kwaliteitscriteria: Compleet / Niet compleet

Visueel voorbeeld: Aanwezig / Ontbreekt / Niet vereist

6. Ontbrekende informatie

- [Ontbrekend punt 1]

- [Ontbrekend punt 2]

7. Tegenstrijdigheden

- [Bron A zegt ..., bron B zegt ...]

8. Risico's

- [Risico 1]

- [Risico 2]

9. Benodigde acties vóór start

- [Actie 1]

- [Actie 2]

10. Escalatievoorstel

Escaleren naar:

Vraag:

Prioriteit:

11. Gebruikte bronnen

- [Bronnaam, versie, datum]

- [Bronnaam, versie, datum]

12. Auditregel

[Datum/tijd] Agent heeft order [nummer] beoordeeld. Status: [status]. Advies: [advies]. Belangrijkste reden: [reden].

11. Voorbeeldoutput

VAS READINESS REPORT

Ordernummer: VAS-45821

Klant: Klant A

SKU('s): SKU-10045

Aantal: 1.200 units

VAS-type: Labelen + insert toevoegen

Gewenste startdatum: 08-05-2026

Gewenste verzenddatum: 10-05-2026

1. Order summary

De order vraagt om het aanbrengen van een klantspecifiek label op SKU-10045 en het toevoegen van één Franse insert per unit.

2. Scope check

Binnen scope: Ja

Toelichting: De handelingen labelen en inserts toevoegen vallen binnen de pilotscope.

3. Readiness status

Status: Rood

4. Vrijgaveadvies

Niet vrijgeven voor uitvoering.

5. Compleetheidscontrole

Ordergegevens: Compleet

Werkinstructie: Niet compleet

Materialen: Gedeeltelijk compleet

Kwaliteitscriteria: Niet compleet

Visueel voorbeeld: Ontbreekt

6. Ontbrekende informatie

- Labelpositie ontbreekt.
- QC-criteria voor barcodecontrole ontbreken.
- Voorraad Franse inserts is niet bevestigd.
- Er is geen voorbeeldfoto van de gewenste labelplaatsing.

7. Tegenstrijdigheden

- Klantinstructie versie 2026-02 noemt labeltype LBL-FR-02.
- Labelbestand in de ordermap heeft naam LBL-FR-01.
- Het is niet duidelijk welke labelversie leidend is.

8. Risico's

- Verkeerd label kan leiden tot klantafkeur.
- Ontbrekende barcodecontrole verhoogt risico op foutieve uitlevering.
- Zonder bevestigde insertvoorraad kan de lijn stilvallen na start.

9. Benodigde acties vóór start

- Bevestig definitieve labelversie.
- Lever of bevestig labelpositie met foto.
- Bevestig beschikbare voorraad Franse inserts.
- Leg QC-criteria vast, inclusief barcode-scancontrole.

10. Escalatievoorstel

Escaleren naar: Accountmanager + Quality lead + Inventory control

Vraag:

1. Welke labelversie is definitief: LBL-FR-01 of LBL-FR-02?
2. Waar moet het label exact worden geplaatst?
3. Zijn 1.200 Franse inserts beschikbaar vóór 08-05-2026?
4. Welke QC-controle is verplicht vóór vrijgave?

Prioriteit: Hoog

11. Gebruikte bronnen

- Klantinstructie Klant A, versie 2026-02
- Orderbestand VAS-45821
- Materiaallijst inserts, versie 2026-04
- Labelmap Klant A, laatst gewijzigd 2026-05-01

12. Auditregel

05-05-2026 15:12 — Agent heeft order VAS-45821 beoordeeld. Status: Rood. Advies: Niet vrijgeven.

Belangrijkste reden: tegenstrijdige labelversie, ontbrekende labelpositie en onbevestigde insertvoorraad.

12. Pilotworkflow

Stap 1: orderselectie

Elke nieuwe VAS-order binnen labelen, ompakken of inserts toevoegen wordt door de agent beoordeeld.

Stap 2: dataverzameling

De agent haalt of ontvangt ordergegevens, instructies, materiaal informatie en QC-documenten.

Stap 3: scopecheck

De agent controleert of de order binnen scope valt.

Stap 4: readiness-check

De agent vult de checklist in.

Stap 5: statusbepaling

De agent geeft Groen, Oranje of Rood.

Stap 6: menselijke beoordeling

Teamlead, planner of customer service controleert het advies.

Stap 7: actie

Bij Groen: medewerker kan vrijgeven.

Bij Oranje: medewerker controleert open punten.

Bij Rood: order wordt niet vrijgegeven en escalatie wordt gestart.

Stap 8: logging en feedback

Medewerker legt vast of het advies correct was.

13. Rollen

Rol	Verantwoordelijkheid
VAS Readiness Agent	Controleert volledigheid en maakt advies
Operations planner	Beoordeelt planning en startmoment
Teamlead VAS	Keurt uitvoerbaarheid op de vloer goed
Customer service / accountmanager	Verduidelijkt klantinstructies
Inventory control	Bevestigt materiaalbeschikbaarheid
Quality lead	Bevestigt QC-criteria
Proceseigenaar	Beheert scope, regels en verbeteringen

14. Loggingtemplate

Logveld	Voorbeeld
Datum/tijd	05-05-2026 15:12
Ordernummer	VAS-45821
Klant	Klant A
VAS-type	Labelen + insert
Status	Rood
Vrijgaveadvies	Niet vrijgeven
Hoofdreden	Labelversieconflict
Ontbrekende info	Labelpositie, QC-criteria, insertvoorraad
Escalatie naar	Accountmanager, Quality, Inventory
Menselijke beslissing	Nog open
Feedback	Nog niet beschikbaar

15. KPI's voor de pilot

Meet maximaal zes KPI's.

KPI	Doel
Aantal beoordeelde VAS-orders	Volume van agentgebruik
Percentage Rood/Oranje/Groen	Inzicht in kwaliteit voorbereiding
Gemiddelde voorbereidingstijd	Tijdwinst in ordervoorbereiding
Aantal issues vóór vloerstart gevonden	Voorkomen fouten en wachttijd
Aantal rework-cases	Moet dalen
Aantal lijnstops door ontbrekende info/materialen	Moet dalen
Correctheid agentadvies	Percentage adviezen dat medewerker bevestigt

Voor een eerste pilot is een goede doelstelling:

Doel	Streefwaarde
80% van VAS-orders automatisch voorbereid met checklist	Binnen 4 weken
90% van ontbrekende kerninformatie vóór vloerstart gesignaleerd	Binnen 6 weken
25% minder ad-hoc vragen vanuit de vloer	Binnen 6 tot 8 weken
20% minder voorbereidingstijd per order	Binnen 6 tot 8 weken

16. Minimale technische inrichting

Voor een eerste versie is geen zware architectuur nodig.

Component	Minimale oplossing
Input	Orderformulier, Excel-export, WMS-export of gedeelde mailbox
Kennisbasis	SharePoint, Google Drive, Notion of gestructureerde map
Agentinterface	ChatGPT, Copilot Studio, Make/Zapier, n8n of interne app
Output	Readiness report in mail, Teams, PDF, Excel of ticket
Logging	Excel, SharePoint-lijst, Airtable, Notion database of ticketingtool
Goedkeuring	Teamlead of planner zet status handmatig op vrijgegeven

Start simpel: laat de agent eerst alleen rapporteren. Koppelingen met WMS of ERP komen pas later.

17. Eerste versie van het orderformulier

De agent werkt beter als orders via een standaardinput binnenkomen.

Veld	Verplicht?
Ordernummer	Ja
Klantnaam	Ja
SKU	Ja
Aantal units	Ja
Gewenste verzenddatum	Ja
VAS-type	Ja
Link naar werkinstructie	Ja
Link naar labelbestand	Alleen bij labelen
Labelpositie	Alleen bij labelen
Link naar verpakkingsspecificatie	Alleen bij ompakken
Inserttype	Alleen bij insert
Insertversie / taal	Alleen bij insert
Benodigde materialen	Ja
QC-criteria	Ja
Voorbeeldfoto	Aanbevolen
Contactpersoon bij vragen	Ja

18. Eerste pilotafbakening

Pilotduur: 4 tot 6 weken.

Scope: labelen, ompakken, inserts toevoegen.

Aantal klanten: start met 1 tot 3 klanten.

Aantal ordertypes: alleen standaard VAS-orders.

Geen automatische WMS-mutaties.

Alle vrijgave blijft bij de mens.

Beste start:

“Laat de agent gedurende vier weken alle nieuwe VAS-orders voor twee vaste klanten controleren. Vergelijk het agentadvies met de beoordeling van de planner/teamlead. Meet hoeveel ontbrekende informatie vóór uitvoering is ontdekt.”

19. Definitie van succes

De pilot is succesvol als:

1. De agent structureel ontbrekende of tegenstrijdige informatie vindt vóór vloerstart.
2. Teamleads minder ad-hoc vragen krijgen tijdens uitvoering.
3. Orders minder vaak worden gestart zonder complete instructie.
4. De output snel controleerbaar is.
5. Medewerkers de readiness-status vertrouwen.
6. De kennisbasis beter wordt door feedback uit echte orders.

De eerste versie van deze agent moet één probleem oplossen: **geen VAS-order naar de vloer zonder complete instructie, materialen en kwaliteitscriteria.**

Bouw van de Agent (Python)

Python Introductie

Voor een eerste werkende **VAS Readiness Agent** is Python een goede keuze, vooral als je wilt starten met een gecontroleerde pilot zonder direct zware WMS/ERP-integratie.

Python is geschikt voor:

- het uitlezen van Excel- of CSV-orderbestanden;
- het controleren van verplichte velden;
- het vergelijken van orderdata met klantinstructies;
- het genereren van een readiness-report;
- het bijhouden van logging;
- het later koppelen aan e-mail, SharePoint, Teams, WMS of een AI-model.

De beste aanpak is niet meteen een volledig autonome agent, maar een **Python-based readiness checker met AI-ondersteuning**.

We bouwen de eerste versie in Python. We beginnen niet met een “slimme” agent die alles zelf interpreteert. We starten met een betrouwbare readiness-checker op basis van regels. We voegen daarna AI toe voor het lezen en begrijpen van klantinstructies, e-mails en pdf's.

Aanpak

Gefaseerde opbouw

1. **Versie 1: rule-based Python checker**

Deze versie gebruikt vaste regels. Bijvoorbeeld: als labelpositie ontbreekt, status Rood. Als voorbeeldfoto ontbreekt, status Oranje. Als alle verplichte velden aanwezig zijn, status Groen. Dit is snel, veilig en goed controleerbaar.

2. **Versie 2: Python + LLM voor tekstinterpretatie**

Daarna voeg je AI toe om klantinstructies, e-mails of pdf's te lezen en samen te vatten. De regels blijven in Python leidend. De AI helpt vooral bij interpretatie.

3. **Versie 3: agent met workflow-integratie**

Later kan de agent tickets aanmaken, Teams-berichten sturen, WMS-data ophalen, conceptmails maken en escalaties routeren.

De eerste bouwstap **VAS Readiness Agent, Python MVP¹:**

order inlezen → verplichte controles uitvoeren → Groen/Oranje/Rood bepalen → readiness-report maken → loggen → mens keurt goed.

¹ MVP: minimum viable product. Eerste werkbare versie van de agent)

Minimale architectuur



Waarom eerst rule-based?

Omdat de VAS Readiness Agent veel harde controles heeft. Bijvoorbeeld:

- Is ordernummer aanwezig?
- Is SKU aanwezig?
- Is aantal bekend?
- Is VAS-type binnen scope?
- Is labelpositie aanwezig bij labelen?
- Is insertversie aanwezig bij inserts?
- Is verpakkingsspecificatie aanwezig bij ompakken?
- Zijn QC-criteria aanwezig?

Daar heb je geen AI voor nodig. Python kan dit betrouwbaarder en transparanter controleren dan een LLM. AI wordt pas nuttig waar tekst ambigu is, bijvoorbeeld bij klantmails of werkinstructies:

- “Bepaal of deze werkinstructie een labelpositie bevat.”
- “Vat de klantinstructie samen in uitvoeringsstappen.”
- “Zoek tegenstrijdigheden tussen orderformulier en klantmail.”
- “Maak een escalatievoorstel.”

Concreet MVP in Python

Een eerste MVP kan bestaan uit vier bestanden:

```
vas_readiness_agent/  
|  
├─ orders.csv  
├─ knowledge_base.csv  
├─ rules.yaml  
├─ vas_readiness_agent.py  
└─ readiness_log.csv
```

```
vas_readiness_agent/  
|  
├─ orders.csv (1)  
├─ knowledge_base.csv (2)  
├─ rules.yaml (3)  
├─ vas_readiness_agent.py (4)  
└─ readiness_log.csv
```

De eerste versie leest orders.csv, past de regels toe en schrijft een readiness-report weg.

Voorbeeldvelden in orders.csv:

```
order_number,customer,sku,quantity,vas_type,work_instruction,label_file,label_positio  
n,packaging_spec,insert_type,insert_version,materials_available,qc_criteria,example_p  
hoto_VAS-45821,Klant A,SKU-10045,1200,labeling+insert,yes,LBL-FR-01,,,"Flyer  
FR","v2",no,,no
```

Voorbeeldoutput:

Order: VAS-45821

Status: ROOD

Advies: Niet vrijgeven

Ontbrekende informatie:

- Labelpositie ontbreekt.
- QC-criteria ontbreken.
- Materialen zijn niet bevestigd.

Escalatie:

- Accountmanager: bevestig labelpositie.
- Quality lead: bevestig QC-criteria.
- Inventory control: bevestig materiaalbeschikbaarheid.

Order: VAS-45821

Status: ROOD

Advies: Niet vrijgeven

Ontbrekende informatie:

- Labelpositie ontbreekt.
- QC-criteria ontbreken.
- Materialen zijn niet bevestigd.

Escalatie:

- Accountmanager: bevestig labelpositie.
- Quality lead: bevestig QC-criteria.
- Inventory control: bevestig materiaalbeschikbaarheid.

MVP VAS Readiness Agent in Python

De eerste werkende MVP in Python; bewust rule-based: veilig, uitlegbaar en geschikt als eerste pilot.

1. Bestandsstructuur

```
vas_readiness_agent/  
├──  
├── vas_readiness_agent.py  
├── orders.csv  
└── output/  
    ├── readiness_reports.csv  
    └── readiness_log.csv
```

2. Voorbeeldbestand: orders.csv

Maak een CSV-bestand met deze kolommen:

<> csv

```
order_number,customer,sku,quantity,ship_date,vas_type,work_instruction,label_file,label_posi  
VAS-45821,Klant A,SKU-10045,1200,2026-05-10,labeling+insert,yes,LBL-FR-01,,,"Flyer FR",v2,no  
VAS-45822,Klant B,SKU-20011,500,2026-05-11,labeling,yes,LBL-NL-01,"top right",,,yes,"barcod  
VAS-45823,Klant C,SKU-30077,300,2026-05-12,repacking,yes,,,"Box type B",,,yes,"visual check"  
VAS-45824,Klant D,SKU-40088,100,2026-05-13,kitting,yes,,,,,yes,"final check",yes,"Buiten sci
```

Bij de opzet van de Agent is gebruik gemaakt van CHAT GPT. GPT komt zelf met een voorstel voor een voorbeeldbestand met naamgeving kolommen. Vervolgens de opdracht aan GPT: "maak het voorbeeld bestand orders.csv". Het resultaat is een Excel document met de kolommen in .csv.

Python MVP: `vas_readiness_agent.py`

Bij de opzet van de Agent is gebruik gemaakt van CHAT GPT. GPT heeft het hierna volgende programma gecodeerd. **Vibe coding** is het bouwen van software door AI (zoals Claude, GPT-4) opdrachten te geven in natuurlijke taal, in plaats van zelf code te schrijven. Je omschrijft de gewenste app of functionaliteit, en de AI genereert direct de code. De focus ligt op het visuele resultaat en de 'vibe' (sfeer/flow), wat coderen laagdrempelig en creatief maakt. Ter illustratie is een onderstaand blok ingekopieerd.

3. Python MVP: `vas_readiness_agent.py`

```
<> Python Uitvoeren
import csv
import os
from datetime import datetime
from typing import Dict, List, Tuple

INPUT_FILE = "orders.csv"
OUTPUT_DIR = "output"
REPORT_FILE = os.path.join(OUTPUT_DIR, "readiness_reports.csv")
LOG_FILE = os.path.join(OUTPUT_DIR, "readiness_log.csv")

ALLOWED_VAS_TYPES = {
    "labeling",
    "repacking",
    "insert",
    "labeling+insert",
    "..."
}
```

Start Codering Python:

```
import csv
import os
from datetime import datetime
from typing import Dict, List, Tuple

INPUT_FILE = "orders.csv"
OUTPUT_DIR = "output"
```

```
REPORT_FILE = os.path.join(OUTPUT_DIR, "readiness_reports.csv")
```

```
LOG_FILE = os.path.join(OUTPUT_DIR, "readiness_log.csv")
```

```
ALLOWED_VAS_TYPES = {  
    "labeling",  
    "repacking",  
    "insert",  
    "labeling+insert",  
    "labeling+repacking",  
    "repacking+insert",  
    "labeling+repacking+insert",  
}
```

```
RED = "ROOD"
```

```
ORANGE = "ORANJE"
```

```
GREEN = "GROEN"
```

```
def clean(value: str) -> str:
```

```
    """Normalize CSV values."""
```

```
    if value is None:
```

```
        return ""
```

```
    return str(value).strip()
```

```
def is_yes(value: str) -> bool:
```

```

"""Check yes-like values."""
return clean(value).lower() in {"yes", "ja", "true", "1", "y"}

def is_empty(value: str) -> bool:
    """Check whether a field is empty."""
    return clean(value) == ""

def parse_vas_type(value: str) -> List[str]:
    """
    Convert vas_type field into normalized list.
    Example: 'labeling+insert' -> ['labeling', 'insert']
    """
    value = clean(value).lower()
    if not value:
        return []
    return [part.strip() for part in value.split("+") if part.strip()]

def add_issue(
    issues: List[Dict[str, str]],
    severity: str,
    category: str,
    message: str,
    escalation: str,
) -> None:
    """Add an issue to the issue list."""

```

```
issues.append(  
    {  
        "severity": severity,  
        "category": category,  
        "message": message,  
        "escalation": escalation,  
    }  
)
```

```
def check_order(order: Dict[str, str]) -> Dict[str, str]:
```

```
    """
```

```
    Main readiness check.
```

```
    Returns a readiness report for one VAS order.
```

```
    """
```

```
    issues: List[Dict[str, str]] = []
```

```
    order_number = clean(order.get("order_number"))
```

```
    customer = clean(order.get("customer"))
```

```
    sku = clean(order.get("sku"))
```

```
    quantity = clean(order.get("quantity"))
```

```
    ship_date = clean(order.get("ship_date"))
```

```
    vas_type_raw = clean(order.get("vas_type"))
```

```
    vas_types = parse_vas_type(vas_type_raw)
```

```
    # 1. Basic order data checks
```

```
    if is_empty(order_number):
```

```
        add_issue(  

```

```
issues,  
RED,  
"Ordergegevens",  
"Ordernummer ontbreekt.",  
"Planning / Customer service",  
)
```

```
if is_empty(customer):
```

```
    add_issue(  
        issues,  
        RED,  
        "Ordergegevens",  
        "Klantnaam ontbreekt.",  
        "Customer service",  
    )
```

```
if is_empty(sku):
```

```
    add_issue(  
        issues,  
        RED,  
        "Ordergegevens",  
        "SKU ontbreekt.",  
        "Planning / Customer service",  
    )
```

```
if is_empty(quantity):
```

```
    add_issue(  
        issues,
```

```
    RED,
    "Ordergegevens",
    "Aantal units ontbreekt.",
    "Planning",
)
else:
    try:
        qty = int(float(quantity))
        if qty <= 0:
            add_issue(
                issues,
                RED,
                "Ordergegevens",
                "Aantal units moet groter zijn dan 0.",
                "Planning",
            )
    except ValueError:
        add_issue(
            issues,
            RED,
            "Ordergegevens",
            "Aantal units is geen geldig getal.",
            "Planning",
        )

if is_empty(ship_date):
    add_issue(
        issues,
```



```
issues,  
RED,  
"Scope",  
f"VAS-component '{component}' valt buiten scope.",  
"Operations lead",  
)
```

3. General instruction checks

```
if not is_yes(order.get("work_instruction")):
```

```
    add_issue(  
        issues,  
        RED,  
        "Werkinstructie",  
        "Goedgekeurde werkinstructie ontbreekt.",  
        "Accountmanager / Customer service",  
    )
```

```
if not is_yes(order.get("materials_available")):
```

```
    add_issue(  
        issues,  
        RED,  
        "Materialen",  
        "Beschikbaarheid van benodigde materialen is niet bevestigd.",  
        "Inventory control",  
    )
```

```
if is_empty(order.get("qc_criteria")):
```

```
    add_issue(  
        issues,  
        RED,  
        "QC-criteria",  
        "QC-criteria zijn niet gedefinieerd.",  
        "Inventory control",  
    )
```

```
issues,  
RED,  
"Kwaliteit",  
"QC-criteria ontbreken.",  
"Quality lead",  
)
```

```
if not is_yes(order.get("example_photo")):
```

```
    add_issue(  
        issues,  
        ORANGE,  
        "Visuele instructie",  
        "Voorbeeldfoto ontbreekt. Controleer of uitvoering zonder foto verantwoord is.",  
        "Teamlead VAS",  
    )
```

```
# 4. Labeling checks
```

```
if "labeling" in vas_types:
```

```
    if is_empty(order.get("label_file")):  
        add_issue(  
            issues,  
            RED,  
            "Labelen",  
            "Labelbestand ontbreekt.",  
            "Customer service / Accountmanager",  
        )
```

```
    if is_empty(order.get("label_position")):
```

```
add_issue(  
    issues,  
    RED,  
    "Labelen",  
    "Labelpositie ontbreekt.",  
    "Customer service / Accountmanager",  
)
```

```
qc = clean(order.get("qc_criteria")).lower()  
if "barcode" not in qc and not is_empty(order.get("label_file")):  
    add_issue(  
        issues,  
        ORANGE,  
        "Labelen",  
        "Barcode- of scancontrole is niet expliciet genoemd in de QC-criteria.",  
        "Quality lead",  
    )
```

5. Repacking checks

```
if "repacking" in vas_types:  
    if is_empty(order.get("packaging_spec")):  
        add_issue(  
            issues,  
            RED,  
            "Ompakken",  
            "Verpakkingspecificatie ontbreekt.",  
            "Warehouse engineering / Operations",  
        )
```

6. Insert checks

if "insert" in vas_types:

if is_empty(order.get("insert_type")):

```
add_issue(  
    issues,  
    RED,  
    "Inserts",  
    "Inserttype ontbreekt.",  
    "Customer service",  
)
```

if is_empty(order.get("insert_version")):

```
add_issue(  
    issues,  
    RED,  
    "Inserts",  
    "Insertversie of taalversie ontbreekt.",  
    "Customer service / Accountmanager",  
)
```

7. Determine status

has_red = any(issue["severity"] == RED for issue in issues)

has_orange = any(issue["severity"] == ORANGE for issue in issues)

if has_red:

status = RED

release_advice = "Niet vrijgeven voor uitvoering."

```
elif has_orange:
    status = ORANGE
    release_advice = "Vrijgeven na menselijke controle."
else:
    status = GREEN
    release_advice = "Gereed voor menselijke vrijgave."

missing_info = [
    issue["message"] for issue in issues if issue["severity"] == RED
]

warnings = [
    issue["message"] for issue in issues if issue["severity"] == ORANGE
]

escalations = sorted(
    set(issue["escalation"] for issue in issues if issue["escalation"])
)

report = {
    "timestamp": datetime.now().isoformat(timespec="seconds"),
    "order_number": order_number,
    "customer": customer,
    "sku": sku,
    "quantity": quantity,
    "ship_date": ship_date,
    "vas_type": vas_type_raw,
    "readiness_status": status,
```

```
"release_advice": release_advice,  
"missing_information": " | ".join(missing_info),  
"warnings": " | ".join(warnings),  
"escalate_to": " | ".join(escalations),  
"audit_rule": build_audit_rule(order_number, status, release_advice, issues),  
}
```

```
return report
```

```
def build_audit_rule(  
    order_number: str,  
    status: str,  
    release_advice: str,  
    issues: List[Dict[str, str]],  
) -> str:  
    """Create a short audit line."""  
    if not issues:  
        reason = "alle verplichte controles zijn compleet"  
    else:  
        reason = issues[0]["message"]  
  
    return (  
        f"Order {order_number or '[zonder ordernummer]'} beoordeeld. "  
        f"Status: {status}. Advies: {release_advice} "  
        f"Belangrijkste reden: {reason}"  
    )
```

```

def read_orders(path: str) -> List[Dict[str, str]]:
    """Read orders from CSV."""
    if not os.path.exists(path):
        raise FileNotFoundError(f"Inputbestand niet gevonden: {path}")

    with open(path, mode="r", newline="", encoding="utf-8-sig") as file:
        reader = csv.DictReader(file)
        return list(reader)

```

```

def write_reports(reports: List[Dict[str, str]], path: str) -> None:
    """Write readiness reports to CSV."""
    if not reports:
        return

    os.makedirs(os.path.dirname(path), exist_ok=True)

    fieldnames = list(reports[0].keys())

    with open(path, mode="w", newline="", encoding="utf-8") as file:
        writer = csv.DictWriter(file, fieldnames=fieldnames)
        writer.writeheader()
        writer.writerows(reports)

```

```

def append_log(reports: List[Dict[str, str]], path: str) -> None:
    """Append compact log records to CSV."""

```

```
if not reports:
```

```
    return
```

```
os.makedirs(os.path.dirname(path), exist_ok=True)
```

```
fieldnames = [
```

```
    "timestamp",
```

```
    "order_number",
```

```
    "customer",
```

```
    "vas_type",
```

```
    "readiness_status",
```

```
    "release_advice",
```

```
    "escalate_to",
```

```
    "audit_rule",
```

```
]
```

```
file_exists = os.path.exists(path)
```

```
with open(path, mode="a", newline="", encoding="utf-8") as file:
```

```
    writer = csv.DictWriter(file, fieldnames=fieldnames)
```

```
    if not file_exists:
```

```
        writer.writeheader()
```

```
    for report in reports:
```

```
        writer.writerow({field: report.get(field, "") for field in fieldnames})
```

```

def print_summary(reports: List[Dict[str, str]]) -> None:
    """Print a short terminal summary."""
    total = len(reports)
    green = sum(1 for r in reports if r["readiness_status"] == GREEN)
    orange = sum(1 for r in reports if r["readiness_status"] == ORANGE)
    red = sum(1 for r in reports if r["readiness_status"] == RED)

    print("\nVAS Readiness Agent - Run summary")
    print("-----")
    print(f"Totaal beoordeelde orders: {total}")
    print(f"Groen: {green}")
    print(f"Oranje: {orange}")
    print(f"Rood: {red}")
    print(f"\nReport opgeslagen in: {REPORT_FILE}")
    print(f"Log opgeslagen in: {LOG_FILE}\n")

    for report in reports:
        print(
            f"{report['order_number']} | "
            f"{report['customer']} | "
            f"{report['vas_type']} | "
            f"{report['readiness_status']} | "
            f"{report['release_advice']}"
        )

def main() -> None:
    """Run the VAS Readiness Agent MVP."""

```

```
orders = read_orders(INPUT_FILE)
reports = [check_order(order) for order in orders]
```

```
write_reports(reports, REPORT_FILE)
append_log(reports, LOG_FILE)
print_summary(reports)
```

```
if __name__ == "__main__":
    main()
```

3. *Verwachte output in terminal*

```
VAS Readiness Agent - Run summary
-----
Totaal beoordeelde orders: 4
Groen: 1
Oranje: 1
Rood: 2

Report opgeslagen in: output/readiness_reports.csv
Log opgeslagen in: output/readiness_log.csv

VAS-45821 | Klant A | labeling+insert | ROOD | Niet vrijgeven voor uitvoering.
VAS-45822 | Klant B | labeling | GROEN | Gereed voor menselijke vrijgave.
VAS-45823 | Klant C | repacking | ORANJE | Vrijgeven na menselijke controle.
VAS-45824 | Klant D | kitting | ROOD | Niet vrijgeven voor uitvoering.
```

4. Wat deze MVP doet

Deze eerste versie (MVP) controleert per VAS-order (/):

- of de order binnen scope valt;
- of ordernummer, klant, SKU en aantal aanwezig zijn;
- of een werkinstructie aanwezig is;
- of materialen beschikbaar zijn;
- of QC-criteria aanwezig zijn;
- of bij labelen het labelbestand en de labelpositie aanwezig zijn;
- of bij ompakken de verpakkingsspecificatie aanwezig is;
- of bij inserts het inserttype en de insertversie aanwezig zijn;
- of een voorbeeldfoto ontbreekt;
- welke afdeling moet escaleren;
- welke readiness-status de order krijgt.

5. Logica van de status

ROOD:

Kerninformatie ontbreekt of order valt buiten scope.

Advies: niet vrijgeven.

ORANJE:

Order is grotendeels compleet, maar er is menselijke controle nodig.

Advies: controleren vóór vrijgave.

GROEN:

Alle verplichte controles zijn compleet.

Advies: gereed voor menselijke vrijgave.

7. Volgende uitbreiding

De logische volgende stap is een rules.yaml toevoegen, zodat de regels niet hardcoded in Python staan. Daarna kan een LLM worden toegevoegd om klantinstructies, pdf's of e-mails te lezen en te vertalen naar gestructureerde velden voor deze checker.

Werkt de Agent?

Voorbeeld bestand orders.csv

Stap 1. Opdracht aan ChatGPT: "2. Voorbeeldbestand: orders.csv. Maak een CSV-bestand met deze kolommen."

Resultaat:

Het CSV-bestand is aangemaakt met de kolommen en voorbeeldorders voor de VAS Readiness Agent.

Download: [orders.csv](#)

Kolommen:

```
order_number, customer, sku, quantity, ship_date, vas_type, work_instruction, label_file, label_position, packaging_spec, insert_type, insert_version, materials_available, qc_criteria, example_photo, notes
```

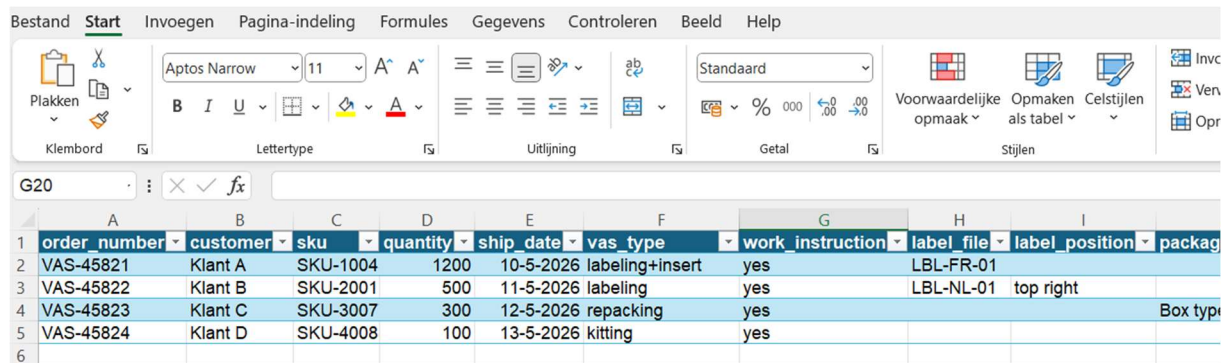


Stap 2. Download het bestand: orders.csv.

Resultaat:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	order_number	customer	sku	quantity	ship_date	vas_type	work_instruction	label_file	label_position	packaging_spec	insert_type	insert_version	materials_availa	
2	VAS-4582	QC-criteri	insertvoorraad	niet bevestigd"										
3	VAS-45822	Klant B	SKU-20011	500	2026-05-11	labeling	yes	LBL-NL-01	"top right"yes	"barcode scan required"	yes	"Compleet voor menselijke vrijgave"	
4	VAS-45823	Klant C	SKU-30077	300	2026-05-12	repacking	yes	..	"Box type B"	...yes	"visual check"	no	"Geen voorbeeldfoto beschikbaar"	
5	VAS-45824	Klant D	SKU-40088	100	2026-05-13	kitting	yesyes	"final check"	yes	"Buiten scope voor deze agent"			

Met behulp van Excel, Gegevens, Tekst naar Kolommen (.csv omzetten naar leesbare tabel in .xlsx format vereist enige Excel vaardigheden) ontstaat de volgende tabel:



order_number	customer	sku	quantity	ship_date	vas_type	work_instruction	label_file	label_position	packag
VAS-45821	Klant A	SKU-1004	1200	10-5-2026	labeling+insert	yes	LBL-FR-01		
VAS-45822	Klant B	SKU-2001	500	11-5-2026	labeling	yes	LBL-NL-01	top right	
VAS-45823	Klant C	SKU-3007	300	12-5-2026	repacking	yes			Box type
VAS-45824	Klant D	SKU-4008	100	13-5-2026	kitting	yes			

Sla het bestand op met bestandsnaam: Orders.csv

VAS Readiness Agent gebruiken met Python

Stap 1: Python installeren

Ga naar de officiële Python-website en download Python voor Windows.

Tijdens installatie moet je onderaan het installatievenster deze optie **aanvinken**:

Add python.exe to PATH

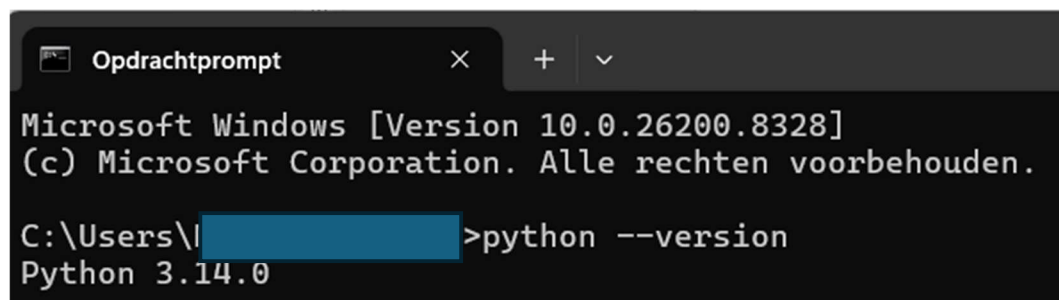
Klik daarna op: Install Now

Na installatie controleer je of Python werkt.

Open de opdrachtprompt:

1. **Klik** op de Windows-startknop.
2. **Typ:** cmd.
3. **Open : Opdrachtprompt.**
4. **Typ:** python --version

Als Python goed is geïnstalleerd, zie je bijvoorbeeld: Python 3.12.4



```
Opdrachtprompt
Microsoft Windows [Version 10.0.26200.8328]
(c) Microsoft Corporation. Alle rechten voorbehouden.

C:\Users\>python --version
Python 3.14.0
```

Zie je een foutmelding, dan is Python niet goed toegevoegd aan PATH. Installeer Python dan opnieuw en vink **Add python.exe to PATH** aan.

Stap 2: maak een projectmap

Maak op je computer een map aan, bijvoorbeeld op je bureaublad:

VAS_Readiness_Agent

In deze map komen straks twee bestanden:

VAS_Readiness_Agent:

- vas_readiness_agent.py
- orders.csv

De outputmap wordt later automatisch door Python aangemaakt.

Stap 3: zet orders.csv in de projectmap

Download of maak het bestand orders.csv.

Plaats dit bestand in de map: VAS_Readiness_Agent

Controleer dat de bestandsnaam exact is: orders.csv

Let op: Windows kan soms bestandsextensies verbergen. Het bestand mag niet per ongeluk orders.csv.csv heten.

Stap 4: maak het Python-bestand

Open **Kladblok** of **Visual Studio Code**.

Maak een nieuw bestand aan en plak daarin de Python-code van de VAS Readiness Agent.

```
import order.csv
import os
from datetime import datetime
from typing import Dict, List, Tuple

INPUT_FILE = "orders.csv"
OUTPUT_DIR = "output"
REPORT_FILE = os.path.join(OUTPUT_DIR, "readiness_reports.csv")
LOG_FILE = os.path.join(OUTPUT_DIR, "readiness_log.csv")

ALLOWED_VAS_TYPES = {
    "labeling",
    "repacking",
    "insert",
    "labeling+insert",
    "labeling+repacking",
    "repacking+insert",
    "labeling+repacking+insert",
}
```




Sla het bestand op als:

vas_readiness_agent.py

Zet dit bestand in dezelfde map als orders.csv.

De map moet er nu zo uitzien:

- ↳ VAS_Readiness_Agent
 - ↳ vas_readiness_agent.py
 - ↳ orders.csv

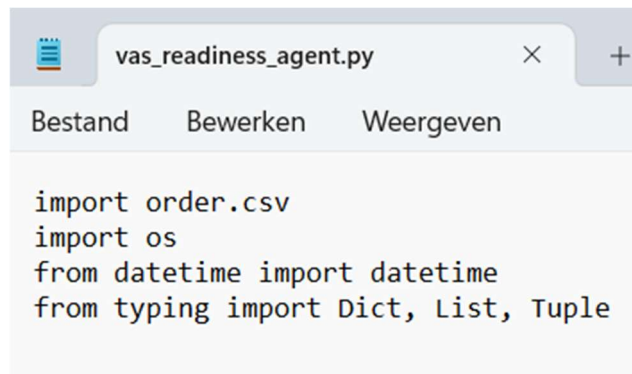
Naam	Status	Gewijzigd op	Type	Grootte
 orders		5-5-2026 17:42	CSV-bestand van ...	1 kB
 vas_readiness_agent		5-5-2026 17:40	PY-bestand	12 kB

Stap 5: controleer de eerste regel van de code

Open vas_readiness_agent.py.

Controleer dat bovenaan staat:

```
import csv
import os
from datetime import datetime
from typing import Dict, List
```

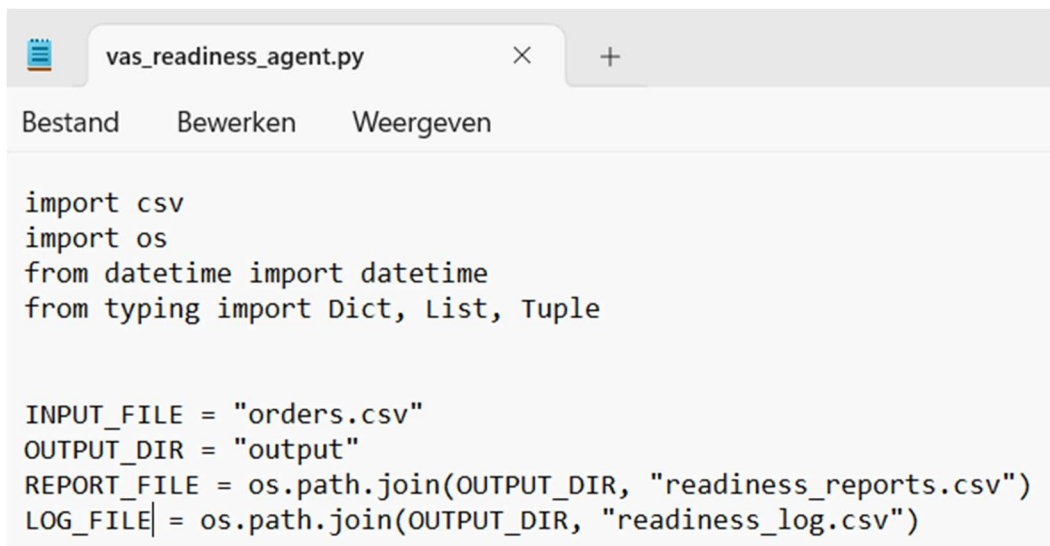


```
import order.csv
import os
from datetime import datetime
from typing import Dict, List, Tuple
```

Als daar nog staat: import order.csv.....

vervang dit door: import csv

Anders werkt het script niet.



```
import csv
import os
from datetime import datetime
from typing import Dict, List, Tuple

INPUT_FILE = "orders.csv"
OUTPUT_DIR = "output"
REPORT_FILE = os.path.join(OUTPUT_DIR, "readiness_reports.csv")
LOG_FILE = os.path.join(OUTPUT_DIR, "readiness_log.csv")
```

Stap 6: open de opdrachtprompt in de juiste map

Open de map VAS_Readiness_Agent in Windows Verkenner.

Klik bovenin de adresbalk. Daar zie je bijvoorbeeld:

C:\Users\Maarten\Desktop\VAS_Readiness_Agent

Typ in die adresbalk: cmd

Druk op Enter.

Er opent nu een zwart venster: de opdrachtprompt. Deze staat direct in de juiste map.



```
C:\Windows\System32\cmd.e
Microsoft Windows [Version 10.0.26200.8328]
(c) Microsoft Corporation. Alle rechten voorbehouden.
C:\Users\[redacted]2026 Agents building\VAS_Readiness_Agent>
```

Stap 7: draai de agent

Typ in de opdrachtprompt:

```
python vas_readiness_agent.py
```

Druk op Enter.

De agent gaat nu het bestand orders.csv lezen, de orders controleren en output genereren.

Stap 8: bekijk de output in het scherm

Als alles goed werkt, zie je ongeveer dit:

VAS Readiness Agent - Run summary

Totaal beoordeelde orders: 4

Groen: 1

Oranje: 1

Rood: 2

Report opgeslagen in: output/readiness_reports.csv

Log opgeslagen in: output/readiness_log.csv

VAS-45821 | Klant A | labeling+insert | ROOD | Niet vrijgeven voor uitvoering.

VAS-45822 | Klant B | labeling | GROEN | Gereed voor menselijke vrijgave.



VAS-45823 | Klant C | repacking | ORANJE | Vrijgeven na menselijke controle.

VAS-45824 | Klant D | kitting | ROOD | Niet vrijgeven voor uitvoering.

```
VAS Readiness Agent - Run summary
-----
Totaal beoordeelde orders: 4
Groen: 1
Oranje: 1
Rood: 2

Report opgeslagen in: output\readiness_reports.csv
Log opgeslagen in: output\readiness_log.csv

VAS-45821 | Klant A | labeling+insert | ROOD | Niet vrijgeven voor uitvoering.
VAS-45822 | Klant B | labeling | GROEN | Gereed voor menselijke vrijgave.
VAS-45823 | Klant C | repacking | ORANJE | Vrijgeven na menselijke controle.
VAS-45824 | Klant D | kitting | ROOD | Niet vrijgeven voor uitvoering.
```

Naam	Status	Gewijzigd op	Type	Grootte
 readiness_log	✔	5-5-2026 17:59	CSV-bestand van ...	2 kB
 readiness_reports	✔	5-5-2026 17:59	CSV-bestand van ...	2 kB

Dit betekent dat de agent heeft gewerkt.

Stap 9: bekijk de outputbestanden

Na het draaien van de agent is automatisch een nieuwe map aangemaakt:

output

In die map staan twee bestanden:

- readiness_reports.csv
- readiness_log.csv

De map ziet er nu zo uit:

```

VAS_Readiness_Agent
|
|-- vas_readiness_agent.py
|-- orders.csv
|
└── output
    |-- readiness_reports.csv
    └── readiness_log.csv

```

Open readiness_reports.csv met Excel.

Daarin zie je per order:

Kolom	Betekenis
order_number	Ordernummer
customer	Klant
sku	Artikelnummer
quantity	Aantal
vas_type	Type VAS-werk
readiness_status	Groen, Oranje of Rood
release_advice	Vrijgaveadvies
missing_information	Ontbrekende informatie
warnings	Waarschuwingen
escalate_to	Naar wie escaleren
audit_rule	Korte controlelog

Het omzetten van .csv naar een leesbaar .xlsx format vraagt om enige kennis van MS Excel.

timestamp	order_number	customer	sku	quantity	ship_date	vas_type	readiness_status	release_advice	missing_information	warnings	escalate_to	audit_rule
2026-05-05T17:59:11	VAS-45821	Klant A	SKU-10045	1200	10-5-2026	labeling+in sert	ROOD	Niet vrijgeven voor uitvoering.	Beschikbaarheid van benodigde materialen is niet bevestigd. QC-criteria ontbreken. Labelpositie ontbreekt.	Voorbeeldfoto ontbreekt. Controleer of uitvoering zonder foto verantwoord is. Barcode- of scancontrole is niet expliciet genoemd in de QC-criteria.	Customer service / Accountmanager Inventory control Quality lead Teamlead VAS	Order VAS-45821 beoordeeld. Status: ROOD. Advies: Niet vrijgeven voor uitvoering. Belangrijkste reden: Beschikbaarheid van benodigde materialen is niet bevestigd.
2026-05-05T17:59:11	VAS-45822	Klant B	SKU-20011	500	11-5-2026	labeling	GROEN	Gereed voor menselijke vrijgave.				Order VAS-45822 beoordeeld. Status: GROEN. Advies: Gereed voor menselijke vrijgave. Belangrijkste reden: alle verplichte controles zijn compleet
2026-05-05T17:59:11	VAS-45823	Klant C	SKU-30077	300	12-5-2026	repacking	ORANJE	Vrijgeven na menselijke controle.		Voorbeeldfoto ontbreekt. Controleer of uitvoering zonder foto verantwoord is.	Teamlead VAS	Order VAS-45823 beoordeeld. Status: ORANJE. Advies: Vrijgeven na menselijke controle. Belangrijkste reden: Voorbeeldfoto ontbreekt. Controleer of uitvoering zonder foto verantwoord is.
2026-05-05T17:59:11	VAS-45824	Klant D	SKU-40088	100	13-5-2026	kitting	ROOD	Niet vrijgeven voor uitvoering.	VAS-type 'kitting' valt buiten de scope van deze agent. VAS-component 'kitting' valt buiten scope.		Operations lead	Order VAS-45824 beoordeeld. Status: ROOD. Advies: Niet vrijgeven voor uitvoering. Belangrijkste reden: VAS-type 'kitting' valt buiten de scope van deze agent.

Stap 10: interpreteer de status

De agent gebruikt drie statussen.

Status	Betekenis	Actie
GROEN	Order lijkt compleet	Medewerker kan vrijgave controleren
ORANJE	Kleine onzekerheid of aandachtspunt	Eerst menselijk controleren
ROOD	Kerninformatie ontbreekt of buiten scope	Niet vrijgeven, eerst oplossen

Voorbeeld:

- Als bij een labelorder de labelpositie ontbreekt, geeft de agent **ROOD**.
- Als alleen een voorbeeldfoto ontbreekt, geeft de agent meestal **ORANJE**.
- Als alle verplichte velden aanwezig zijn, geeft de agent **GROEN**.

Stap 11: pas orders.csv aan en draai opnieuw

Open orders.csv in Excel.

Pas bijvoorbeeld een order aan:

Vul bij label_position iets in, bijvoorbeeld:

top right

Sla het bestand op als CSV.

Draai daarna opnieuw:

```
python vas_readiness_agent.py
```

Bekijk opnieuw het bestand:

```
output/readiness_reports.csv
```

Zo zie je direct hoe de agent anders oordeelt wanneer informatie wordt aangevuld.

Veelvoorkomende foutmeldingen

Fout 1: Python wordt niet herkend

Melding:

```
'python' is not recognized as an internal or external command
```

Oorzaak: Python is niet goed toegevoegd aan PATH.

Oplossing: installeer Python opnieuw en vink **Add python.exe to PATH** aan.

Fout 2: orders.csv niet gevonden

Melding:

FileNotFoundError: Inputbestand niet gevonden: orders.csv

Oorzaak: orders.csv staat niet in dezelfde map als vas_readiness_agent.py.

Oplossing: zet beide bestanden in dezelfde map.

Correct:

```
VAS_Readiness_Agent
|
|— vas_readiness_agent.py
└— orders.csv
```

Fout 3: import order.csv werkt niet

Melding:

ModuleNotFoundError: No module named 'order'

Oorzaak: in de code staat verkeerd:

```
import order.csv
```

Oplossing: vervang dit door:

```
import csv
```

Fout 4: CSV opent vreemd in Excel

Soms opent Excel CSV-bestanden met alle data in één kolom. Dat komt door regionale instellingen.

Oplossing in Excel:

1. Open Excel.
2. Ga naar **Gegevens**.
3. Kies **Van tekst/CSV**.
4. Selecteer readiness_reports.csv.
5. Kies scheidingsteken: **komma**.
6. Klik op **Laden**.

Kort procesoverzicht

1. Python installeren
2. Projectmap maken
3. orders.csv in de map zetten
4. vas_readiness_agent.py in dezelfde map zetten
5. cmd openen in die map
6. python vas_readiness_agent.py uitvoeren
7. output/readiness_reports.csv openen
8. resultaat beoordelen

Het doel van deze eerste versie is niet dat de agent al volledig geïntegreerd is met WMS of ERP. Het doel is om zichtbaar te maken dat de agent orders kan lezen, regels kan toepassen en een readiness-advies kan genereren.

Bijlage 1. Template: Agent-charter

Naam agent

[Bijvoorbeeld: Klantvraag-agent / Offerte-agent / Factuurcontrole-agent]

Proces

[Welk proces ondersteunt de agent? Bijvoorbeeld: afhandeling van standaard klantvragen via e-mail.]

Doel

De agent ondersteunt medewerkers door [taak] voor te bereiden, zodat [gewenst resultaat] wordt bereikt.

Voorbeeld:

De agent ondersteunt customer service door conceptantwoorden te maken op standaard klantvragen, zodat klanten sneller en consistentere antwoorden krijgen.

Scope: wat doet de agent wel?

De agent mag:

Onderdeel Beschrijving

Taak 1	[Bijvoorbeeld: klantvraag lezen en classificeren]
Taak 2	[Bijvoorbeeld: relevante informatie zoeken in de kennisbasis]
Taak 3	[Bijvoorbeeld: conceptantwoord opstellen]
Taak 4	[Bijvoorbeeld: ontbrekende informatie of risico's markeren]

Buiten scope: wat doet de agent niet?

De agent mag niet:

Beperking	Toelichting
[Bijvoorbeeld: e-mails zelfstandig verzenden]	Altijd goedkeuring door medewerker nodig
[Bijvoorbeeld: prijsafspraken wijzigen]	Alleen sales/management mag dit doen
[Bijvoorbeeld: juridische uitspraken doen]	Altijd escaleren
[Bijvoorbeeld: klantcompensatie toezeggen]	Alleen na menselijke goedkeuring

Input

De agent gebruikt de volgende input:

Inputbron	Voorbeeld
Binnenkomende informatie	[E-mail, formulier, klantvraag, factuur, offerteaanvraag]
Interne gegevens	[Ordernummer, klantnaam, projectnummer, openstaande posten]
Kennisbasis	[FAQ, voorwaarden, prijslijst, procedures, voorbeeldmails]
Historische voorbeelden	[Eerdere antwoorden, oude offertes, afgeronde cases]

Output

De agent levert de volgende output op:

Output	Beschrijving
Samenvatting	Korte samenvatting van de vraag of taak
Classificatie	Categorie, urgentie en type verzoek
Conceptresultaat	Conceptmail, conceptofferte, controlelijst of actielijst
Bronnen	Welke documenten of regels zijn gebruikt
Onzekerheden	Wat ontbreekt of moet gecontroleerd worden
Advies	Voorgestelde vervolgstap voor de medewerker

Regels voor de agent

De agent moet:

1. Alleen werken met goedgekeurde bronnen uit de kennisbasis.
2. Geen informatie verzinnen wanneer gegevens ontbreken.
3. Onzekerheden expliciet benoemen.
4. Altijd aangeven welke bron is gebruikt.
5. Afwijkingen en risico's markeren.
6. Geen definitieve toezegging doen zonder menselijke goedkeuring.
7. De afgesproken toon en structuur gebruiken.
8. De taak stoppen en escaleren als de situatie buiten scope valt.

Uitzonderingen en escalaties

De agent moet escaleren wanneer:

Situatie	Escaleren naar
Klacht of boos taalgebruik	[Customer service lead / eigenaar]
Juridische vraag	[Management / juridisch adviseur]
Afwijkende prijsafspraken	[Salesmanager]
Ontbrekende of tegenstrijdige informatie	[Procesverantwoordelijke]
Bedrag boven [€ bedrag]	[Finance / management]
Gevoelige persoonsgegevens	[Privacyverantwoordelijke]
Onbekende vraagcategorie	[Medewerker ter beoordeling]

Verantwoordelijke gebruiker

De verantwoordelijke gebruiker is:

[Naam / functie]

Deze persoon is verantwoordelijk voor:

Verantwoordelijkheid Toelichting

Controle van output	Beoordeelt concepten voordat actie wordt ondernomen
Goedkeuring	Keurt verzending, wijziging of vervolgstap goed

Verantwoordelijkheid Toelichting

Feedback	Geeft aan wat de agent fout of goed deed
Kennisbeheer	Signaleert ontbrekende of verouderde kennis
Escalatie	Bepaalt wat met uitzonderingen gebeurt

Goedkeuringsmodel

De agent werkt op niveau:

- Niveau 1: voorbereiden, mens beslist
- Niveau 2: klaarzetten, mens keurt goed
- Niveau 3: zelfstandig uitvoeren binnen harde regels

Voor eerste MKB-toepassingen is meestal niveau 1 passend.

Succescriteria

De agent is succesvol wanneer:

KPI	Doel
Tijd per taak	[Bijvoorbeeld: 30% minder verwerkingstijd]
Reactietijd	[Bijvoorbeeld: klantvragen binnen 4 uur voorbereid]
Bruikbaarheid output	[Bijvoorbeeld: 80% van concepten bruikbaar na kleine correctie]
Fouten	[Bijvoorbeeld: minder correcties of gemiste informatie]
Escalaties	[Bijvoorbeeld: uitzonderingen worden correct herkend]

Startdatum pilot

[Datum]

Evaluatiemoment

[Datum, bijvoorbeeld na 4 weken]

Eigenaar agent-charter

[Naam / functie]

Bijlage 2. Template: gecontroleerde kennisbasis

Naam kennisbasis

[Bijvoorbeeld: Kennisbasis Klantvraag-agent]

Doel van de kennisbasis

Deze kennisbasis bevat alleen goedgekeurde informatie die de agent mag gebruiken om [taak] voor te bereiden.

Voorbeeld:

Deze kennisbasis bevat goedgekeurde informatie die de Klantvraag-agent mag gebruiken om conceptantwoorden te maken op standaardvragen over levertijd, retouren, garantie, facturen en productinformatie.

1. Kennisregister

ID	Document / bron	Type kennis	Eigenaar	Versie / datum	Status	Mag agent gebruiken?
KB-001	Algemene voorwaarden	Juridisch / klantafspraken	Directie	01-03-2026	Goedgekeurd	Ja
KB-002	Retourbeleid	Procedure	Customer service	15-02-2026	Goedgekeurd	Ja
KB-003	Prijslijst 2026	Prijzen	Sales	01-01-2026	Goedgekeurd	Ja
KB-004	Productinformatie standaardlijn	Productkennis	Operations	20-02-2026	Goedgekeurd	Ja
KB-005	Voorbeeldmails klantvragen	Communicatievoorbeelden	Customer service	10-03-2026	Goedgekeurd	Ja
KB-006	Oude prijslijst 2025	Prijzen	Sales	01-01-2025	Verlopen	Nee
KB-007	Interne notities medewerker	Informeel kennis	Onbekend	Onbekend	Niet gevalideerd	Nee

De agent mag alleen bronnen gebruiken met status "Goedgekeurd" en "Mag agent gebruiken = Ja".

2. Toegestane kenniscategorieën

Categorie	Mag gebruiken?	Toelichting
Goedgekeurde procedures	Ja	Bijvoorbeeld retourproces, garantieproces, factuurproces
Actuele prijslijsten	Ja	Alleen laatste goedgekeurde versie
Algemene voorwaarden	Ja	Alleen officiële versie
Productinformatie	Ja	Alleen gecontroleerde productdocumentatie
Voorbeeldmails	Ja	Alleen voorbeelden die als goed zijn beoordeeld
Oude e-mails	Beperkt	Alleen als voorbeeld, niet als harde regel
Persoonlijke notities	Nee	Te onzeker en vaak niet actueel
Verouderde documenten	Nee	Risico op verkeerde informatie
Mondelinge afspraken	Nee	Alleen gebruiken nadat ze schriftelijk zijn vastgelegd

3. Kenniskaarten

Naast documenten is het nuttig om korte kenniskaarten te maken. Die maken regels concreet voor de agent.

Voorbeeld kenniskaart 1: levertijdvraag

Veld	Inhoud
ID	KK-001
Onderwerp	Klant vraagt naar levertijd
Wanneer gebruiken	Als klant vraagt wanneer bestelling wordt geleverd
Antwoordregel	Gebruik alleen bevestigde orderstatus. Doe geen harde leverbelofte als datum niet bevestigd is.

Veld	Inhoud
Benodigde input	Ordernummer, klantnaam, actuele orderstatus
Standaardantwoord	“Wij hebben uw bestelling gecontroleerd. De verwachte leverdatum is [datum], op basis van de huidige orderstatus.”
Niet doen	Geen exacte datum noemen als die niet bevestigd is
Escaleren bij	Spoedlevering, boze klant, ontbrekend ordernummer, vertraging zonder duidelijke oorzaak
Bron	Leveringsprocedure KB-008
Eigenaar	Operations
Laatst gecontroleerd	01-03-2026

Voorbeeld kenniskaart 2: retourvraag

Veld	Inhoud
ID	KK-002
Onderwerp	Klant vraagt om retour
Wanneer gebruiken	Als klant vraagt of een product retour mag
Antwoordregel	Controleer retourtermijn, productconditie en aankoopbewijs
Benodigde input	Aankoopdatum, ordernummer, productstatus
Standaardantwoord	“U kunt een retour aanvragen wanneer het product binnen de retourtermijn valt en voldoet aan onze retourvoorwaarden.”
Niet doen	Geen retour goedkeuren zonder controle van aankoopdatum en conditie
Escaleren bij	Beschadigd product, klacht, maatwerkproduct, verlopen retourtermijn
Bron	Retourbeleid KB-002

Veld	Inhoud
Eigenaar	Customer service
Laatst gecontroleerd	15-02-2026

Voorbeeld kenniskaart 3: prijs- of kortingsvraag

Veld	Inhoud
ID	KK-003
Onderwerp	Klant vraagt korting of afwijkende prijs
Wanneer gebruiken	Als klant vraagt om korting, aangepaste prijs of betere voorwaarden
Antwoordregel	Agent mag geen korting toezeggen. Alleen standaardprijs noemen op basis van actuele prijslijst.
Benodigde input	Product, aantal, klanttype, actuele prijslijst
Standaardantwoord	“Op basis van onze actuele prijslijst bedraagt de prijs [prijs]. Afwijkende prijsafspraken worden apart beoordeeld.”
Niet doen	Geen korting, compensatie of aangepaste voorwaarden toezeggen
Escaleren bij	Grote order, bestaande raamafpraak, klacht, concurrentieaanbod
Bron	Prijslijst KB-003
Eigenaar	Sales
Laatst gecontroleerd	01-01-2026

4. Bronhiërarchie

Niet alle informatie is even zwaar. De agent moet weten welke bron voorrang heeft als bronnen elkaar tegenspreken.

Rang	Bronsoort	Voorbeeld	Gewicht
1	Contract / algemene voorwaarden	Getekende klantafpraak	Hoogste
2	Actuele prijslijst / procedure	Goedgekeurde versie 2026	Hoog
3	Productdocumentatie	Handleiding, specificatieblad	Hoog
4	Kenniskaart	Samenvatting van goedgekeurde regels	Middel
5	Voorbeeldmail	Toon en structuur	Laag
6	Oude e-mail	Alleen context, geen regel	Zeer laag

Regel voor de agent: bij conflict tussen bronnen altijd de hoogste bron gebruiken en de tegenstrijdigheid melden.

5. Verboden kennis

Leg ook vast wat de agent niet mag gebruiken.

Verboden bron	Waarom verboden
Verouderde prijslijsten	Kans op verkeerde prijzen
Conceptprocedures	Nog niet goedgekeurd
Persoonlijke notities	Niet gecontroleerd
Oude klantmails met uitzonderingen	Kunnen oude of individuele afspraken bevatten
Mondelinge afspraken	Niet verifieerbaar
Internetbronnen zonder goedkeuring	Risico op onjuiste of irrelevante informatie
Gevoelige persoonsgegevens	Alleen gebruiken als dit noodzakelijk en toegestaan is

6. Escalatie-informatie

De kennisbasis moet ook aangeven wie verantwoordelijk is bij twijfel.

Onderwerp	Escaleren naar	Wanneer
Prijsafwijking	Salesmanager	Bij korting, maatwerkprijs of raamafpraak
Juridische vraag	Directie / juridisch adviseur	Bij aansprakelijkheid, schadeclaim of contractgeschil
Betalingsvraag	Finance	Bij openstaande posten of betalingsregeling
Leveringsprobleem	Operations	Bij vertraging, spoed of ontbrekende voorraad
Klacht	Customer service lead	Bij ontevreden of boze klant
Producttechniek	Productspecialist	Bij technische vraag buiten standaarddocumentatie

7. Update- en beheerregels

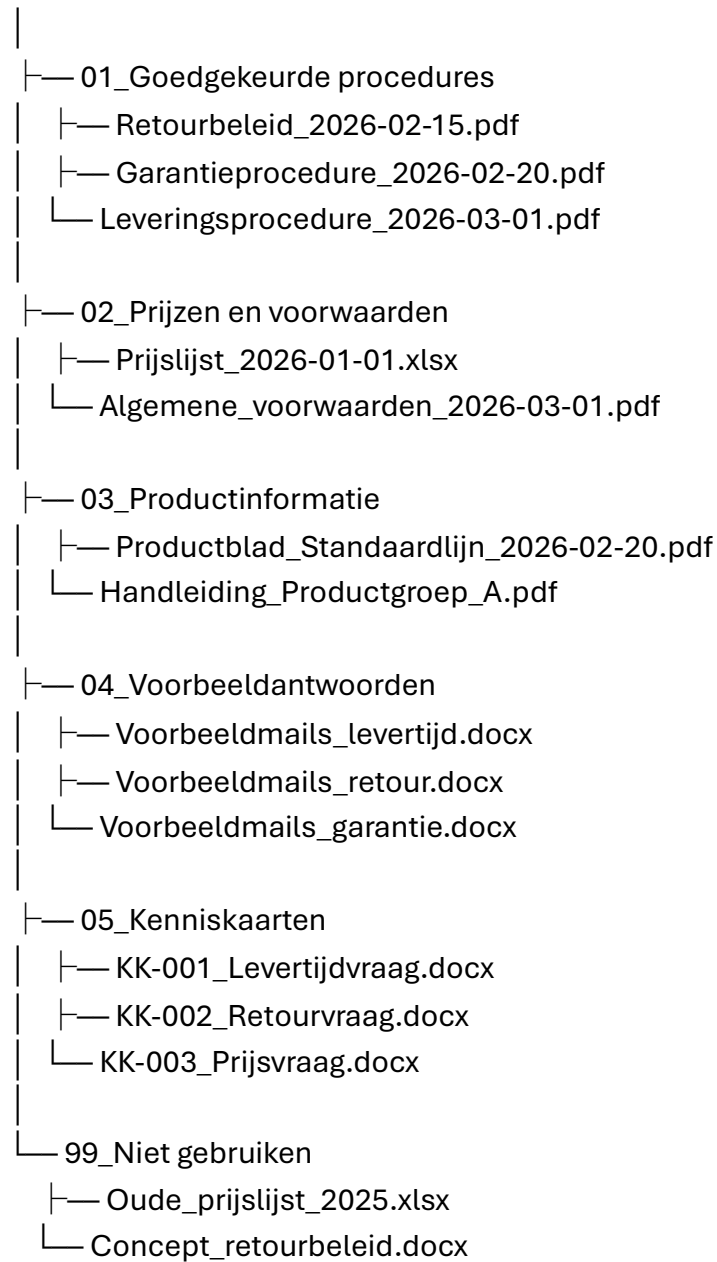
Een kennisbasis blijft alleen gecontroleerd als iemand beheer voert.

Regel	Toelichting
Elke bron heeft een eigenaar	Zonder eigenaar geen opname in kennisbasis
Elke bron heeft een datum	De agent moet weten of informatie actueel is
Elke bron heeft een status	Concept, goedgekeurd, verlopen of geblokkeerd
Verlopen bronnen worden uitgesloten	Agent mag ze niet gebruiken
Wijzigingen worden gelogd	Zichtbaar wat is aangepast en door wie
Periodieke review	Bijvoorbeeld maandelijks of per kwartaal
Feedback uit gebruik wordt verwerkt	Veelvoorkomende fouten leiden tot betere kenniskaarten

8. Minimale mapstructuur

Een praktische mapstructuur kan er zo uitzien:

Kennisbasis Klantvraag-agent



9. Voorbeeld van agent-instructie bij gebruik van de kennisbasis

De agent krijgt bijvoorbeeld deze instructie:

“Gebruik alleen documenten en kenniskaarten met status ‘Goedgekeurd’. Gebruik geen bronnen uit de map ‘Niet gebruiken’. Wanneer informatie ontbreekt, verzin niets, maar vermeld welke informatie ontbreekt. Wanneer bronnen elkaar tegenspreken, gebruik de bron met de hoogste rang volgens de bronhiërarchie en markeer de tegenstrijdigheid. Geef bij elk conceptantwoord aan welke bron is gebruikt.”

Bijlage 3. Template: controlemodel voor een MKB-agent

Naam controlemodel

Controlemodel voor [naam agent]

Proces

[Bijvoorbeeld: klantvragen via e-mail / offertevoorbereiding / factuurcontrole]

Doel van het controlemodel

Dit controlemodel beschrijft welke acties de agent mag voorbereiden, welke acties verboden zijn, wanneer escalatie nodig is, wie goedkeurt en welke handelingen worden vastgelegd.

1. Rechten van de agent

Leg concreet vast wat de agent wél mag doen. Begin conservatief. Voor een eerste MKB-agent is “voorbereiden” meestal voldoende.

Actie	Mag de agent dit doen?	Voorwaarde
Inkomende informatie lezen	Ja	Alleen binnen het gekozen proces
Vraag of document classificeren	Ja	Categorie moet uit vaste lijst komen
Relevante kennisbron zoeken	Ja	Alleen in goedgekeurde kennisbasis
Samenvatting maken	Ja	Moet feitelijk en controleerbaar zijn
Conceptantwoord maken	Ja	Niet zelfstandig verzenden
Conceptofferte voorbereiden	Ja	Alleen standaardproducten en actuele prijslijst
Ontbrekende informatie signaleren	Ja	Duidelijk vermelden wat ontbreekt
Risico of afwijking markeren	Ja	Met reden en bronverwijzing
Taak klaarzetten voor medewerker	Ja	Status: “controle nodig”

Actie	Mag de agent dit doen?	Voorwaarde
Mail verzenden	Nee	Alleen na menselijke goedkeuring
Prijs wijzigen	Nee	Alleen bevoegde medewerker
Betaling uitvoeren	Nee	Nooit door agent in pilotfase
Klantcompensatie toezeggen	Nee	Altijd menselijke goedkeuring
Contract aanpassen	Nee	Altijd buiten scope

Concrete formulering:

“De agent mag informatie lezen, ordenen, samenvatten, classificeren en concepten voorbereiden. De agent mag geen externe communicatie verzenden, geen financiële handelingen uitvoeren en geen bindende toezeggingen doen zonder menselijke goedkeuring.”

2. Beperkingen van de agent

Beperkingen zijn harde grenzen. Ze moeten duidelijker zijn dan algemene richtlijnen.

Beperking	Concrete regel
Geen zelfstandige verzending	De agent mag geen e-mail, offerte of bericht automatisch naar klanten sturen
Geen prijsafwijkingen	De agent mag alleen actuele standaardprijzen gebruiken
Geen juridische beoordeling	Juridische vragen worden samengevat en geëscaleerd
Geen verzonnen informatie	Als informatie ontbreekt, moet de agent dit melden
Geen gebruik van verouderde bronnen	Alleen bronnen met status “goedgekeurd”

Beperking	Concrete regel
Geen verwerking buiten scope	Onbekende categorieën worden doorgestuurd naar medewerker
Geen gevoelige beslissingen	Geen besluiten over krediet, ontslag, claims, contracten of compensatie
Geen systeemmutaties zonder goedkeuring	De agent mag niets definitief wijzigen in CRM, ERP of boekhouding

Gebruik ook een “stopregel”:

“Als de agent onvoldoende zekerheid heeft, tegenstrijdige informatie vindt of de situatie buiten de afgesproken scope valt, stopt de agent en markeert hij de taak als ‘menselijke beoordeling nodig’.”

3. Autonomie niveaus

Bepaal per taak hoeveel vrijheid de agent heeft.

Niveau	Omschrijving	Toepassing
Niveau 0	Alleen signaleren	Agent meldt afwijking of ontbrekende informatie
Niveau 1	Voorbereiden	Agent maakt concept, mens beslist
Niveau 2	Klaarzetten	Agent vult velden of maakt taak klaar, mens keurt goed
Niveau 3	Zelfstandig uitvoeren	Agent voert actie uit binnen harde regels

Voor de eerste MKB-pilot geldt meestal:

Gekozen niveau: Niveau 1 — voorbereiden, mens beslist.

Voorbeeld:

Taak	Autonomieniveau
Klantvraag samenvatten	Niveau 1
Conceptantwoord maken	Niveau 1
Mail klaarzetten als concept	Niveau 2
Mail verzenden	Niet toegestaan
Factuurafwijking signaleren	Niveau 1
Factuur goedkeuren	Niet toegestaan
Offerte voorbereiden	Niveau 1
Offerte verzenden	Niet toegestaan

4. Escalatieregels

Escalatieregels bepalen wanneer de agent niet verder mag. Maak ze zo concreet mogelijk.

Situatie	Agent actie	Escaleren naar
Klacht of boos taalgebruik	Samenvatten, geen antwoord verzenden	Customer service lead
Juridische termen zoals aansprakelijkheid, schade, ingebrekestelling	Markeren als juridisch risico	Directie / juridisch adviseur
Prijsafwijking of kortingsverzoek	Standaardprijs tonen, afwijking markeren	Salesmanager
Bedrag boven afgesproken grens	Concept maken, goedkeuring vragen	Finance / directie

Situatie	Agent actie	Escaleren naar
Ontbrekende informatie	Lijst ontbrekende gegevens maken	Verantwoordelijke medewerker
Tegenstrijdige bronnen	Hoogste bronhiërarchie gebruiken en conflict melden	Proces verantwoordelijke
Onbekende vraagcategorie	Geen antwoord maken of alleen samenvatten	Medewerker
Gevoelige persoonsgegevens	Verwerking beperken en markeren	Privacy verantwoordelijke
Boze of dreigende klant	Geen standaardantwoord sturen	Manager
Spoed of leveringsrisico	Prioriteit hoog markeren	Operations
Mogelijke reputatieschade	Stoppen en escaleren	Directie

Voorbeeldregel:

“Bij klachten, juridische vragen, prijsafwijkingen, ontbrekende kerninformatie of bedragen boven €1.000 mag de agent alleen samenvatten en een advies voor vervolgstap geven. De definitieve actie ligt bij de verantwoordelijke gebruiker.”

5. Goedkeuringsstappen

Leg vast wie de output controleert en wanneer goedkeuring nodig is.

Stap	Actie	Uitvoerder	Status
1	Agent leest input en maakt samenvatting	Agent	Concept
2	Agent zoekt relevante kennisbron	Agent	Concept
3	Agent maakt conceptoutput	Agent	Controle nodig

Stap	Actie	Uitvoerder	Status
4	Medewerker controleert inhoud, bronnen en toon	Verantwoordelijke gebruiker	Goedgekeurd / aanpassen / afwijzen
5	Medewerker voert actie uit of verzendt bericht	Medewerker	Uitgevoerd
6	Feedback wordt vastgelegd	Medewerker	Verbeterinput

Goedkeuringsopties:

Optie	Betekenis
Goedgekeurd	Output kan worden gebruikt
Aanpassen	Output is deels bruikbaar, medewerker corrigeert
Afgewezen	Output is onjuist of onbruikbaar
Escaleren	Specialist of manager moet beoordelen
Kennisbasis aanpassen	Bron ontbreekt of is onduidelijk

Maak de controle praktisch. De medewerker moet snel kunnen zien:

Wat heeft de agent geconcludeerd? Welke bron is gebruikt? Welke onzekerheden zijn er?
Wat moet ik goedkeuren? Wat gebeurt er daarna?

Voorbeeld controleblok bij een conceptmail:

Controlepunt	Inhoud
Vraagcategorie	Retourvraag
Gebruikte bron	Retourbeleid KB-002, versie 15-02-2026
Conceptantwoord	Klaargezet
Onzekerheid	Aankoopdatum ontbreekt
Agentadvies	Vraag aankoopdatum op voordat retour wordt bevestigd
Vereiste actie medewerker	Controleren en eventueel verzenden

6. Logging

Logging zorgt voor traceerbaarheid. Dit is belangrijk bij fouten, verbetering en vertrouwen.

Leg minimaal vast:

Logveld	Voorbeeld
Datum en tijd	05-05-2026, 10:42
Taak-ID	T-2026-00124
Type taak	Klantvraag retour
Inputbron	Mail klant / formulier / factuur
Gebruikte kennisbron	KB-002 Retourbeleid
Agentactie	Conceptantwoord gemaakt
Zekerheid / onzekerheid	Onzeker: aankoopdatum ontbreekt

Logveld	Voorbeeld
Escalatie nodig	Ja / Nee
Reden escalatie	Ontbrekende informatie
Medewerkerbesluit	Aangepast en verzonden
Correcties	Medewerker voegde aankoopdatum toe
Status	Afgerond
Feedback	Concept bruikbaar, bron correct

Voorbeeld logboek:

Tijd	Taak	Agentoutput	Bron	Escalatie	Menselijke actie	Status
10:42	Retourvraag klant A	Conceptantwoord	KB-002	Ja, aankoopdatum ontbreekt	Medewerker vraagt datum op	Open
11:15	Levertijdvraag klant B	Conceptantwoord	KB-008	Nee	Verzonden na controle	Afgerond
12:05	Kortingsvraag klant C	Standaardprijs + escalatie	KB-003	Ja, kortingsverzoek	Naar salesmanager	Open

De agent moet bij elke output een korte “auditregel” tonen:

“Gebaseerd op KB-002 Retourbeleid, versie 15-02-2026. Geen retour bevestigd omdat aankoopdatum ontbreekt. Menselijke controle vereist.”

7. Rollen en verantwoordelijkheden

Een controlemodel werkt alleen als duidelijk is wie waarvoor verantwoordelijk is.

Rol	Verantwoordelijkheid
Agentgebruiker	Controleert dagelijkse output
Proceseigenaar	Bepaalt scope, regels en escalaties
Kennis-eigenaar	Houdt documenten actueel
Manager	Keurt uitzonderingen en risicovolle acties goed
IT / systeembeheer	Beheert toegang en technische instellingen
Privacyverantwoordelijke	Controleert verwerking van gevoelige gegevens

Voorbeeld:

Onderwerp	Eigenaar
Klantantwoord goedkeuren	Customer service medewerker
Prijsafwijking goedkeuren	Salesmanager
Retourbeleid beheren	Customer service lead
Algemene voorwaarden beheren	Directie
Factuurafwijking beoordelen	Finance
Agentinstellingen wijzigen	Proceseigenaar + IT

8. Toegangsrechten

Beperk wat de agent kan zien en gebruiken.

Gegevenssoort	Toegang agent	Toelichting
Goedgekeurde kennisbasis	Ja	Nodig voor taak
Algemene klantgegevens	Ja, beperkt	Alleen relevante gegevens
Orderstatus	Ja	Alleen lezen
Prijslijst	Ja	Alleen actuele versie
CRM-notities	Beperkt	Alleen relevante notities
Bankgegevens	Nee, tenzij strikt noodzakelijk	Hoog risico
HR-gegevens	Nee	Buiten scope
Persoonlijke mailboxen	Nee	Alleen gedeelde procesmailbox
Boekhoudsysteem	Alleen lezen / beperkt	Geen betalingen of boekingen

Regel:

“De agent krijgt minimale toegang: alleen de informatie die nodig is voor de afgesproken taak.”

9. Foutafhandeling

Leg vast wat gebeurt als de agent een fout maakt.

Fouttype	Actie
Onjuiste bron gebruikt	Output afwijzen, brongebruik corrigeren
Verouderde informatie gebruikt	Kennisbasis aanpassen, document blokkeren
Verkeerde classificatie	Feedback registreren, categorie-instructie aanpassen
Onjuist conceptantwoord	Medewerker corrigeert, fout loggen
Escalatie gemist	Regel aanscherpen, voorbeeld toevoegen
Te veel escalaties	Scope of kennisbasis verbeteren
Onjuiste systeemactie	Direct blokkeren, rechten beperken

Belangrijk: fouten zijn verbeterinput. In de pilotfase moet elke fout worden gebruikt om de kennisbasis, prompt of escalatieregels te verbeteren.

10. Concrete output van stap 5

Na stap 5 moet er een werkbaar controlemodel liggen met deze onderdelen:

Onderdeel	Concreet resultaat
Rechten	Lijst van acties die de agent wel mag uitvoeren
Beperkingen	Harde grenzen en verboden acties
Autonomieniveau	Duidelijk niveau per taak
Escalatieregels	Wanneer de agent moet stoppen of doorzetten
Goedkeuringsstappen	Wie controleert, keurt goed of wijst af
Logging	Wat wordt vastgelegd per agentactie
Rollen	Wie is verantwoordelijk voor output, kennis en uitzonderingen
Toegang	Welke systemen en gegevens de agent mag gebruiken

Onderdeel Concreet resultaat

Foutafhandeling Wat gebeurt bij verkeerde output

Compact voorbeeld: controlemodel klantvraag-agent

Agent: Klantvraag-agent
Proces: Afhandeling standaard klantvragen
Autonomieniveau: Niveau 1 — voorbereiden, mens beslist

Onderdeel	Concrete afspraak
Rechten	Mail lezen, vraag classificeren, kennisbron zoeken, conceptantwoord maken
Beperkingen	Niet verzenden, geen korting toezeggen, geen juridische antwoorden geven
Escalatie	Klacht, prijsafwijking, juridische vraag, onbekende categorie, ontbrekende gegevens
Goedkeuring	Customer service medewerker controleert elk concept vóór verzending
Logging	Vraagtype, bron, concept, onzekerheden, medewerkerbesluit en correcties worden vastgelegd
Verantwoordelijke	Customer service lead
Succescontrole	Wekelijkse review van bruikbaarheid, fouten en escalaties