



PROJECT 4.0

Conceptfase

M2 - Cropwatchers

Voorwoord

Wij zijn 6 studenten Toegepaste Informatica uit verschillende keuzetrajecten, namelijk Application Development, Artificial Intelligence en Cloud & Cyber Security. Voor het OPO Project 4.0 moeten we onze krachten bundelen om een webapplicatie te maken voor Vito waarmee landbouwers hun landbouwschade kunnen doorgeven. We doen dit onder de naam Cropwatchers.

Dit document omvat de conceptfase van het project. Om te beginnen willen we graag Vito bedanken voor de ondersteuning en de duidelijke antwoorden op onze vragen. Daarnaast zouden we graag onze coach Mitch Mommers bedanken die ons begeleid heeft in deze conceptfase.

Inhoudstafel

1. INLEIDING	5
2. PERSONA'S	6
2.1. Landbouwer	6
2.2. Consultant	7
2.3. Overheidsmedewerker	7
3. EISENANALYSE	8
3.1. Functionele eisen	8
3.1.1. Use Case Diagram & Entity Relationship Diagram	8
3.1.2. Use Case beschrijvingen met prototypes	11
3.1.3. Flowdiagram use cases	51
3.1.4. Prioriteit per functionaliteit	52
3.2. Niet-functionele eisen	53
3.2.1. Implementatie	53
3.2.2. Externe interface	53
3.2.3. Prestatie	53
3.2.4. Kwaliteitsvereisten	53
4. DATA VERZAMELING	55
4.1. Analyse GeoPunt	55
4.1.1. Wat is GeoPunt Vlaanderen?	55
4.1.2. Hoe kunnen we dit gebruiken	55
4.1.3. Manueel werken met de kaart	55
4.1.4. Werken met kaartlagen	56
4.1.5. Waterinfo vlaanderen	57
4.1.6. Definities voor het begrijpen van data	57
4.1.7. Usefull links	57
4.2. Analyse OpenEO	58
4.2.1. Parameters	58
4.2.2. Anomaly detection	58
4.2.3. Variability Map	59
4.2.4. Yield Potential Map	61
4.3. Analyse kaartintegratie	63
4.3.1. Vergelijking tussen Leaflet en OpenLayer	63
4.3.2. Conclusie waarom Leaflet	63
4.3.3. Test case	64
4.3.4. React-Leaflet	66
4.4. Analyse LLM	67
4.4.1. Privacy rechten sociale media	67
4.4.2. Sociale Media data krijgen	67
4.4.3. (Betrouwbare) Twitter/X accounts	68
4.4.4. Media data	68
4.4.5. Bestaande LLM's	70
4.4.6. Data ophalen voor LLM	73
4.4.7. Formulier invullen	74
4.5. Analyse Visualisation tools	74

4.5.1. Integratie	74
4.5.2. Power BI	74
4.5.3. Python Plotly	75
4.5.4. Chart.js	76
5. DEPLOYMENT	78
5.1. FARM stack	78
5.1.1. Back-end (FastAPI)	78
5.1.2. Front-end (React)	79
5.1.3. Database (MongoDB)	80
5.2. Hosting	82
5.2.1. Introductie tot de Big 3 Cloud Providers	82
5.2.2. Vergelijking: Weighted Ranking Method (WRM)	82
5.2.3. Conclusie	83
5.3. Infrastructuur	84
5.3.1. VPC en Subnets	84
5.3.2. Internet Gateway en NAT Gateway	84
5.3.3. Routing en Beveiliging	85
5.3.4. MongoDB Atlas	85
5.3.5. Schaalbaarheid en Beschikbaarheid	85
5.3.6. Diagram	86
5.4. Kostenanalyse	87
5.4.1. VPC en Subnets	87
5.4.2. EC2 Instanties	87
5.4.3. MongoDB Atlas	87
5.4.4. Data Transfer	87
5.4.5. Productie omgeving	87
5.4.6. Conclusie	88
5.5. Gitlab CI/CD	88
5.5.1. Workflow Overzicht	88
5.5.2. Branching Strategie	89
5.5.3. CI/CD Pipelines	89
5.5.4. AWS Deployment Strategie	90
5.5.5. Stappenplan Workflow	90
5.5.6. Monitoring en Onderhoud	91
5.5.7. Conclusie en Voordelen	91
6. PLANNING	92
6.1. Week 1 – 13/01/2025	92
6.2. Week 2 – 20/01/2025	92
6.3. Week 3 – 10/02/2025	92
6.4. Week 4 – 17/02/2025	92
7. CONCLUSIE	93
8. BIBLIOGRAFIE	94

1. Inleiding

Dit document legt onze visie uit die is voortgekomen hoe wij de opdracht van Vito hebben begrepen en wat we allemaal gaan nodig hebben om daadwerkelijk een innovatief systeem te ontwikkelen dat landbouwers, consultants en overheidsmedewerkers helpt bij het nemen en ondersteunen van data gedreven beslissingen.

We beginnen met het voorstellen van de belangrijkste gebruikers van het systeem. Door middel van persona's brengen we de behoeften en verwachtingen van landbouwers, consultants en overheidsmedewerkers in kaart. Vervolgens duiken we in de eisen die aan het systeem worden gesteld. Hier maken we onderscheid tussen functionele en niet-functionele eisen. Deze stap vormt de fundering voor het ontwerp van een gebruiksvriendelijke en schaalbare oplossing.

Daarna verkennen we de wereld van data-analyse. We laten zien hoe we bronnen zoals GeoPunt Vlaanderen, OpenEO en sociale media benutten om een robuust dataplatform te creëren. De toepassing van geavanceerde visualisatietools zoals Power BI, Python Plotly en Chart.js maakt het mogelijk om complexe gegevens op een intuïtieve manier te interpreteren.

De technische kant van het project krijgt ook volop aandacht. We introduceren de FARM-stack (FastAPI, React, MongoDB) waarmee zowel de back-end als de front-end van de applicatie wordt gerealiseerd. We nemen je mee door het proces van de systeemimplementatie, inclusief het opzetten van de infrastructuur, hosting en continue integratie/continue levering (CI/CD) via GitLab. Met een duidelijk deploymentplan waarborgen we een soepele uitrol en onderhoud van het systeem.

Tenslotte staan we stil bij de kostenkant van het project. We vergelijken verschillende cloudproviders en onderbouwen onze keuze aan de hand van een Weighted Ranking Method (WRM). Door deze methodiek te volgen, bieden we inzicht in de kosten, schaalbaarheid en beschikbaarheid van de oplossing.

2. Persona's

In dit deel worden de belangrijkste gebruikers van de applicatie beschreven aan de hand van persona's. Een persona vertegenwoordigt een typische gebruiker van het systeem met een (denkbeeldige) naam en leeftijd. Vervolgens is er het profiel van de persona, dat aangeeft welke rol deze gebruiker vervult binnen het systeem.

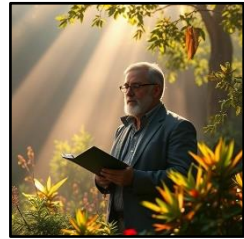
Vervolgens worden de IT-behoefte en vaardigheden van de persona beschreven: wat de gebruiker al kan op IT-gebied en wat hij of zij zou moeten kunnen met het systeem. Ten slotte wordt besproken welke mogelijke uitdagingen de persona kan ondervinden bij het gebruik van het systeem. Deze aanpak helpt om de behoeften en knelpunten van de belangrijkste gebruikers goed te begrijpen en hier rekening mee te houden bij de ontwikkeling van het systeem.

2.1. Landbouwer

- **Naam:** Jan De Boer
- **Leeftijd:** 29 jaar
- **Profiel:** Een zelfstandige boer die dagelijks werkt op het land.
- **IT-behoefte en vaardigheden:**
 - **Behoeftte aan eenvoud en mobiliteit:** Hij wil technologie die gemakkelijk te begrijpen en te bedienen is, bij voorkeur op mobiele apparaten. Het gebruik moet praktisch en snel zijn, zonder uitgebreide training.
 - **Interesse in agritech-oplossingen:** Hij zoekt tools die kunnen helpen bij het verzamelen van bewijsmateriaal voor een schadeclaim te ondersteunen.
- **Uitdagingen:** Hij heeft beperkte tijd om te investeren in nieuwe IT-tools, beperkte toegang tot internet in landelijke gebieden en mogelijk wantrouwen tegenover complexe technologie.



2.2. Consultant



- **Naam:** Mark Cools
- **Leeftijd:** 54 jaar
- **Profiel:** Een specialist in landbouw met een adviserende rol. Hij werkt samen met de landbouwers om de productie en efficiëntie te verbeteren.
- **IT-behoefte en vaardigheden:**
 - **Hoge behoefte aan data-inzicht:** Hij zoekt naar gedetailleerde data-analyse, historische trends, bodem- en weersinformatie en planningssoftware om klanten beter te adviseren.
 - **Multichannel communicatie:** Hij gebruikt verschillende platformen om met klanten te communiceren en rapporten en aanbevelingen te delen.
- **Uitdagingen:** Hij moet toegang hebben tot informatie van meerdere klanten met beveiliging van klantgegevens, en consistentie in rapportage voor duidelijkheid

2.3. Overheidsmedewerker



- **Naam:** Veronique Bosmans
- **Leeftijd:** 37 jaar
- **Profiel:** Ze werkt binnen de overheid aan landbouw- en milieubeleid, subsidies en controle van regelgeving. Ze heeft een overzichtspositie en vaak invloed op beleidsvorming.
- **IT-behoefte en vaardigheden:**
 - **Gewend aan robuuste dataverzamelingssystemen:** Zij werkt met systemen die landelijke data over landbouw, biodiversiteit en bodemgebruik opslaan en analyseren.
 - **Toegang tot rapporten en dashboards:** Zij wil makkelijk toegang tot inzichtelijke dashboards voor monitoring van landbouwvelden om snel beleidsbeslissingen te ondersteunen.
- **Uitdagingen:** Het waarborgen van gegevensbeveiliging en privacy, het werken en het beheren van grote hoeveelheden data op efficiënte wijze.

3. Eisenanalyse

In dit onderdeel worden de eisen van de applicatie besproken. Eerst worden de functionele eisen besproken. Dit gebeurt aan de hand van een Use Case Diagram en Entity Relationship Diagram, gevolgd door de Use Case Descriptions. Tot slot worden eveneens de niet-functionele eisen besproken.

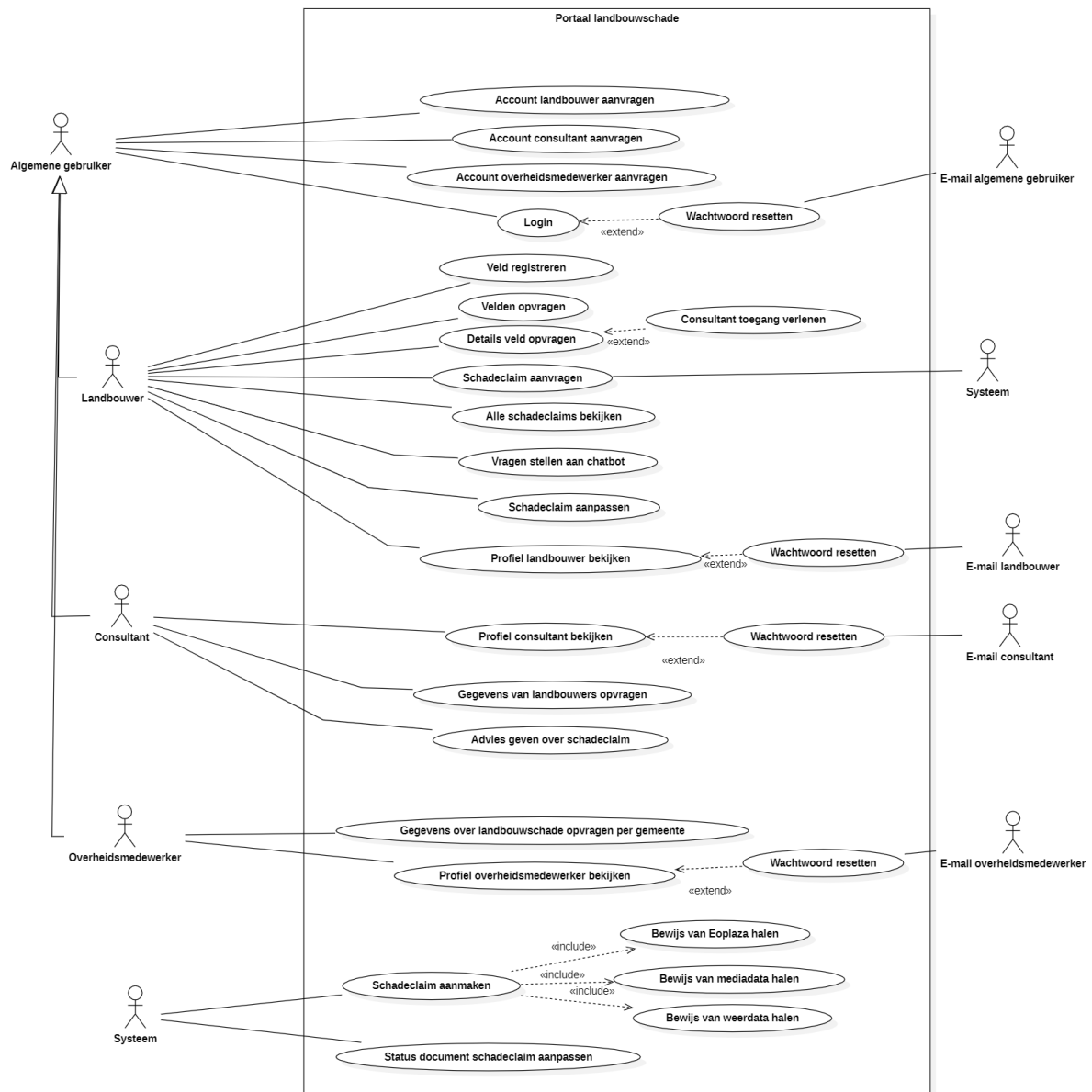
3.1. Functionele eisen

Met de functionele eisen, wordt er bedoeld wat het systeem moet kunnen volgens Vito. Dit wordt weergegeven in een Use Case Diagram samen met het bijhorende Entity Relationship Diagram en Use Case descriptions.

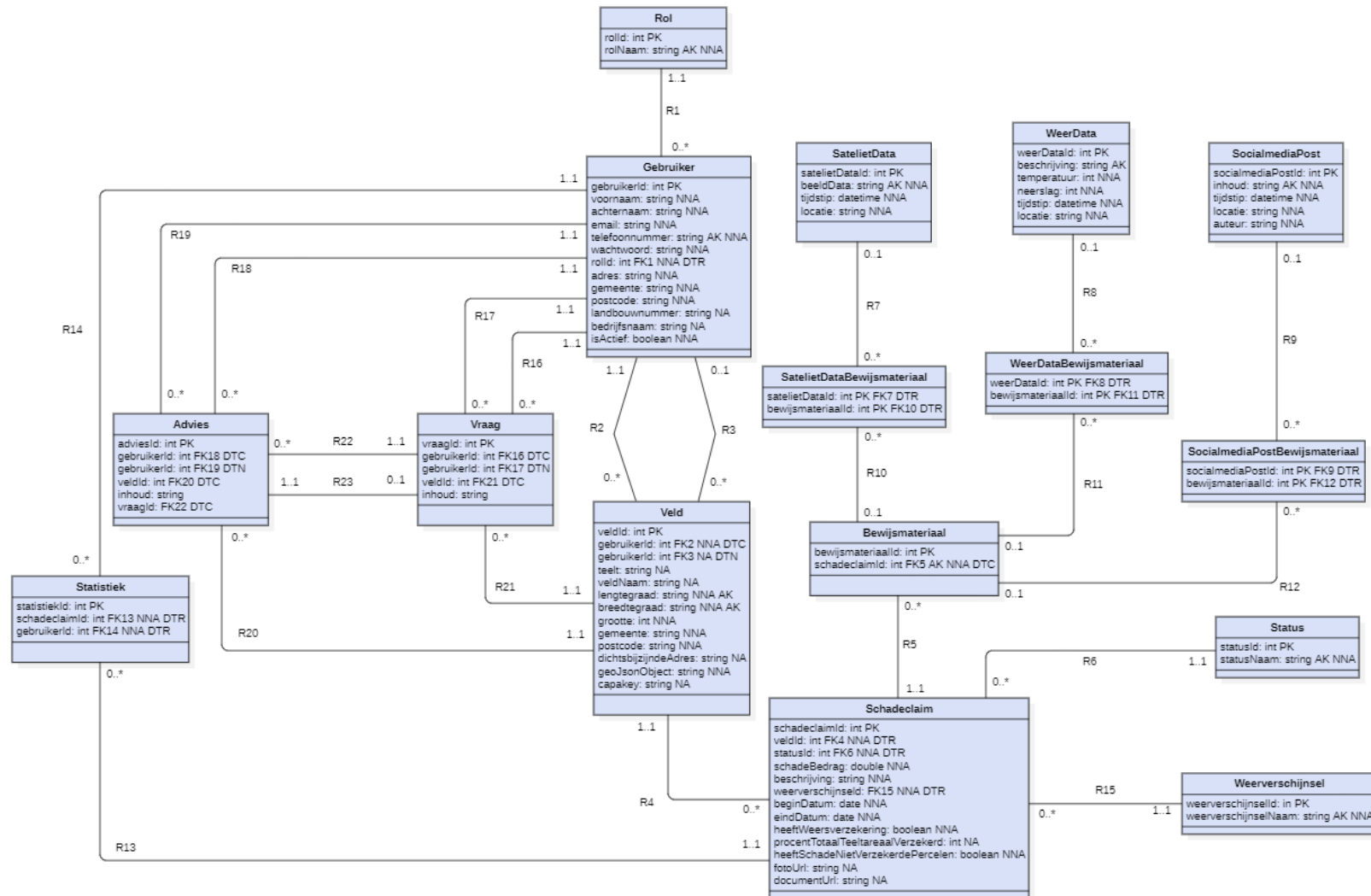
3.1.1. Use Case Diagram & Entity Relationship Diagram

Dit is het Use Case Diagram (UCD) en Entity Relationship Diagram (ERD) dat Cropwatchers heeft uitgewerkt. Er wordt gebruik gemaakt van vier soorten gebruikers, namelijk: de algemene gebruiker, de landbouwer, de consultant en de overheidsmedewerker. In dit UCD wordt voorgesteld wat elke gebruiker zou moeten kunnen doen in het systeem. In het ERD wordt daartegen weergegeven welke gegevens Cropwatchers allemaal gaat bijhouden en beheren.

3.1.1.1. Use Case Diagram



3.1.1.2. Entity Relationship Diagram



3.1.2. Use Case beschrijvingen met protoypes

3.1.2.1. Homepagina algemene gebruiker

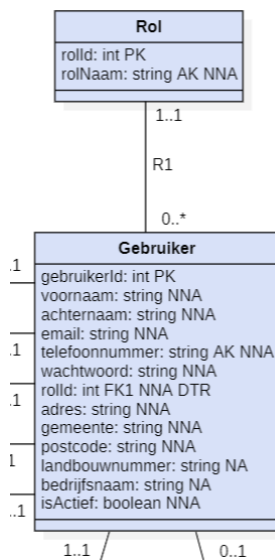
Scherf(en):

CW1_homepage_algemene_gebruiker



Action	Explanation
Account aanvragen	Gaat naar CW5_account_aanvragen.
Login	Gaat naar use case "Login".

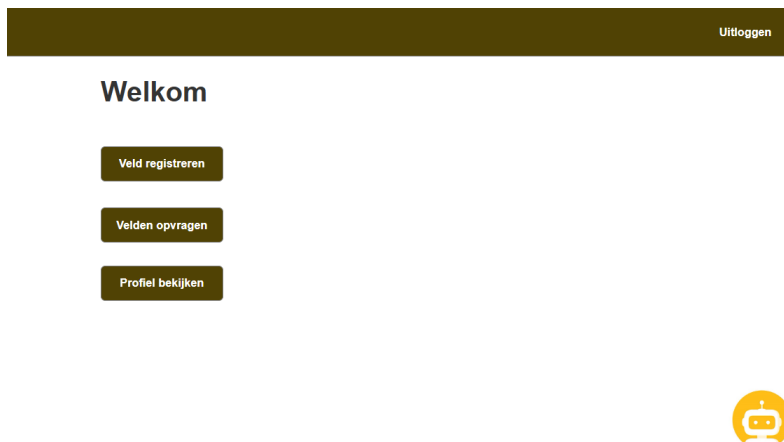
Betrokken tabellen ERD:




3.1.2.2. Homepagina landbouwer

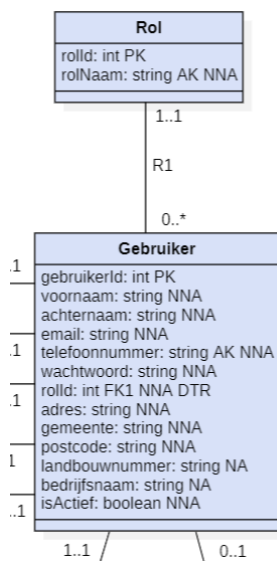
Scherm(en):

CW2_homepage_landbouwer



Action	Explanation
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Veld registreren	Gaat naar use case "Veld registreren".
Velden opvragen	Gaat naar use case "Velden opvragen".
Profiel bekijken	Gaat naar use case "Profiel landbouwer bekijken".
	Gaat naar use case "Vragen stellen aan chatbot".

Betrokken tabellen ERD:



3.1.2.3. Homepagina consultant

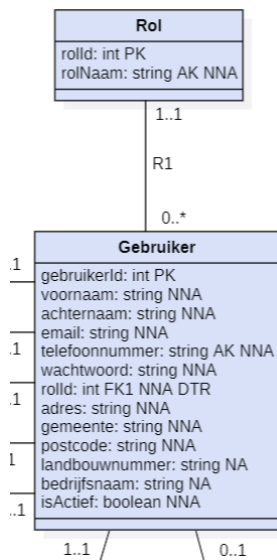
Scher(m)en:

CW3_homepage_consultant



Action	Explanation
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Landbouwers opvragen	Gaat naar use case “Gegevens van landbouwers opvragen”.
Profiel bekijken	Gaat naar use case “Profiel consultant bekijken”.

Betrokken tabellen ERD:



3.1.2.4. Homepagina overheidsmedewerker

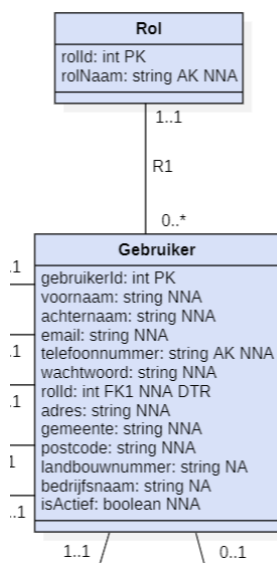
Schermb(en):

CW4_homepage_overheidsmedewerker



Action	Explanation
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Bekijk landbouwschade	Gaat naar use case “Gegevens over landbouwschade opvragen per gemeente”.
Profiel bekijken	Gaat naar use case “Profiel overheidsmedewerker bekijken”.

Betrokken tabellen ERD:



3.1.2.5. Homepagina account aanvragen

Scher(m)en:

CW5_account_aanvragen

[Home](#)[Login](#)

Account aanvragen

Vraag hieronder uw account aan. Kies de juiste optie.

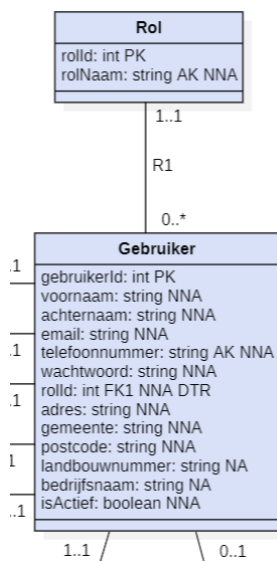
Account landbouwer aanvragen

Account consultant aanvragen

Account overheidsmedewerker aanvragen

Action	Explanation
Home	Gaat naar CW1_homepage_algemene_gebruiker.
Login	Gaat naar use case "Login".
Account landbouwer aanvragen	Gaat naar use case "Account landbouwer aanvragen".
Account consultant aanvragen	Gaat naar use case "Account consultant aanvragen".
Account overheidsmedewerker aanvragen	Gaat naar use case "Account overheidsmedewerker aanvragen".

Betrokken tabellen ERD:



3.1.2.6. Algemene gebruiker

ACCOUNT LANDBOUWER AANVRAGEN

Functionaliteit: Als een algemene gebruiker, kan ik een account voor een landbouwer aanvragen.

Normaal verloop: De actor vult zijn voornaam, achternaam, landbouwnummer, e-mailadres, telefoonnummer en wachtwoord in. Het systeem toont een melding dat het aanmaken van het account succesvol is.

Alternatief:

Ontbrekende gegevens: Het systeem toont een melding dat nog niet alle gegevens zijn ingevuld. De actor vult de ontbrekende gegevens in.

Scherm(en):

CW6_account_landbouwer_aanvragen

Home Login

Account landbouwer aanvragen

Voornaam:

Achternaam:

Landbouwnummer:

E-mailadres:

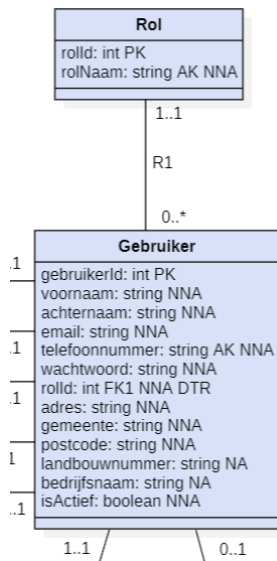
Telefoonnummer:

Wachtwoord:

Verzenden ¹

Action	Explanation
Home	Gaat naar CW1_homepage_algemene_gebruiker.
Login	Gaat naar use case "Login".
Verzenden	Het systeem maakt het account van de landbouwer aan. Gaat naar CW2_homepage_landbouwer.

Betrokken tabellen ERD:



ACCOUNT CONSULTANT AANVRAGEN

Functionaliteit: Als een algemene gebruiker, kan ik een account voor een consultant aanvragen.

Normaal verloop: De actor vult zijn voornaam, achternaam, bedrijfsnaam, e-mailadres, telefoonnummer en wachtwoord in. Het systeem toont een melding dat het aanmaken van het account succesvol is.

Alternatief:

Ontbrekende gegevens: Het systeem toont een melding dat nog niet alle gegevens zijn ingevuld. De actor vult de ontbrekende gegevens in.

Scherm(en):

CW7_account_consultant_aanvragen

[Home](#)[Login](#)

Account consultant aanvragen

Voornaam:

Achternaam:

Bedrijfsnaam:

E-mailadres:

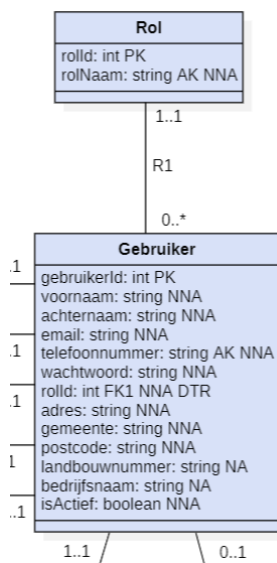
Telefoonnummer:

Wachtwoord:

Verzenden

Action	Explanation
Home	Gaat naar CW1_homepage_algemene_gebruiker.
Login	Gaat naar use case "Login".
Verzenden	Het systeem maakt het account van de consultant aan. Gaat naar CW3_homepage_consultant.

Betrokken tabellen ERD:



ACCOUNT OVERHEIDSMEDEWERKER AANVRAGEN

Functionaliteit: Als een algemene gebruiker, kan ik een account voor een overheidsmedewerker aanvragen.

Normaal verloop: De actor vult zijn voornaam, achternaam, gemeente, e-mailadres, telefoonnummer en wachtwoord in. Het systeem toont een melding dat het aanmaken van het account succesvol is.

Alternatief:

Ontbrekende gegevens: Het systeem toont een melding dat nog niet alle gegevens zijn ingevuld. De actor vult de ontbrekende gegevens in.

Scherf(en):

CW8_account_overheidsmedewerker_aanvragen

[Home](#)[Login](#)

Account overheidsmedewerker aanvragen

Voornaam:

Achternaam:

Gemeente:

E-mailadres:

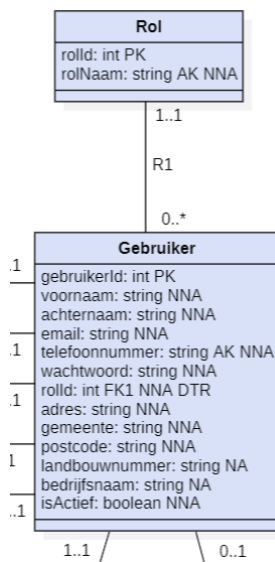
Telefoonnummer:

Wachtwoord:

[Verzenden](#)

Action	Explanation
Home	Gaat naar CW1_homepage_algemene_gebruiker.
Login	Gaat naar use case "Login".
Verzenden	Het systeem maakt het account van de overheidsmedewerker aan. Gaat naar CW4_homepage_overheidsmedewerker.

Betrokken tabellen ERD:



LOGIN

Functionaliteit: Als een algemene gebruiker, kan ik inloggen in het systeem.

Normaal verloop: De actor voert zijn/haar e-mailadres en wachtwoord in. Het systeem toont de homepagina van de actor.

Alternatieven:

Onbekend e-mailadres: Het systeem toont een foutmelding dat er geen profiel bestaat met dit e-mailadres.

Fout wachtwoord: Het systeem toont een foutmelding dat het wachtwoord verkeerd is. De actor vult zijn wachtwoord opnieuw in.

Reset wachtwoord: Als de actor kiest om zijn wachtwoord te resetten, switcht het systeem naar de use case “Wachtwoord resetten”.

Scherm(en):

CW9_inloggen

Home Account aanvragen

E-mailadres:

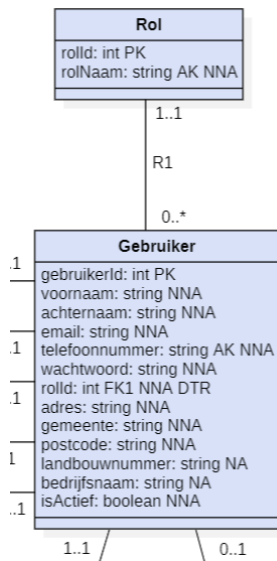
Wachtwoord:

[Wachtwoord vergeten?](#) ²

¹

Action	Explanation
Home	Gaat naar CW1_homepage_algemene_gebruiker.
Account aanvragen	Gaat naar CW5_account aanvragen.
Wachtwoord vergeten	Gaat naar use case “Wachtwoord resetten”.
Login	Het systeem verwijst de gebruiker naar de juiste homepagina.

Betrokken tabellen ERD:



WACHTWOORD RESETTEN

Functionaliteit: Als een algemene gebruiker, kan ik mijn wachtwoord resetten.

Normaal verloop: De actor vult zijn e-mailadres in. Het systeem stuurt een e-mail met een nieuwe link voor een nieuw wachtwoord naar dit e-mailadres.

Scherm(en):

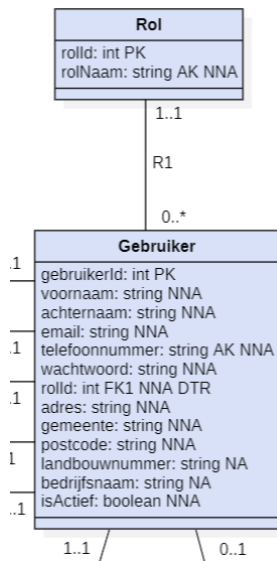
CW10_wachtwoord_vergeten

[Home](#)
[Login](#)

E-mailadres:

Action	Explanation
Home	Gaat naar CW1_homepage_algemene_gebruiker.
Login	Gaat naar use case "Login".
Verzenden	Het systeem verzendt een link naar het e-mailadres van de gebruiker.

Betrokken tabellen ERD:



3.1.2.7. Landbouwer

VELD REGISTREREN

Functionaliteit: Als landbouwer, kan ik een veld registreren.

Preconditie: De landbouwer is ingelogd.

Normaal verloop: De actor duidt op de kaart één van zijn velden aan en geeft het veld een naam en bevestigt dit. Het systeem toont een melding dat het succesvol is geregistreerd.

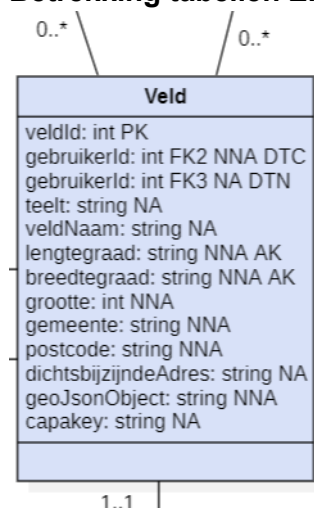
Scherm(en):

CW11a_veld registreren



Action	Explanation
Dashboard	Gaat naar CW2_homepage_landbouwer.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
1	De landbouwer selecteert zijn veld op de kaart. Gaat naar CW11b_veld_registreren.

Betrekking tabellen ERD:



CW11b_veld_registreren

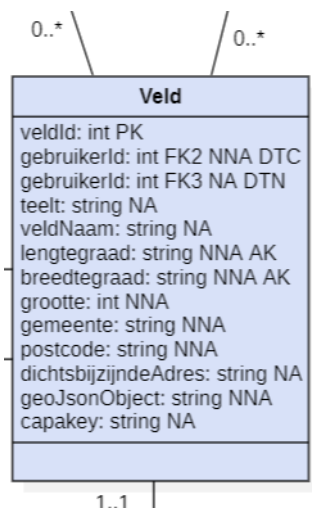
Dashboard	Uitloggen
-----------	-----------

Gegevens veld

Veldnaam:

Action	Explanation
Dashboard	Gaat naar CW2_homepage_landbouwer.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Bevestigen	Het systeem slaat het veld op.

Betrekking tabellen ERD:



VELDEN OPVRAGEN

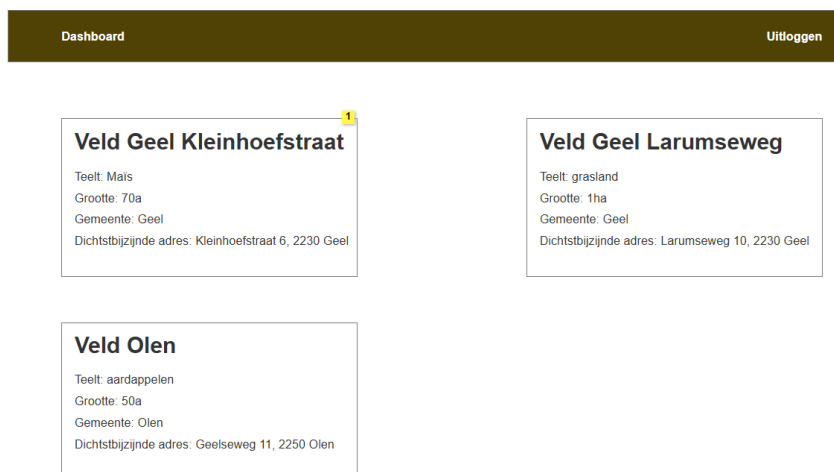
Functionaliteit: Als landbouwer, kan ik mijn velden opvragen.

Preconditie: De landbouwer is ingelogd.

Normaal verloop: Het systeem toont een lijst van alle geregistreerde velden van de actor.

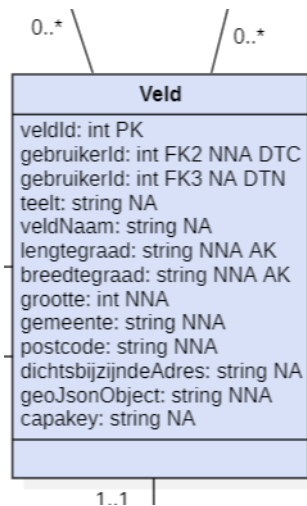
Scherm(en):

CW12_velden_opvragen



Action	Explanation
Dashboard	Gaat naar CW2_homepage_landbouwer.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
1	Gaat naar use case “Details veld opvragen”.

Betrekking tabellen ERD:



DETAILS VELD OPVRAGEN

Functionaliteit: Als landbouwer, kan ik details van mijn velden opvragen.

Preconditie: De landbouwer is ingelogd en heeft één veld geselecteerd.

Normaal verloop: Het systeem toont alle details, zoals teelt, grootte, gemeente en dichtsbijzijnde adres. Verder toont het systeem de adviezen, schadeclaims en de status van de schadeclaims van het geselecteerde veld.

Alternatief:

Consultant toegang geven: Als de actor ervoor kiest om een consultant toegang te geven, switcht het systeem naar de use case ‘Consultant toegang verlenen’.

Scher(m)en):
CW13_details_veld_opvragen

Dashboard

Uitloggen

Veld Geel Kleinhoefstraat

Teelt: Maïs

Grootte: 70a

Gemeente: Geel

Dichtstbijzijnde adres: Kleinhoefstraat 6, 2230 Geel

Advies

Mark Cools - 15/12/2024

U kunt best duidelijke foto's uploaden van uw schade aan uw veld.

Schadeclaims:

Aan het verwerken

Schadeclaim - 17/12/2024

Klaar

Schadeclaim - 08/03/2021

Schadeclaim - 25/08/2024

Bekijken

Bewerken

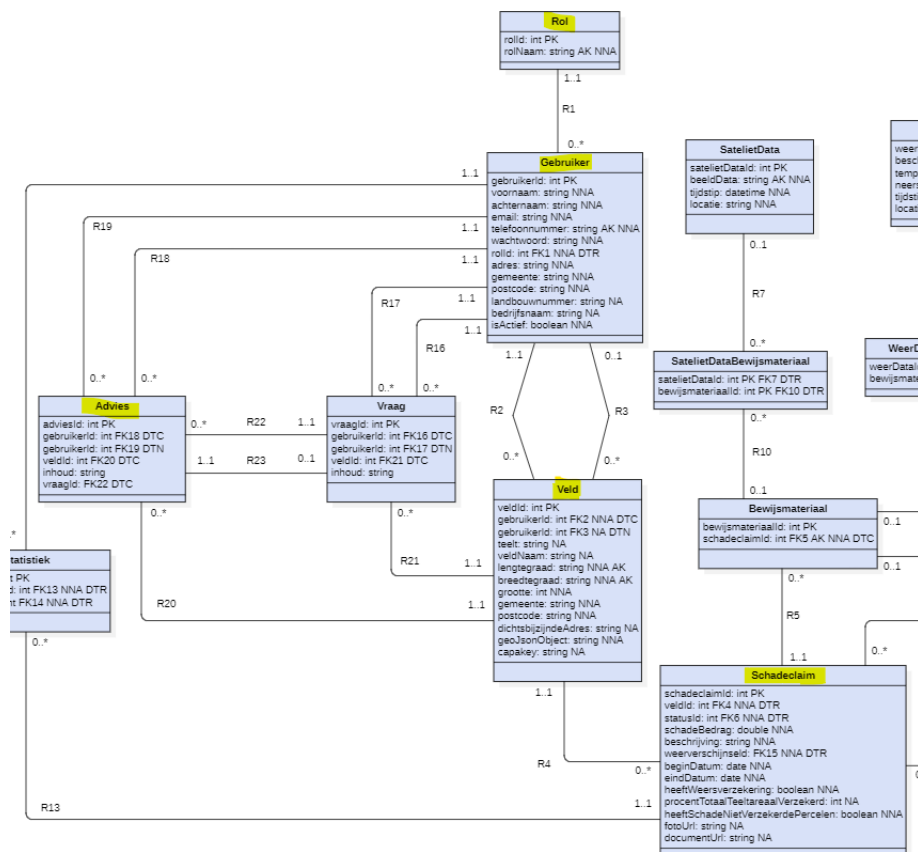
Nieuwe schadeclaim aanvragen

Geef een consultant toegang

Action	Explanation
Dashboard	Gaat naar CW2_homepage_landbouwer.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Nieuwe schadeclaim aanvragen	Gaat naar use case “Schadeclaim aanvragen”.
Geef een consultant toegang	Gaat naar use case “Consultant toegang verlenen”.
Bekijken	Het systeem toont het schadeclaimformulier.
Bewerken	Gaat naar use case “Schadeclaim aanpassen”.
1	De landbouwer selecteert een schadeclaim uit de lijst.

26

Betrekking tabellen ERD:



CONSULTANT TOEGANG VERLENEN

Functionaliteit: Als landbouwer, kan ik een consultant toegang verlenen tot één van mijn velden.

Preconditie: De landbouwer is ingelogd.

Normaal verloop: De actor selecteert een consultant uit de lijst en stelt zijn vraag en bevestigt deze. Het systeem toont een melding dat het toevoegen van de consultant succesvol is.

Scher(m)en:

CW15_consultant_toegang_verlenen

Dashboard

Uitloggen

Geef een consultant toegang

Naam consultant:

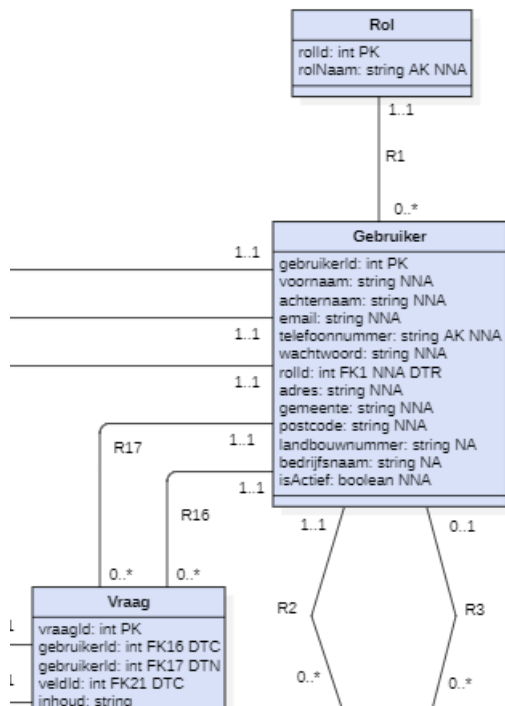
Selecteer een consultant

Stel hier uw vraag:

Geef toegang

Action	Explanation
Dashboard	Gaat naar CW2_homepage_landbouwer.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Geef toegang	Gaat naar CW13_details_veld_opvragen. Het systeem geeft de consultant toegang tot dat veld.

Betrekking tabellen ERD:



SCHADECLAIM AANVRAGEN

Functionaliteit: Als landbouwer, kan ik een schadeclaim aanvragen voor een veld.

Preconditie: De landbouwer is ingelogd en heeft een veld geselecteerd.

Normaal verloop: De actor vult gegevens zoals schadebedrag, beschrijving, weersverschijnsel, begindatum, einddatum, over het veld in en heeft de mogelijkheid om een foto of een document te uploaden.

Alternatief:

Niet alle gegevens zijn ingevuld: Het systeem toont een foutmelding. De actor vult de ontbrekende gegevens in.

Scherm(en):

CW14_schadeclaim_aanvragen

Dashboard

Uitloggen

Vul onderstaande gegevens in

Schadebedrag:

Beschrijving:

Weersverschijnsel dat schade heeft veroorzaakt:

hevige en aanhoudende regenval
overstroming
stormwinden en rukwinden met een lokaal karakter
aardbeving
ernstige droogte (zonnebrand komt niet in aanmerking)
uitzonderlijke vorst
List item 1natuurbrand van natuurlijke oorsprong
aardverschuiving of grondverzakking
hagelbuien
sneeuw- of ijsdruk

Begindatum :

Einddatum:

Beschikt u over een brede weersverzekering? ☐ Ja ☐ Nee

Hoeveel percent van het totaal teeltareaal is verzekerd
via een brede weersverzekering?

Is er schade aan niet-verzekerde percelen? ☐ Ja ☐ Nee

Voeg een foto toe:

Foto uploaden

Voeg een document toe:

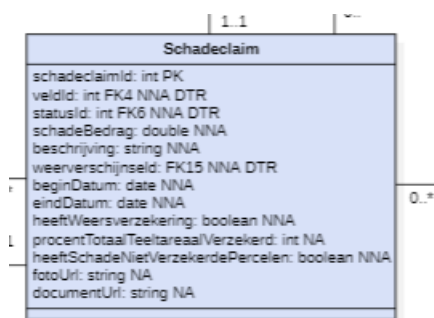
Document uploaden

Bevestigen

Annuleren

Action	Explanation
Dashboard	Gaat naar CW2_homepage_landbouwer.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Foto uploaden	De landbouwer selecteert een foto uit zijn verkenner.
Document uploaden	De landbouwer selecteert een document uit zijn verkenner.
Bevestigen	Gaat naar CW13_details_veld_opvragen. Het systeem verwerkt de schadeclaim.
Annuleren	Gaat naar CW13_details_veld_opvragen. Het systeem gaat terug naar de details veld pagina zonder iets van gegevens te verwerken.

Betrekking tabellen ERD:



VRAGEN STELLEN AAN CHATBOT

Functionaliteit: Als een landbouwer, kan ik vragen stellen aan een chatbot.

Preconditie: De landbouwer is ingelogd.

Normaal verloop: De actor stelt een vraag aan de chatbot. Het systeem geeft een antwoord op deze vraag.

Scherm(en):

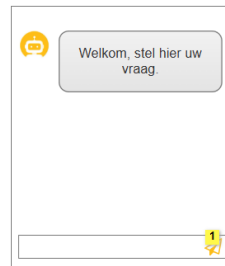
CW2_homepage_landbouwer

Welkom

Veld registreren

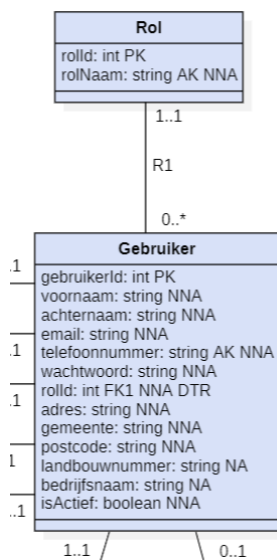
Velden opvragen

Profiel bekijken



Action	Explanation
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Veld registreren	Gaat naar use case "Veld registreren".
Velden opvragen	Gaat naar use case "Velden opvragen".
Profiel bekijken	Gaat naar use case "Profiel landbouwer bekijken".
1	De landbouwer verstuurt zijn vraag naar de chatbot.

Betrokken tabellen ERD:



SCHADECLAIM AANPASSEN

Functionaliteit: Als landbouwer, kan ik een schadeclaim aanpassen.

Preconditie: De landbouwer is ingelogd en heeft een veld en een schadeclaim geselecteerd.

Normaal verloop: De actor past de gegevens, zoals schadebedrag, beschrijving, weersverschijnsel, begindatum, einddatum, ... aan en bevestigt deze. Het systeem toont een melding dat de aanpassing is gedaan.

Scher(m)en:

CW17_formulier_schadeclaim_aanpassen

Dashboard

Uitloggen

Formulier schadeclaim aanpassen

Schadebedrag:

Beschrijving:

Weersverschijnsel dat schade heeft veroorzaakt:

hevige en aanhoudende regenval
overstroming
stormwinden en rukwinden met een lokaal karakter
aardbeving
ernstige droogte (zonnebrand komt niet in aanmerking)
uitzonderlijke vorst
List item 1natuurbrand van natuurlijke oorsprong
aardverschuiving of grondverzakking
hagelbuien
sneeuw- of ijsdruk

Begindatum :

Einddatum:

Beschikt u over een brede weersverzekering? ☒ Ja ☐ Nee

Hoeveel percent van het totaal teeltareaal is verzekerd
via een brede weersverzekering?

Is er schade aan niet-verzekerde percelen? ☐ Ja ☒ Nee

Voeg een foto toe:

Foto uploaden

Voeg een document toe:

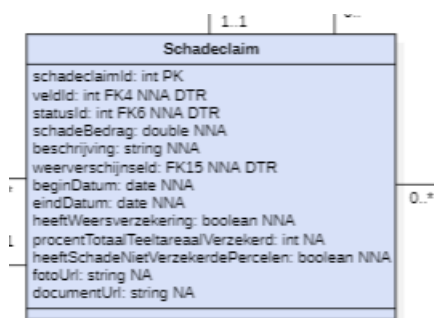
Document uploaden

Bevestigen

Annuleren

Action	Explanation
Dashboard	Gaat naar CW2_homepage_landbouwer.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Foto uploaden	De landbouwer selecteert een foto uit zijn verkenner.
Document uploaden	De landbouwer selecteert een document uit zijn verkenner.
Bevestigen	Gaat naar CW13_details_veld_opvragen. Het systeem verwerkt de wijzigingen aan de schadeclaim.
Annuleren	Gaat naar CW13_details_veld_opvragen. Het systeem gaat terug naar de details veld pagina zonder de gegevens van de schadeclaim aan te passen.

Betrekking tabellen ERD:



ALLE SCHADECLAIMS BEKIJKEN

Functionaliteit: Als landbouwer, kan ik al mijn schadeclaims bekijken.

Preconditie: De landbouwer is ingelogd.

Normaal verloop: Het systeem toont een lijst van alle schadeclaims.

Scherm(en):

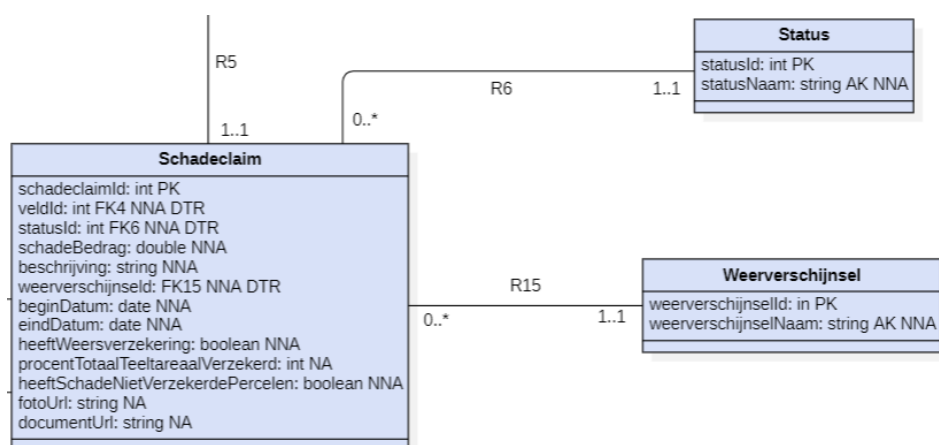
CW18_alle_schadeclaims_bekijken

Schadeclaims

Naam veld	Datum	Status
Veld Geel Kleinhoefstraat ¹	17/12/2024	Aan het verwerken
Veld Geel Kleinhoefstraat	25/08/2024	Klaar
Veld Geel Kleinhoefstraat	08/03/2021	Klaar
Veld Geel Larumseweg	12/03/2020	Klaar

Action	Explanation
Dashboard	Gaat naar CW2_homepage_landbouwer.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
1	De landbouwer selecteert een schadeclaim uit de lijst. Het systeem toont het schadeclaimformulier.

Betrekking tabellen ERD:



PROFIEL BEKIJKEN

Functionaliteit: Als landbouwer, kan ik mijn profiel bekijken

Preconditie: De landbouwer is ingelogd.

Normaal verloop: Het systeem toont alle informatie over het profiel van de actor.

Alternatief:

Wachtwoord resetten: Als de actor kiest om zijn wachtwoord te resetten, switcht het systeem naar de use case “Wachtwoord resetten”.

Scher(m)en:

CW16_profiel_bekijken

Dashboard Logout

Account bewerken

Voornaam:

Achternaam:

Landbouwnummer:

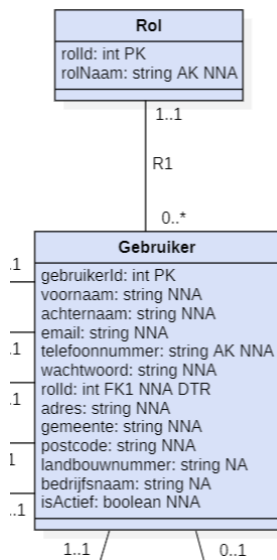
E-mailadres:

Telefoonnummer:

Wachtwoord:

Action	Explanation
Dashboard	Gaat naar CW2_homepage_landbouwer.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Wachtwoord resetten	Gaat naar Use Case ‘Wachtwoord resetten’.
Opslaan	Het systeem slaat de gegevens op.

Betrekking tabellen ERD:



WACHTWOORD RESETTEN

Functionaliteit: Als landbouwer, kan ik mijn wachtwoord resetten.

Preconditie: De landbouwer is ingelogd.

Normaal verloop: De actor vult zijn e-mailadres in. Het systeem stuurt een e-mail met een nieuwe link voor een nieuw wachtwoord naar dit e-mailadres.

Scherm(en):

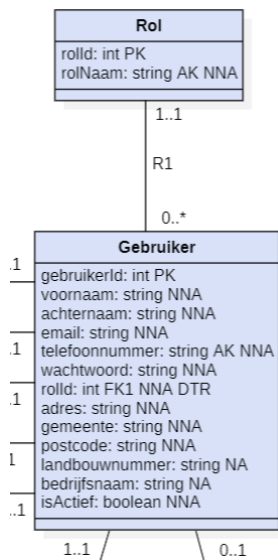
CW10_wachtwoord_vergeten

Home	Login
------	-------

E-mailadres:

Action	Explanation
Home	Gaat naar CW1_homepage_algemene_gebruiker.
Login	Gaat naar use case "Login".
Verzenden	Het systeem verzendt een link naar het e-mailadres van de gebruiker.

Betrekking tabellen ERD:



3.1.2.8. Consultant

GEGEVENS VAN LANDBOUWERS OPVRAGEN

Functionaliteit: Als consultant, kan ik de gegevens van de landbouwers opvragen.

Preconditie: De consultant is ingelogd.

Normaal verloop: De actor vult een landbouwnummer in. Het systeem toont de gegevens van deze landbouwer. De actor bevestigt dit. Het systeem toont alle velden waarvoor de landbouwer toegang heeft gegeven aan de actor.

Scherm(en):

CW19a_gegevens_landbouwers_opvragen

Dashboard
Uitloggen

Gegevens landbouwer opvragen

Landbouwnummer:

2

Zoek

Voornaam:

Achternaam:

E-mailadres:

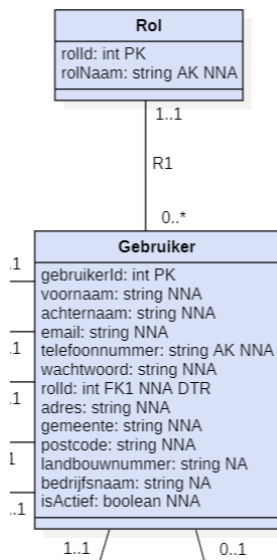
Telefoonnummer:

1

Bekijk velden

Action	Explanation
Dashboard	Gaat naar CW3_homepage_consultant.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Zoek	Het systeem zoekt en toont de gegevens van de landbouwer met dit landbouwnummer.
Bekijk Velden	Gaat naar CW19b_gegevens_landbouwers_bekijken. Het systeem toont alle velden waarvoor toegang is gegeven door de landbouwer.

Betrekking tabellen ERD:



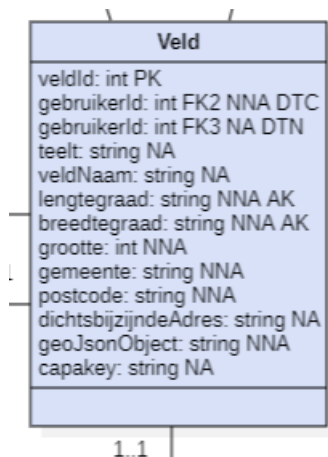
CW19b_gegevens_landbouwers_bekijken

Dashboard	Uitloggen
-----------	-----------

Veld Geel Kleinhoefstraat Teelt: Mais Grootte: 70a Gemeente: Geel Dichtstbijzijnde adres: Kleinhoefstraat 6, 2230 Geel

Action	Explanation
Dashboard	Gaat naar CW3_homepage_consultant.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
1	De consultant selecteert een veld. Gaat naar use case “Advies geven over schadeclaim.

Betrekking tabellen ERD:



ADVIES GEVEN OVER SCHADECLAIM

Functionaliteit: Als consultant, kan ik advies geven aan landbouwers over de schadeclaim.

Preconditie: De consultant is ingelogd en heeft toegang gekregen tot een veld.

Normaal verloop: De actor vult zijn advies in en verzendt dit.

Scherm(en):

CW20_advies_geven

Dashboard
Uitloggen

Advies geven

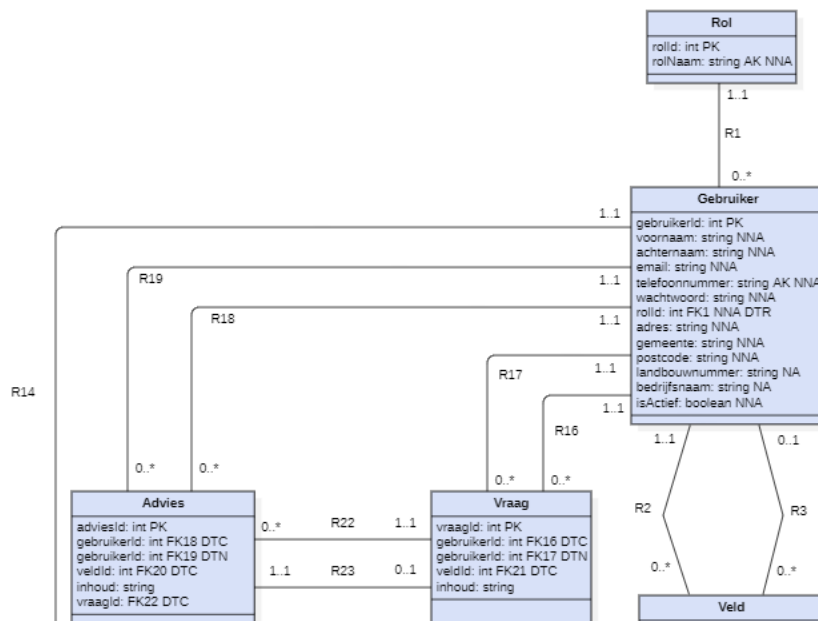
Vraag: Hoe toon ik het best de schade aan mijn veld?

Advies:

Bevestigen

Action	Explanation
Dashboard	Gaat naar CW3_homepage_consultant.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Bevestigen	Gaat naar CW19b_gegevens_landbouwers_bekijken. Het systeem slaat het advies op.

Betrekking tabellen ERD:



PROFIEL BEKIJKEN

Functionaliteit: Als consultant, kan ik mijn profiel bekijken

Preconditie: De consultant is ingelogd.

Normaal verloop: Het systeem toont alle informatie over het profiel van de actor.

Alternatief:

Wachtwoord resetten: Als de actor kiest om zijn wachtwoord te resetten, switcht het systeem naar de use case "Wachtwoord resetten".

Scher(m)en:
CW21_profiel_bekijken

Dashboard
Logout

Account bewerken

Voornaam:

Achternaam:

Bedrijfsnaam:

E-mailadres:

Telefoonnummer:

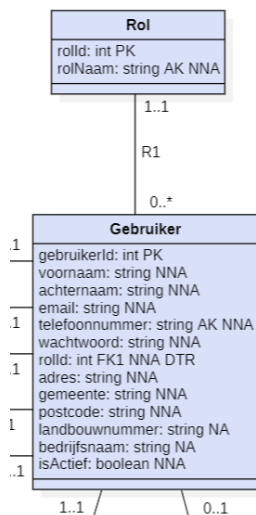
Wachtwoord:

Wachtwoord resetten

Opslaan

Action	Explanation
Dashboard	Gaat naar CW3_homepage_consultant.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Wachtwoord resetten	Gaat naar use case “Wachtwoord resetten”.
Opslaan	Het systeem slaat de gegevens op.

Betrekking tabellen ERD:



WACHTWOORD RESETTEN

Functionaliteit: Als consultant, kan ik mijn wachtwoord resetten.

Preconditie: De consultant is ingelogd.

Normaal verloop: De actor vult zijn e-mailadres in. Het systeem stuurt een e-mail met een nieuwe link voor een nieuw wachtwoord naar dit e-mailadres.

Scherf(en):

CW10_wachtwoord_vergeten

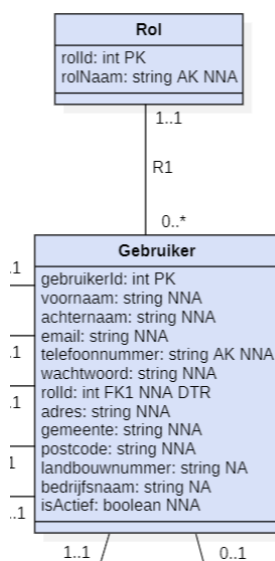
[Home](#)[Login](#)

E-mailadres:

[Verzenden](#)

Action	Explanation
Home	Gaat naar CW1_homepage_algemene_gebruiker.
Login	Gaat naar use case "Login".
Verzenden	Het systeem verzendt een link naar het e-mailadres van de gebruiker.

Betrekking tabellen ERD:



3.1.2.9. Overheidsmedewerker

GEGEVENS OVER LANDBOUWSCHADE OPVRAGEN PER GEMEENTE

Functionaliteit: Als een overheidsmedewerker, kan ik de gegevens over landbouwschade opvragen per gemeente.

Preconditie: De overheidsmedewerker is ingelogd.

Normaal verloop: Het systeem toont de gegevens over de schade van de verschillende gemeenten op een dashboard, zoals totaal aantal velden, aantal verzekerde velden, totaal aantal velden met schade, totaal schadebedrag,

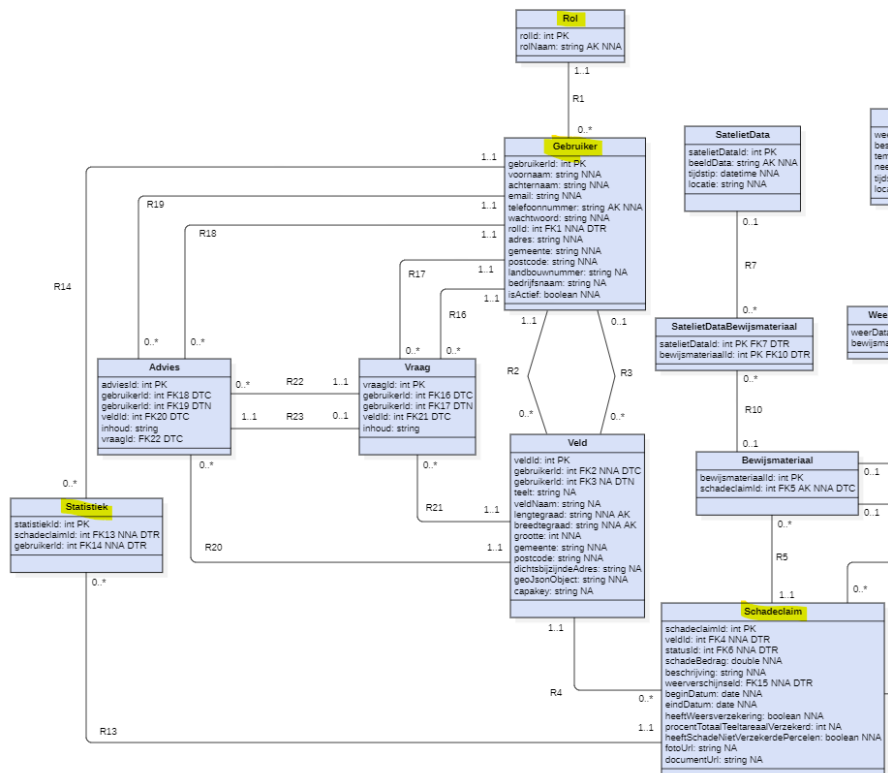
Scherm(en):

CW22_gegevens_landbouwschade_per_gemeente

Dashboard								Uitloggen
Totaal aantal velden	Aantal verzekerde velden	Totaal aantal velden met schade	Aantal verzekerde velden met schade	Aantal niet-verzekerde velden met schade	Totaal schadebedrag	Schadebedrag verzekerde velden	Schadebedrag niet-verzekerde velden	Meest voorkomende weersverschijnselen
401	287	112	95	17	€ 9.512	€ 7.145	€ 2.367	1. Vulkaanuitbarsting 2. Aardbeving 3. Tsunami

Action	Explanation
Dashboard	Gaat naar CW4_homepage_overheidsmedewerker.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.

Betrekking tabellen ERD:



PROFIEL BEKIJKEN

Functionaliteit: Als overheidsmedewerker, kan ik mijn profiel bekijken

Preconditie: De overheidsmedewerker is ingelogd.

Normaal verloop: Het systeem toont alle informatie over het profiel van de actor.

Alternatief:

Wachtwoord resetten: Als de actor kiest om zijn wachtwoord te resetten, switcht het systeem naar de use case “Wachtwoord resetten”.

Scher(m)en:
CW23_profiel_bekijken

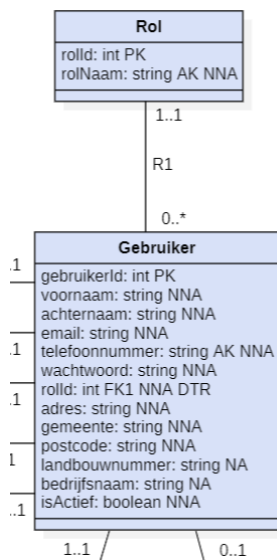
Dashboard
Logout

Account bewerken

Voornaam:
Achternaam:
Gemeente:
E-mailadres:
Telefoonnummer:
Wachtwoord:

Action	Explanation
Dashboard	Gaat naar CW4_homepage_overheidsmedewerker.
Uitloggen	Gaat naar CW1_homepage_algemene_gebruiker.
Wachtwoord resetten	Gaat naar use case “Wachtwoord resetten”.
Opslaan	Het systeem slaat de gegevens op.

Betrekking tabellen ERD:



WACHTWOORD RESETTEN

Functionaliteit: Als overheidsmedewerker, kan ik mijn wachtwoord resetten.

Preconditie: De overheidsmedewerker is ingelogd.

Normaal verloop: De actor vult zijn e-mailadres in. Het systeem stuurt een e-mail met een nieuwe link voor een nieuw wachtwoord naar dit e-mailadres.

Scher(m)en:

CW10_wachtwoord_vergeten

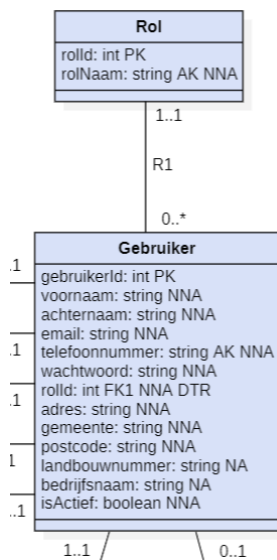
Home Login

E-mailadres:

Verzenden

Action	Explanation
Home	Gaat naar CW1_homepage_algemene_gebruiker.
Login	Gaat naar use case "Login".
Verzenden	Het systeem verzendt een link naar het e-mailadres van de gebruiker.

Betrekking tabellen ERD:



3.1.2.10. Systeem

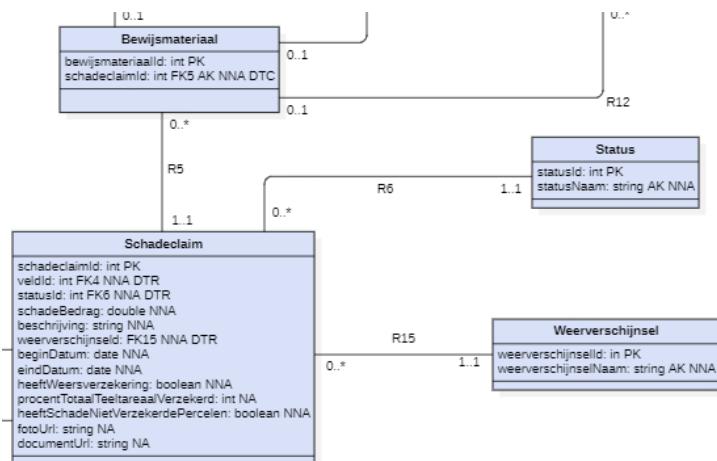
SCHADECLAIM AANMAKEN

Functionaliteit: Als het systeem, kan ik een schadeclaim aanmaken.

Preconditie: De use case “Schadeclaim aanvragen” is succesvol uitgevoerd.

Normaal verloop: Het systeem maakt een schadeclaim aan met behulp van de gegevens die de actor heeft ingegeven en schakelt hiervoor over naar de volgende use cases om dit proces te voltooien: “Bewijs van Eoplaza halen”, “Bewijs van mediadata halen” en “Bewijs van weerdata halen”.

Betrekking tabellen ERD:

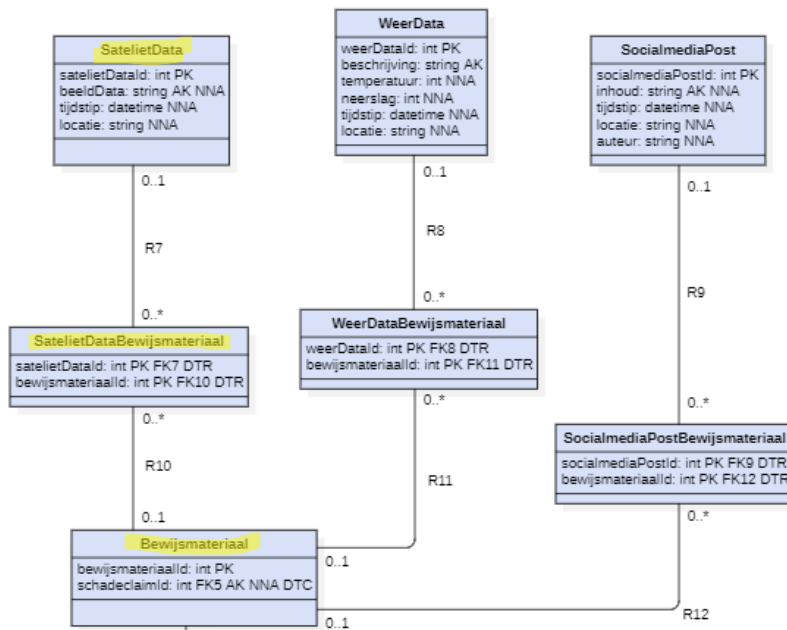


BEWIJS VAN EOPLAZA SERVICES HALEN

Functionaliteit: Als het systeem, kan ik bewijs van Eoplaza services halen.

Normaal verloop: Het systeem verzamelt gegevens uit Eoplaza services. De actor ziet het resultaat van deze use case bij het bekijken van de schadeclaim.

Betrekking tabellen ERD:

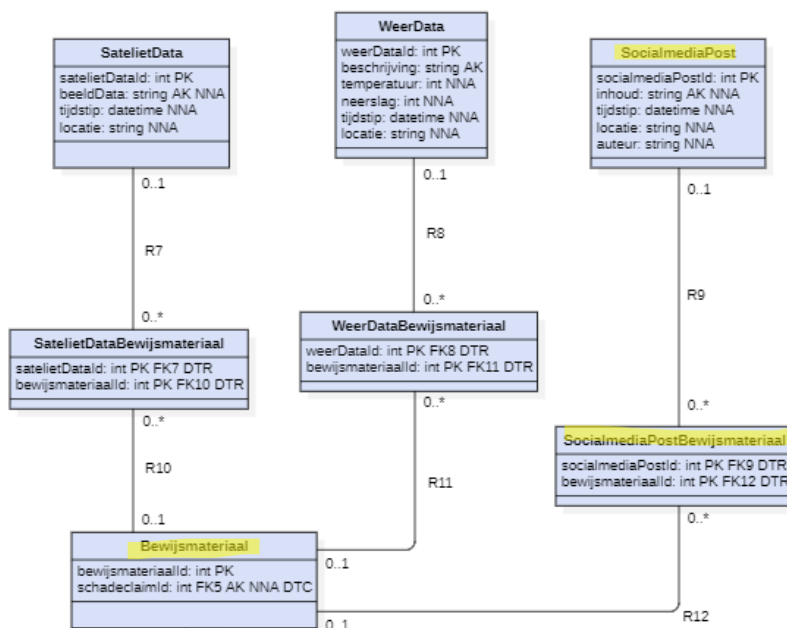


BEWIJS VAN MEDIADATA HALEN

Functionaliteit: Als het systeem, kan ik bewijs van mediadata halen.

Normaal verloop: Het systeem verzamelt gegevens uit mediadata. De actor ziet het resultaat van deze use case bij het bekijken van de schadeclaim.

Betrekking tabellen ERD:

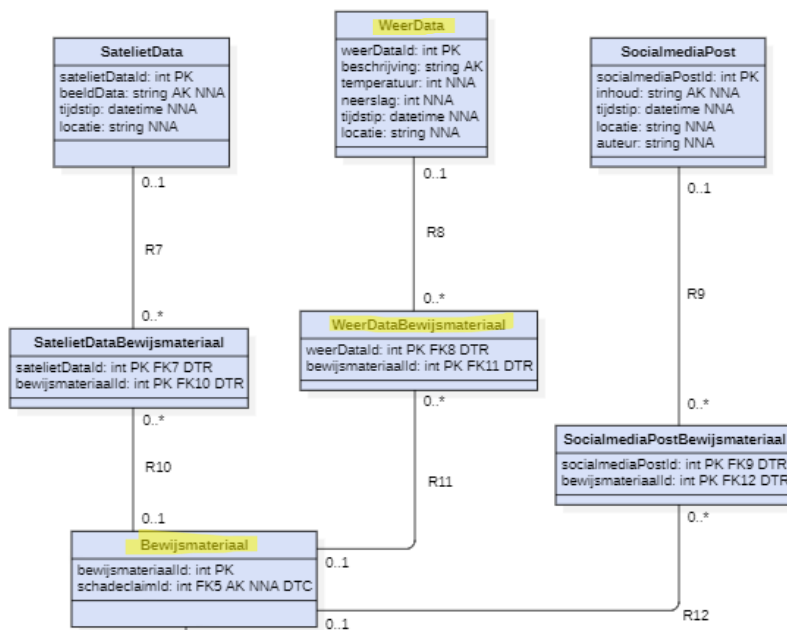


BEWIJS VAN WEERDATA HALEN

Functionaliteit: Als het systeem, kan ik bewijs van weerdata halen.

Normaal verloop: Het systeem verzamelt gegevens uit weerdata. De actor ziet het resultaat van deze use case bij het bekijken van de schadeclaim.

Betrekking tabellen ERD:



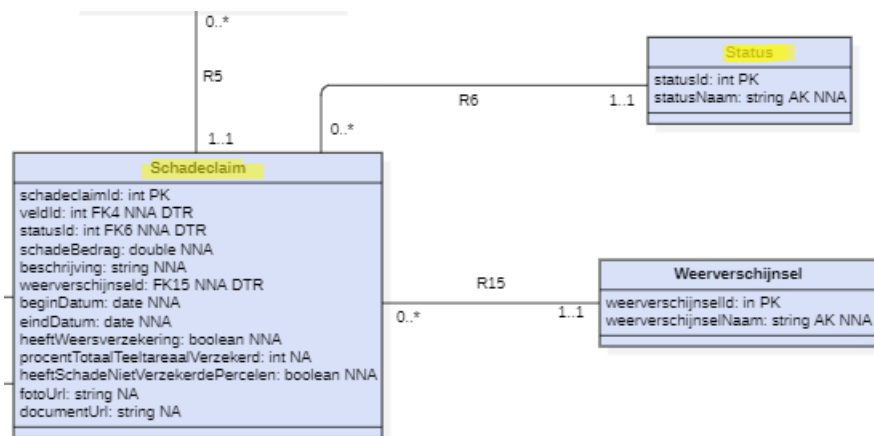
STATUS DOCUMENT SCHADECLAIM AANPASSEN

Functionaliteit: Als het systeem, kan ik de status van een schadeclaim aanpassen.

Preconditie: De use case “schadeclaim aanmaken” is succesvol uitgevoerd.

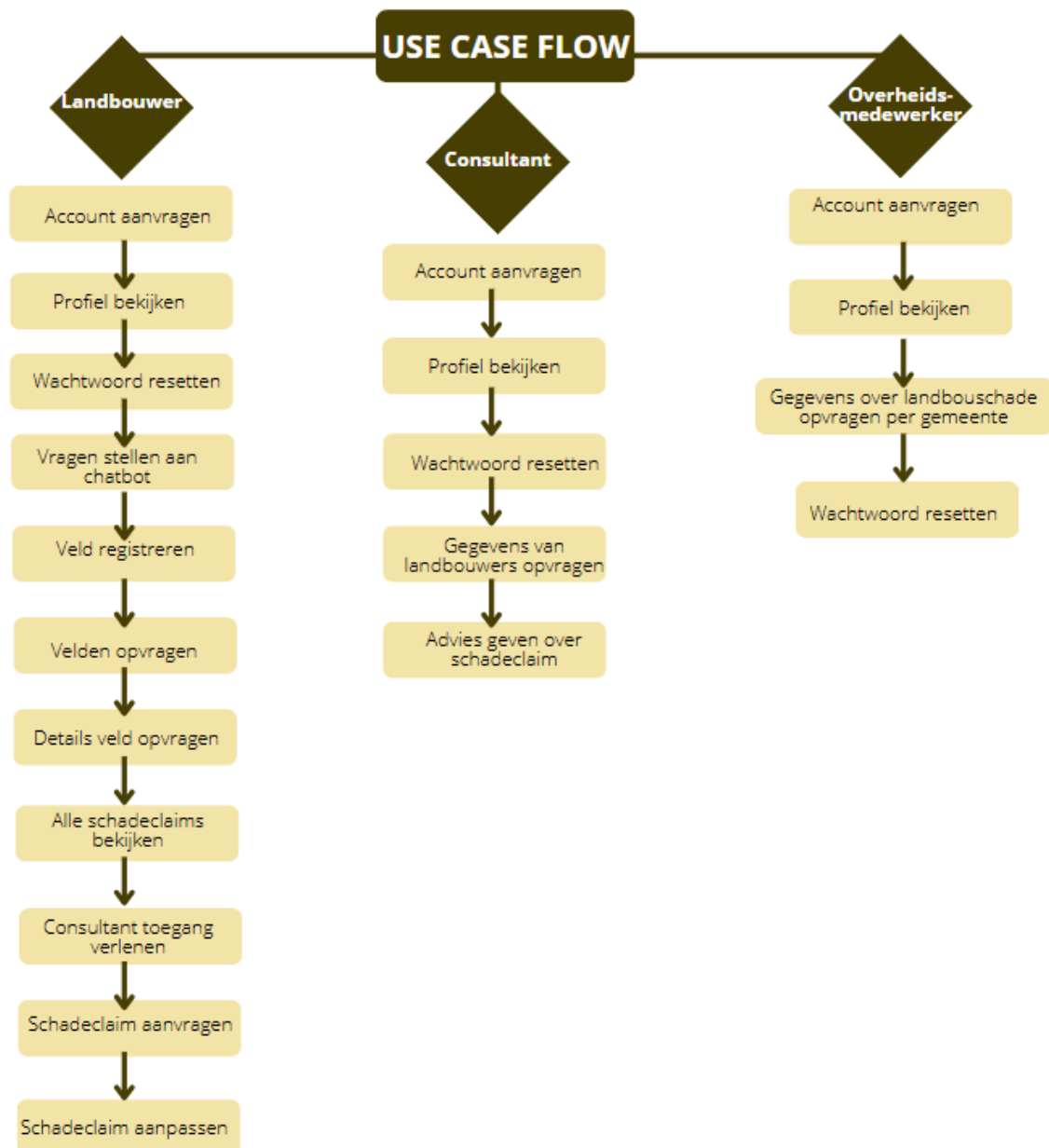
Normaal verloop: Het systeem update de status van de schadeclaim naderhand van het verdere verloop van de use case “Schadeclaim aanmaken”.

Betrekking tabellen ERD:



3.1.3. Flowdiagram use cases

Hieronder wordt het flowdiagram van de use cases getoond. Hierin worden de use cases gevisualiseerd en gerangschikt. Wat bovenaan staat (account aanvragen) moet eerst uitgevoerd worden. Vervolgens kan de pijl gevolgd worden om alle use cases op het juiste moment uit te voeren. Er zijn enkele use cases, zoals profiel bekijken en wachtwoord resetten, die overal na account aanvragen kunnen gebeuren.



3.1.4. Prioriteit per functionaliteit

Hieronder wordt aangegeven welke functionaliteiten (use cases) welke MoSCoW-prioriteit hebben.

Letter	Statement	Use-cases
M	Must have this	<ul style="list-style-type: none">- Registreren- Login- Veld registreren- Velden opvragen- Details veld opvragen- Schadeclaim aanmaken- Schadeclaim aanvragen- Gegevens over landbouwschade opvragen per gemeente- Alle schadeclaims bekijken
S	Should have this if at all possible	<ul style="list-style-type: none">- Vragen stellen aan chatbot- Advies geven over schadeclaim- Bewijs van Eoplaza halen- Bewijs van mediadata halen- Bewijs van weerdata halen- Gegevens van landbouwers opvragen- Consultant toegang verlenen- Schadeclaim aanpassen
C	Could have this if it does not affect anything else	<ul style="list-style-type: none">- Wachtwoord resetten- Profiel bekijken- Status document schadeclaim aanpassen
W	Would like to have but won't have this time around	<ul style="list-style-type: none">-

3.2. Niet-functionele eisen

In dit onderdeel worden de niet-functionele eisen besproken. Dit wordt gedaan aan de hand van vier categorieën, namelijk: implementatie, externe interface, prestatie en kwaliteitsvereisten.

3.2.1. Implementatie

Dit project gaat gebruik maken van een FARM-stack. Dit staat voor FastAPI als back-end, React als front-end en MongoDB als database. Meer informatie kan teruggevonden worden bij het deel Deployment.

3.2.2. Externe interface

Het systeem gaat gebruik maken van externe API's, namelijk Groq API, Custom search JSON API en Plotly API. Meer informatie kan teruggevonden worden bij het deel Data Acquisition.

3.2.3. Prestatie

De response time van het systeem wil Cropwatchers zo hoog mogelijk krijgen, zodat de landbouwers, consultants en overheidsmedewerkers vlot het systeem kunnen gebruiken zonder grote vertragingen. Het is niet erg als er eens een vertraging is, maar dit moet beperkt blijven.

De schaalbaarheid van het systeem is enorm belangrijk. Na bijvoorbeeld een overstroming of een lange droogte gaan er meer landbouwers actief zijn op de applicatie. Ze gaan allemaal bijna gelijktijdig een schadeclaim willen aanvragen voor verschillende velden.

3.2.4. Kwaliteitsvereisten

Cropwatchers moet tevens rekening houden met security. Hiervoor wordt er gebruik gemaakt van CIA. Dit staat voor confidentiality (confidentialiteit), integrity (gaafheid) en availability (beschikbaarheid). Met confidentialiteit wordt bedoeld dat enkel de juiste personen bepaalde gegevens mogen te zien krijgen. Dit is het geval bij de overheidsmedewerker. Enkel hij/zij mag de gegevens over landbouwschade per gemeente te zien krijgen.

Daarnaast mag de consultant enkel gegevens van de velden van landbouwers kunnen opvragen die toegang gegeven hebben. De consultant mag tevens enkel advies geven over

de velden waar hij/zij toegang tot heeft gekregen. De gaafheid van gegevens betekent dat alle gegevens op elk moment correct moeten zijn. Dit omvat de gegevens van de gebruikers, maar eveneens van de velden en de schadeclaims. Tot slot moet ervoor gezorgd worden dat het systeem altijd beschikbaar is. Als dit even niet het geval is, is dit niet zo heel erg, maar dit moet beperkt blijven.

De CIA moet worden nageleefd zodat er geen problemen ontstaan met bijvoorbeeld de GDPR-regelgeving. De gegevens van de gebruikers zijn namelijk allemaal privé. Het is niet de bedoeling dat iedereen deze kan zien. Verder moeten de gegevens, die worden gebruikt om de schadeclaim aan te maken, correct zijn. Anders kan het voorvallen dat het document van de schadeclaim niet geldig is. Tot slot zouden de landbouwers geen schadeclaim kunnen aanvragen als het systeem niet beschikbaar is.

Om de gebruiksvriendelijkheid te bevorderen, gaat er voldoende uitleg geven worden om de gebruikers verder te helpen. Cropwatchers gaat bijvoorbeeld de stappen beschrijven die een landbouwer moet volgen om een veld te registreren, maar eveneens om een schadeclaim aan te vragen. Het systeem mag niet te ingewikkeld zijn.

4. Data verzameling

4.1. Analyse GeoPunt

4.1.1. Wat is GeoPunt Vlaanderen?

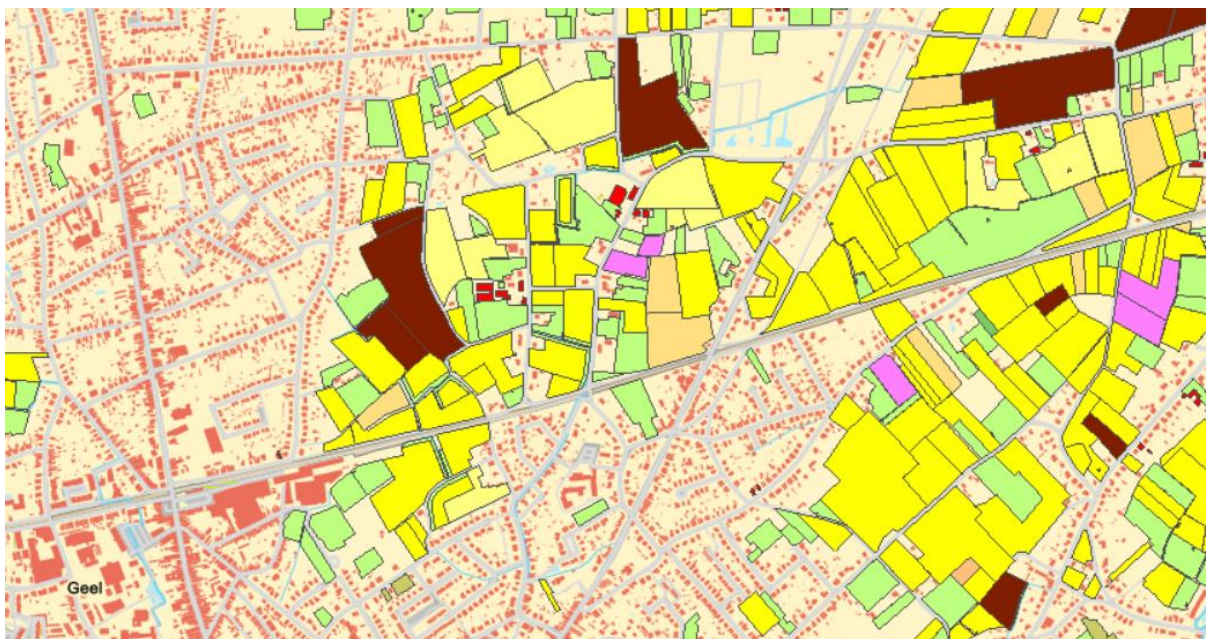
Geopunt Vlaanderen, de centrale toegangspoort tot geografische overheidsinformatie in Vlaanderen, brengt essentiële geografische gegevens en diensten op een intuïtieve en slimme manier bij gebruikers. Het geoportaal richt zich op een divers publiek, van individuele burgers op zoek naar geschikte bouwgrond tot overheidsinstanties, organisaties en bedrijven die geografische informatie nodig hebben voor hun activiteiten. Geopunt Vlaanderen biedt een uitgebreid scala aan data, diensten en toepassingen en een toegankelijke kaartweergave.

4.1.2. Hoe kunnen we dit gebruiken

Aangezien GeoPunt al data heeft over landbouw percelen kunnen we deze dataset gaan raadplegen. We kunnen deze dataset ook gaan combineren met een kaartweergaven waar we alle percelen kunnen weergeven. Een perceel bevat verschillende coördinaten die we kunnen gaan ingeven op de interactieve kaart. Hierdoor wordt het voor de boer duidelijk welke percelen van hen zijn.

4.1.3. Manueel werken met de kaart

De algemene kaart van Geopunt is uw toegang tot een schat aan geografische informatie. Ze bevat meer dan 700 kaartlagen en heel wat nuttige functionaliteiten. Op deze pagina's leggen we u uit hoe de verschillende schermen van de kaart eruit zien, hoe u de kaart kan gebruiken, en krijgt u een overzicht van de kaarttoepassingen.



4.1.4. Werken met kaartlagen

Link naar pagina: [klik hier](#)

Deze pagina geeft een overzicht van hoe je op Geopunt werkt met kaartlagen om toegang te krijgen tot geografische overheidsinformatie in Vlaanderen. Het biedt stapsgewijze uitleg over het gebruik en beheer van kaartlagen via het "lagenpaneel".

Enkele belangrijke onderdelen van de pagina zijn:

1. **Lagenpaneel:** Instructies over hoe je kaartlagen kunt toevoegen, beheren en aanpassen, inclusief opties voor het wijzigen van de achtergrondlaag en historische kaarten.
2. **Kaartlagen toevoegen:**
 - **Via de themaboom:** Kaartlagen zijn onderverdeeld in categorieën en subcategorieën, wat het vinden van thematisch gerelateerde lagen vergemakkelijkt.
 - **Via een service (WMS/WMTS):** Gebruikers kunnen externe kaartlagen toevoegen door een service-URL in te voeren en de gewenste lagen te selecteren.
3. **Kaartlaagbeheer:**
 - Acties voor elke kaartlaag omvatten tonen/verbergen, transparantie aanpassen, verwijderen, en informatie raadplegen. Let op dat sommige lagen afhankelijk zijn van het zoomniveau voor zichtbaarheid.

4. **Legende raadplegen:** De legende van sommige lagen is toegankelijk via het lagenpaneel en biedt extra context voor de weergegeven data.
5. **Volgorde van kaartlagen:** Gebruikers kunnen de volgorde van lagen aanpassen door lagen te slepen binnen het lagenpaneel, wat de leesbaarheid van de kaart verbetert.
6. **Bulk verwijderen:** Om efficiënt te werken, kunnen gebruikers meerdere geselecteerde kaartlagen en locaties tegelijk verwijderen.

Deze handleiding is bedoeld om gebruikers snel en effectief toegang te geven tot de uitgebreide geografische datasets op Geopunt.

4.1.5. Waterinfo vlaanderen

Waterinfo is een website van de Vlaamse overheid. Met deze website willen we jou zo correct mogelijk informeren over overstromingen en droogte. Hier worden alle metingen en voorspellingen samengebracht. Zo kunnen de nodige maatregelen getroffen worden om waterschade tot een minimum te beperken.

Link naar site: [klik hier](#)

4.1.6. Definities voor het begrijpen van data

Fluviaal: overstromingen vanuit waterlopen

Pluviaal: verstromingen door intense neerslag

4.1.7. Usefull links

[Landbouwgebruikspercelen](#) - Bundeling van de datasets die een overzicht geven van de percelen die in landbouwgebruik zijn. - [Kaartweergaven](#)

[Overstromingsgevaar en overstromingsrisico](#) - De overstromingsgevaarkaarten beschrijven de 'fysische eigenschappen' van de overstromingen zoals overstromingscontouren, waterdieptes en stroomsnelheden.

4.2. Analyse OpenEO

4.2.1. Parameters

Parameter	Beschrijving	Type	Default
date	Datumbereik te gebruiken voor de generatie van de opbrengstpotentiële kaart	Date range	
polygon	Feature verzameling van velden waarvoor u de opbrengstpotentiële kaart kan genereren	FeatureCollection	

4.2.2. Anomaly detection

Dit hoofdstuk beschrijft een technologiegedreven aanpak voor **anomaliedetectie** en **regionale benchmarking** van gewasgroei met behulp van de **CropSAR-methode**. De dienst stelt landbouwers of analisten in staat om de prestaties van hun percelen te vergelijken met vergelijkbare percelen in de regio, gebruikmakend van data-analyse. Hieronder volgt een gedetailleerde analyse:

4.2.2.1. Methodologie

fAPAR: (Fraction of Absorbed Photosynthetically Active Radiation) is een maat voor de hoeveelheid licht dat een gewas opneemt voor fotosynthese.

CropSAR: combineert satellietgegevens en andere geo-informatie om deze waarden ruimtelijk en temporeel te berekenen.

CropSAR combineert de sterke punten van **Sentinel-1 (radar)** en **Sentinel-2 (optisch)** door middel van geavanceerde data-integratie en machine learning. Radar, dat onafhankelijk is van weersomstandigheden, zendt een eigen signaal uit en analyseert de terugkaatsing, waardoor consistente gegevens beschikbaar zijn, zelfs bij bewolking. Deze **radarbackscatter** biedt waardevolle informatie over de structuur en het vochtgehalte van vegetatie en de onderliggende bodem. Optische sensoren zoals die van **Sentinel-2 meten biochemische processen**, zoals de fAPAR-waarde, en vullen de radarobservaties aan. CropSAR maakt gebruik van de correlatie tussen deze datasets, met name tijdens groeifasen, om optische datagaten te overbruggen. Met behulp van neurale netwerken en GPU-clusters traint CropSAR een algoritme dat complexe verbanden tussen radar- en optische gegevens leert. Dit resulteert in dagelijkse, cloudvrije fAPAR-tijdreeksen met nauwkeurige onzekerheidsschattingen, waardoor consistente en betrouwbare gewasmonitoring mogelijk wordt.

Het heeft diverse toepassingsgebieden die het tot een waardevolle tool maken voor moderne landbouwpraktijken. Op operationeel niveau wordt de technologie toegepast op veldschaal en geïntegreerd in platforms zoals WatchITgrow, waar het agronomen en boeren bruikbare inzichten biedt om gewasbeheer te optimaliseren en onderbouwde beslissingen te nemen. Daarnaast maakt de geplande uitbreiding naar pixelniveau het mogelijk om variaties binnen een veld te analyseren, een essentieel aspect van precisielandbouw dat bijdraagt aan efficiëntere middeleninzet en hogere opbrengsten. Door zijn vermogen om consistente en cloud-onafhankelijke gegevens te leveren, is CropSAR ook bijzonder geschikt voor landbouw in tropische en gematigde regio's waar bewolking vaak een uitdaging vormt.

4.2.2.2. Toepassing

CropSAR biedt de mogelijkheid om duidelijke dashboardvisualisaties van gewasprestaties, met cloudvrije fAPAR-tijdreeksen en regionale benchmarks. Gebruikers kunnen eenvoudig groei, afwijkingen en trends volgen, terwijl waarschuwingssystemen risico's signaleren en helpen bij het optimaliseren van gewasbeheer.

Daarnaast ondersteunt CropSAR schadeclaims met objectieve data en analyses. Het toont de impact van schade-incidenten via visuele voor- en na-vergelijkingen en afwijkingen van regionale gemiddelden. Deze rapporten versterken claims en maken het proces transparant en efficiënt voor verzekeraars.

4.2.3. Variability Map

4.2.3.1. Crop Performance

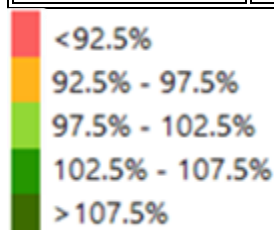
Dit hoofdstuk beschrijft een technologiegedreven benadering voor het berekenen en visualiseren van de variabiliteit in gewasprestaties binnen een veld. De Variability Map biedt landbouwers en analisten de mogelijkheid om de ruimtelijke variatie in gewasgroei op een specifiek moment te analyseren. Dit kan variëren door verschillende factoren zoals bodemtype, hydrologie, plagen, ziektes, of extreme weersomstandigheden zoals droogte, stormen, en overstromingen. Hieronder volgt een gedetailleerde analyse:

4.2.3.2. Methodologie

De basisindex voor het berekenen van de variabiliteitskaarten is fAPAR (Fraction of Absorbed Photosynthetically Active Radiation), een maat voor het licht dat door gewassen wordt

opgenomen voor fotosynthese. fAPAR wordt afgeleid van Sentinel-2 satellietbeelden met een ruimtelijke resolutie van 10 meter. Voor elke wolkvrije satellietafbeelding wordt de fAPAR-waarde van elke pixel vergeleken met de mediaanwaarde van de fAPAR voor het hele veld. Het resultaat is een geotiff-afbeelding die de afwijkingen van de fAPAR-waarden toont. Deze afwijkingen worden gecategoriseerd en gevisualiseerd in een kleurenkaart met vijf verschillende klassen, variërend van rood (onder de mediaan) tot donker groen (boven de mediaan).

De bereik	De klas	De kleur
-85%	1	De rode
85-95%	2	De oranje
95-105%	3	Licht groen
105-115% van de	4	Donker groen
€ 115%	5	'donkerste' groen



Deze aanpak maakt gebruik van de capaciteit van satellietbeelden om gewasvariaties binnen het veld te detecteren. Dit geeft boeren gedetailleerde informatie over waar in hun veld de gewasgroei achterblijft of juist uitblinkt, waardoor ze gerichte acties kunnen ondernemen.

4.2.3.3. Toepassing

De Variability Map biedt verschillende toepassingen die landbouwers kunnen helpen bij het nemen van informele beslissingen over hun gewassen. De kaarten kunnen gebruikt worden voor variabele bemesting of irrigatie, waarbij de dosis kunstmest of water per locatie in het veld wordt aangepast op basis van de waargenomen variaties in gewasgroei. Dit helpt bij het optimaliseren van middelen en het verbeteren van de gewasopbrengst door te focussen op de gebieden die het meeste aandacht nodig hebben. Daarnaast kunnen de kaarten ook helpen bij het monitoren van afwijkingen die mogelijk duiden op schades door plagen, ziektes of extreme weersomstandigheden, en dienen ze als basis voor gerichte beheersmaatregelen.

4.2.4. Yield Potential Map

Opbrengstpotentiekaarten tonen de ruimtelijke variatie in gewasprestaties binnen een veld over een heel seizoen. Een boer kan deze opbrengst potentiële kaarten gebruiken om de geschiedenis van een veld te bekijken. Terugkerende of goede of slechte zones worden meestal veroorzaakt door verschillen in bodem, omringende vegetatie of hydrologie. Tegelijkertijd kan meer abrupte en onverwachte veldvariatie te wijten zijn aan extreme weersomstandigheden zoals droogte, hagel, storm of schade aan overstromingen.

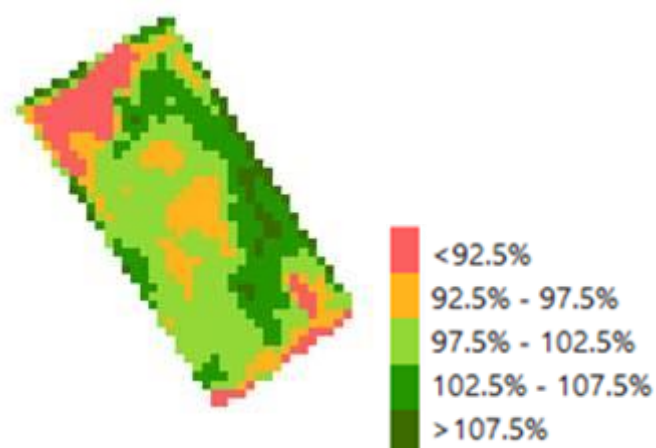
4.2.4.1. Methodologie

De basisindex voor het berekenen van de opbrengstpoten is fAPAR, de fractie van geabsorbeerde fotosynthetisch actieve straling afgeleid van Sentinel-2 satellietbeelden met een ruimtelijke resolutie van 10m. Om de kaarten te berekenen, gebruiken we alle cloud-free afbeeldingen die zijn verkregen tussen volledige opkomst en het begin van crop senescence wanneer de gemiddelde fAPAR-waarde van het veld 0,5 is.

Voor deze afbeeldingen vergelijken we de fAPAR-waarde van elke pixel met de gemiddelde fAPAR-waarde van het veld (pixelwaarden worden uitgedrukt in % van de mediaan) en berekenen vervolgens de gemiddelde afwijking over het seizoen. Deze gemiddelde afwijkingen worden als GeoTIFF-beelden ter beschikking gesteld.

Ten slotte worden de afwijkingen ingedeeld in vijf categorieën op basis van hun relevantie en worden kleurkaarten gegenereerd.

De bereik	De klas	De kleur
> 92,5%	1	De rode
92,5 tot 97,5%	2	De oranje
97,5-102,5%	3	Licht groen
102,5-107,5%	4	Donker groen
> 107,5%	5	'donkerste' groen



In de rode en oranje zones zijn de lagere fAPAR-waarden gedurende het seizoen te vinden; in de groene en donkergroene zones zijn de fAPAR-waarden (veel) hoger dan de mediane waarde. We gaan ervan uit dat het rendementspotentieel hoger is in de donkergroene zones dan in de oranje en rode zones.

4.2.4.2. Toepassing

Een boer kan deze variabiliteitskaarten gebruiken om te controleren op afwijkingen of ze kunnen worden gebruikt als input voor bemesting of irrigatie met variabele rente, om de dosis mest of water aan te passen aan de ruimtelijke variatie binnen het veld.

4.2.4.3. Useful links

[Terrascope Cataloge](#) – Catalogus van de hele EOplaza

[Anomaly Detection](#) – Regional Benchmarking service van terrascope om variaties in percelen te indentificeren.

4.3. Analyse kaartintegratie

CropWatchers heeft gekozen voor een open-source kaartintegratie vanwege de kosteneffectiviteit en brede ondersteuning door de ontwikkelaarsgemeenschap. Na een vergelijking tussen Leaflet en OpenLayers is besloten om Leaflet te implementeren in de toepassing.

4.3.1. Vergelijking tussen Leaflet en OpenLayer

Criteria	Leaflet	OpenLayers
Gebruikersgemak	Kleine leercurve	Complexere setup, GIS-kennis vereist
WMS-ondersteuning	Ondersteund met eenvoudige code	Uitgebreide ondersteuning
Klikfunctionaliteiten	Eenvoudig met ingebouwde events	Mogelijk, maar complexer
GeoJSON-ondersteuning	Native ondersteuning	Native + meer formaten
React-integratie	Uitstekend met React-Leaflet	React-bindings beschikbaar
Documentatie & Community	Groot, actief, veel tutorials	Kleinere, complexere documentatie
Performance	Lichtgewicht, ideaal voor webapps	Zwaarder bij complexe kaarten
Licentie	Open-source, gratis	Open-source, gratis

(Geoapify, 2019)

4.3.2. Conclusie waarom Leaflet

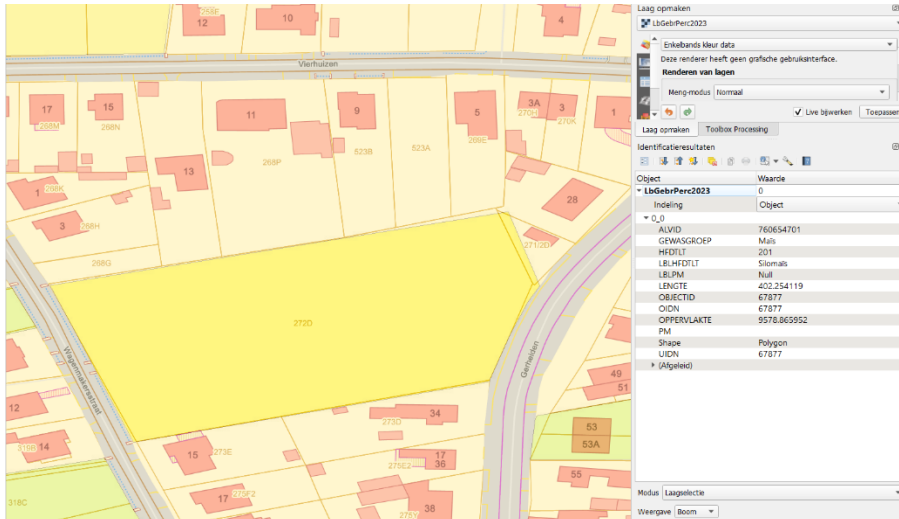
CropWatchers heeft gekozen voor **Leaflet** vanwege de volgende redenen:

- **Lichtgewicht:** Minder code en snellere integratie in de toepassing.
- **WMS- en GeoJSON-ondersteuning:** Onmisbaar voor het werken met landbouwpercelen en AI-resultaten.
- **Compatibiliteit met React:** Dankzij de bibliotheek **React-Leaflet** is Leaflet eenvoudig te integreren in de React-gebaseerde toepassing.
- **Actieve Community en Documentatie:** Uitgebreide documentatie, tutorials en een actieve ontwikkelaarsgemeenschap versnellen de implementatie.

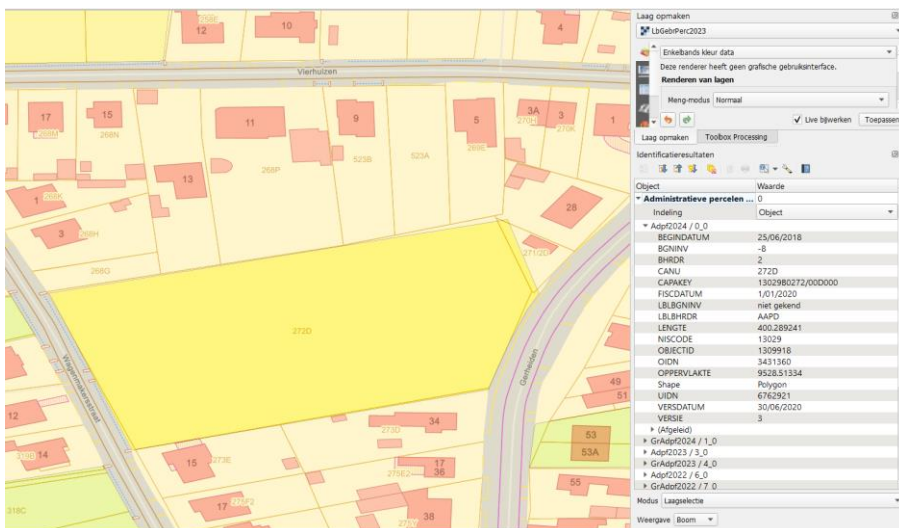
4.3.3. Test case

4.3.3.1. QGIS

In eerste instantie hebben we de WMS-lagen van de Vlaamse Overheid getest. Deze testen zijn uitgevoerd met QGIS, een open source geografisch informatiesysteem (GIS).



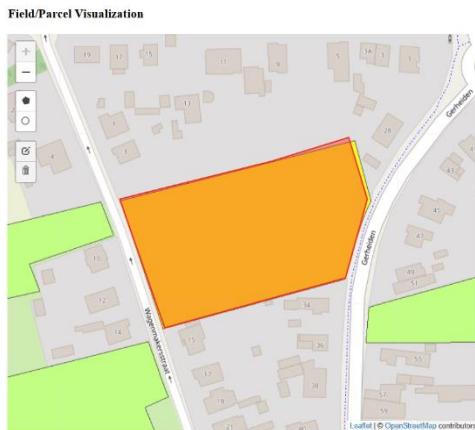
LbGebrPerc2023 – Landbouwgebruikspercelen LV, 2023



Adpf - Administratieve percelen fiscaal

4.3.3.2. Leaflet

In dit Leaflet-voorbeeld in de browser is gebruikgemaakt van de WMS voor Landbouwgebruikspercelen om de landbouwpercelen op een basiskaart weer te geven en Administratieve percelen fiscaal om informatie over een perceel op te vragen.



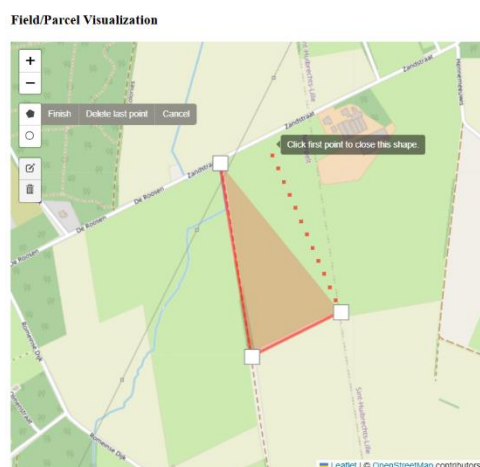
Integratie Leaflet en WMS Vlaanderen

Wanneer een gebruiker op een perceel klikt, wordt de volgende data weergegeven (Adpf - Administratieve percelen fiscaal). Deze data kunnen gebruikt worden om extra informatie op te vragen, zoals de shapeGml van het perceel.

```
<FeatureInfoResponse xmlns:esri_wms="http://www.esri.com/wms" xmlns="http://www.esri.com/wms">
  <FIELDS OBJECTID="1309918" UIDN="6762921" OIDN="3431360" VERSIE="3" BEGINDATUM="25/06/2018"
    VERSDATUM="30/06/2020" CAPAKEY="1302980272/00D000" NISCODE="13029" FISCDATUM="1/01/2020"
    BHRDR="2" LBLBHRDR="AAPD" CANU="272D" BGNINV="-8" LBLBGNINV="niet gekend" Shape="Polygon"
    OPPERVLAKE="9528.51334" LENGTE="400.289241"></FIELDS>
</FeatureInfoResponse>
```

Informatie perceel via Leaflet integratie

Als een perceel niet terug te vinden is via de WMS Vlaanderen, heeft de gebruiker de mogelijkheid om het perceel zelf te tekenen.



Perceel tekenen

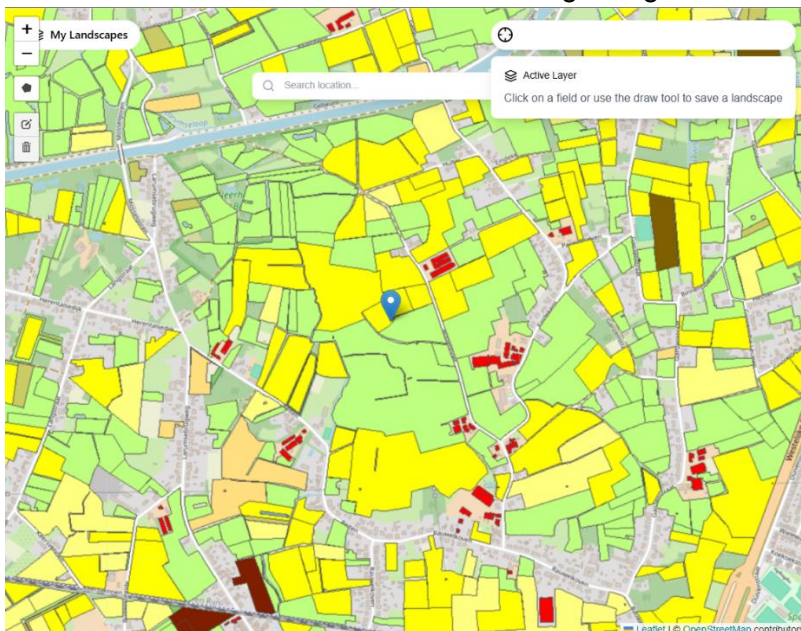
4.3.4. React-Leaflet

React Leaflet is een open-source bibliotheek die een set React-componenten biedt voor Leaflet-kaarten. Leaflet is een krachtige, open-source kaartenbibliotheek waarmee ontwikkelaars interactieve kaarten voor webapplicaties kunnen maken. Met React Leaflet kunnen Leaflet-kaarten eenvoudig worden geïntegreerd in React-toepassingen, zodat ontwikkelaars de kracht van Leaflet kunnen benutten met de eenvoud en het gebruiksgemak van React. React Leaflet biedt een verscheidenheid aan componenten voor Leaflet-kaarten, waaronder het Map-component, Marker-component, Popup-component en nog veel meer. Met deze componenten kun je eenvoudig functies aan je Leaflet-kaart toevoegen, zoals markers, pop-ups en aangepaste pictogrammen. Een van de belangrijkste voordelen van React Leaflet is dat je op een “React-achtige” manier met Leaflet-maps kunt werken. Dit betekent dat je bekende React-concepten, zoals componenten, props en state, kunt gebruiken om je Leaflet-map op te bouwen. Dit maakt React Leaflet een uitstekende keuze voor ontwikkelaars die al bekend zijn met React.

(react-leaflet, n.d.)

4.3.4.1. Test case React-Leaflet

In dit voorbeeld ziet u hoe React-Leaflet is geïntegreerd in een React-applicatie.



Integratie React-Leaflet

4.4. Analyse LLM

4.4.1. Privacy rechten sociale media

De beschikbare data van de API's mogen gebruikt worden. Data gaan scrapen van platforms zoals Facebook, ... zal niet gaan, omdat de kans groot is dat de IP-adressen geblokkeerd gaan worden.

4.4.2. Sociale Media data krijgen

De beste manier om toegang te krijgen tot de data is om deze van sociale media te halen via de API van het platform.

Facebook Graph API: <https://developers.facebook.com/docs/graph-api/> (Meta, Graph API, 2024)

- Weinig informatie te verkrijgen.
- Toestemming van gebruikers hebben

X/Twitter API: <https://developer.x.com/en/docs/x-api> (X, sd)

- Posts kunnen opgehaald worden van een bepaalde regio.
- Post kunnen opgehaald worden op basis van tags.
- **Types:**
 - o **Gratis:**
 - Beperkte toegang (afgelopen 7 dagen).
 - Beperking op aantal API-verzoeken per maand.
 - o **Enterprise:**
 - Onbeperkte toegang tot historische en real-time gegevens.
 - Ideaal voor grote bedrijven of onderzoeksprojecten.

Instagram API: <https://developers.facebook.com/docs/instagram-platform/instagram-api-with-instagram-login> (Meta, Instagram API with Instagram Login, sd)

- Weinig informatie te verkrijgen.
- Je hebt specifieke toestemmingen nodig via het Facebook Developer-platform.
- Alleen eigen informatie van accounts.

TikTok Research API: <https://developers.tiktok.com/products/research-api/> (TikTok, sd)

- Content kan van tiktok gehaald worden.
- Moet een account voor aangemaakt worden en het gebruik moet goedgekeurd worden met geldige locatie, bedrijf en reden voor gebruik.

4.4.3. (Betrouwbare) Twitter/X accounts

België:

- <https://x.com/HetweerinBelgie>
- <https://x.com/meteobenl>
- <https://x.com/frankdeboosere>
- <https://x.com/BuienradarBE>
- https://x.com/weerenradar_be

West/Oost Vlaanderen:

- <https://x.com/GeertNaessens>

4.4.4. Media data

Mogelijke opties om de data van de mediabronnen te verkrijgen die te maken hebben met weerproblemen in Vlaanderen.

Scraping:

- Scrape data direct van populaire Vlaamse nieuwswebsites zoals:
 - o VRT NWS, De Standaard, Het Nieuwsblad, HLN, De Morgen.
- Gebruik scrapingtools zoals BeautifulSoup of Scrapy.
- Specifieke pagina's over weer en rampen kunnen worden gescraped voor relevante artikelen.
- Zorg dat je voldoet aan de robots.txt-regels en privacyrichtlijnen

SerpAPI:

- Google News API:
 - o Scrape resultaten van Google News pagina
 - o Parameters die kunnen meegegeven worden:
 - Query
 - Gl: landcode (be, uk, us, ...)
 - Hl: taal
 - Topic token: topic (weer, business, ...)
 - o Output: JSON
 - o Kosten:
 - 100 gratis searches
 - €75/maand à 5 000 searches
 - o Link: <https://serpapi.com/google-news-api> (SerpApi, sd)

Custom Search JSON API:

- Parameters:
 - o dateRestrict:
 - **d[number]**: vraagt resultaten op van het opgegeven aantal afgelopen dagen.
 - **w[number]**: vraagt resultaten op van het opgegeven aantal afgelopen dagen.
 - **m[number]**: vraagt resultaten op van het opgegeven aantal afgelopen maanden.
 - **y[number]**: vraagt resultaten op van het opgegeven aantal afgelopen jaren.
 - o Gl: landcode (be, uk, us, ...)
 - o Hl: taal
 - o Q: Query
- Kosten:
 - o 100 gratis searches per dag
 - o \$5 per 1 000 query's
 - o Max 10 000 query's per dag
- Link: <https://developers.google.com/custom-search/v1/overview> (Google, Custom Search JSON API, sd)

Conclusie:

Voor Cropwatchers is Custom Search JSON API het beste, omdat we hier het meeste mee kunnen doen met de gratis versie.

4.4.5. Bestaande LLM's

Criteria	GPT	BERT	T5	PaLM	LLaMA
Prijs (40%)	2	5	5	3	5
Hardware (20%)	2	4	4	3	5
Gegevens beveiliging (25%)	2 (OpenAI krijgt alle gegevens)	4	5	3 (gegevens gaan door google cloud)	5
API ondersteuning (15%)	5	3	4	5	4
Totaal	2,45	4,25	4,65	3,3	4,85

GPT (Generative Pretrained Transformer)

Ontwikkeld door: OpenAI

Voorbeelden: GPT-3, GPT-4

Toepassingen: Deze modellen worden vaak gebruikt in chatbots (zoals ChatGPT), tekstgeneratie, vertalingen en andere NLP-taken.

Kenmerken: GPT-modellen zijn autoregressieve taalmodellen die tekst genereren op basis van voorafgaande tekst, waarbij ze getraind zijn op enorme hoeveelheden tekstdata.

Link:

- <https://platform.openai.com/docs/overview> (OpenAI, sd)

Kosten:

- **GPT-4 API:**
 - Context van 8k tokens: \$0.03 per 1k tokens (prompt) en \$0.06 per 1k tokens (output).
 - Context van 32k tokens: \$0.06 per 1k tokens (prompt) en \$0.12 per 1k tokens (output).
- **GPT-3.5 API:**
 - \$0.0015 per 1k tokens (prompt) en \$0.002 per 1k tokens (output).
- **Abonnement ChatGPT Plus:**
 - \$20 per maand voor gebruik van GPT-4 in de ChatGPT-app.

BERT (Bidirectional Encoder Representations from Transformers)

Ontwikkeld door: Google

Toepassingen: BERT wordt vaak gebruikt voor NLP-taken zoals vraag-antwoord-systemen, sentimentanalyse en tekstclassificatie.

Kenmerken: BERT is bidirectioneel, wat betekent dat het zowel de context aan de linker- als rechterkant van een woord begrijpt, in plaats van alleen de context van links naar rechts (zoals bij GPT).

Link:

- <https://github.com/google-research/bert> (Google, bert, sd)
- https://huggingface.co/docs/transformers/model_doc/bert (Face, BERT, sd)

Kosten:

- **Open-source:** BERT zelf is gratis beschikbaar op platforms zoals GitHub en Hugging Face.

T5 (Text-to-Text Transfer Transformer)

Ontwikkeld door: Google

Toepassingen: T5 is ontworpen voor een breed scala aan NLP-taken, zoals vertalen, samenvatten, en zelfs het beantwoorden van vragen.

Kenmerken: T5 behandelt alle taken als een tekst-naar-tekstprobleem, wat betekent dat elke input en output wordt beschouwd als een tekstuele string.

Link:

- <https://github.com/google-research/text-to-text-transfer-transformer> (Google, text-to-text-transformers, sd)
- https://huggingface.co/docs/transformers/model_doc/t5 (Face, T5, sd)

Kosten:

- **Open-source:** Net als BERT is T5 open-source en gratis beschikbaar.

PaLM (Pathways Language Model)

Ontwikkeld door: Google

Toepassingen:

PaLM is gericht op een breder scala van natuurlijke taalverwerkingsproblemen, met name diegene die meer menselijke redenering en complexe taken vereisen.

Kenmerken: Het model maakt gebruik van het Pathways-systeem van Google, dat efficiënter is in het trainen van modellen voor meerdere taken.

Link:

- <https://cloud.google.com/vertex-ai/generative-ai/docs/language-model-overview#palm-api> (Google, Overview of language models, 2024)

Kosten:

- **API-prijzen:**
 - Prijzen variëren op basis van de grootte van het model (bijvoorbeeld PaLM 2).
 - Vertex AI rekent doorgaans \$0.05-\$0.20 per 1k tokens, afhankelijk van het gebruik (standaardaanroep vs. fine-tuning).
- **Enterprise-opties:**
 - Kosten kunnen oplopen bij grotere inzet in een zakelijke omgeving, afhankelijk van het volume.

LLaMA (Large Language Model Meta AI)

Ontwikkeld door: Meta (voorheen Facebook)

Toepassingen: LLaMA is bedoeld voor wetenschappelijk onderzoek en toepassingen in de AI-gemeenschap.

Kenmerken: Het is ontworpen om kleinere, efficiëntere modellen te leveren die toegang bieden tot krachtige taalverwerkingscapaciteiten.

Link:

- <https://www.llama.com> (Meta, Llama, sd)
- <https://www.llama.com/docs/overview> (Meta, Get started with Llama, sd)
- <https://huggingface.co/meta-llama> (Llama, sd)

Kosten:

- **Open-source:** Gratis beschikbaar voor niet-commerciële doeleinden.
- **Hardwarevereisten:**
 - Kleinere modellen (LLaMA-7B) kunnen draaien op een enkele krachtige GPU (32GB VRAM).
 - Grotere modellen (LLaMA-65B) vereisen clusters van GPU's of TPUs, wat duurder is.

Conclusie:

Het LLaMA-model is voor Cropwatchers de meest geschikte keuze, omdat het gratis beschikbaar is en wetenschappelijk onderbouwd. Om dit model te gebruiken, maakt Cropwatchers gebruik van Groq, een platform waarmee we het model kunnen inzetten via de

Groq API. Deze API is eveneens gratis beschikbaar, wat het een kostenefficiënte oplossing maakt.

Groq biedt een krachtige infrastructuur die geoptimaliseerd is voor AI-modellen zoals LLaMA. Hierdoor kan Cropwatchers profiteren van hoge prestaties zonder complexe hardware zelf te hoeven beheren. Deze combinatie van LLaMA en Groq stelt Cropwatchers in staat om hoogwaardige AI-functionaliteiten te benutten op een toegankelijke en betaalbare manier.

4.4.6. Data ophalen voor LLM

RAG (Retrieval Augmented Generation)

Documentatie MongoDB: <https://www.mongodb.com/resources/basics/artificial-intelligence/retrieval-augmented-generation> (MongoDB, sd)

Hoe werkt RAG?

Retrieval (Ophalen):

Het systeem haalt relevante informatie op uit een externe bron, zoals een database, documentenset, of API. Deze bron bevat feiten, tabellen of tekst die nodig zijn voor het antwoord.

Denk aan een zoekproces vergelijkbaar met Google, maar specifiek gericht op je eigen data.

Augmentation (Aanvullen):

De opgehaalde informatie wordt gebruikt als context. Dit wordt gecombineerd met een taalmodel (zoals GPT-3) dat deze informatie begrijpt en gebruikt om een antwoord te formuleren.

Generation (Genereren):

Het taalmodel genereert een antwoord gebaseerd op de opgehaalde gegevens én zijn eigen taalbegrip. Hierdoor krijg je een meer accuraat, inhoudelijk correct en goed geformuleerd antwoord.

Waarom is RAG nuttig?

Betere nauwkeurigheid: Informatie komt uit betrouwbare bronnen, wat hallucinerende antwoorden (onjuiste antwoorden van AI) beperkt.

Flexibiliteit: Kan zowel algemene als specifieke vragen beantwoorden, afhankelijk van de beschikbare gegevens.

Efficiëntie: Door relevante informatie eerst op te halen, hoeft het taalmodel niet alles te "onthouden", wat de prestaties verhoogt.

4.4.7. Formulier invullen

Link formulier:

https://assets.vlaanderen.be/image/upload/v1694696285/Meldingsformulier_Vlaams_Rampe_nfonds_am7ogy.pdf

4.5. Analyse Visualisation tools

4.5.1. Integratie

De dashboards die we ontwikkelen, zullen worden afgestemd op de specifieke behoeften van de verschillende gebruikersgroepen. Elke gebruiker krijgt inzichten die relevant zijn voor zijn of haar functie. Hieronder volgt een overzicht van de geplande dashboards:

Boeren:

- Inzichten in schadepercentages, teeltpercentages, wateroverlast in het gebied en weersvoorspellingen.
- Deze informatie zou boeren kunnen helpen bij het maken van beter geïnformeerde beslissingen.

Consultants:

- Consultants krijgen toegang tot dezelfde dashboards als de boeren, mits dit door de boer is toegestaan.
- Daarnaast kunnen zij een overzicht bekijken van de gegevens van meerdere boeren om trends en patronen te analyseren en beter advies te geven.

Staat:

- De staat zal inzicht krijgen in overkoepelende gegevens zoals schadeclaims en weerdata, om beleidsbeslissingen te ondersteunen.

4.5.2. Power BI

Bij het ontwikkelen van rapporten voor onze applicatie kan Cropwatchers Power BI gebruiken om interactieve dashboards te maken en in te bedden in een externe website. Echter, wanneer gebruikers onze website bezoeken en het rapport willen bekijken, zullen ze eerst moeten inloggen met een Microsoft-account dat beschikt over een Power BI-licentie. Aangezien de meeste van onze gebruikers waarschijnlijk niet over een dergelijke licentie beschikken, is Power BI geen geschikte optie voor ons.



4.5.3. Python Plotly

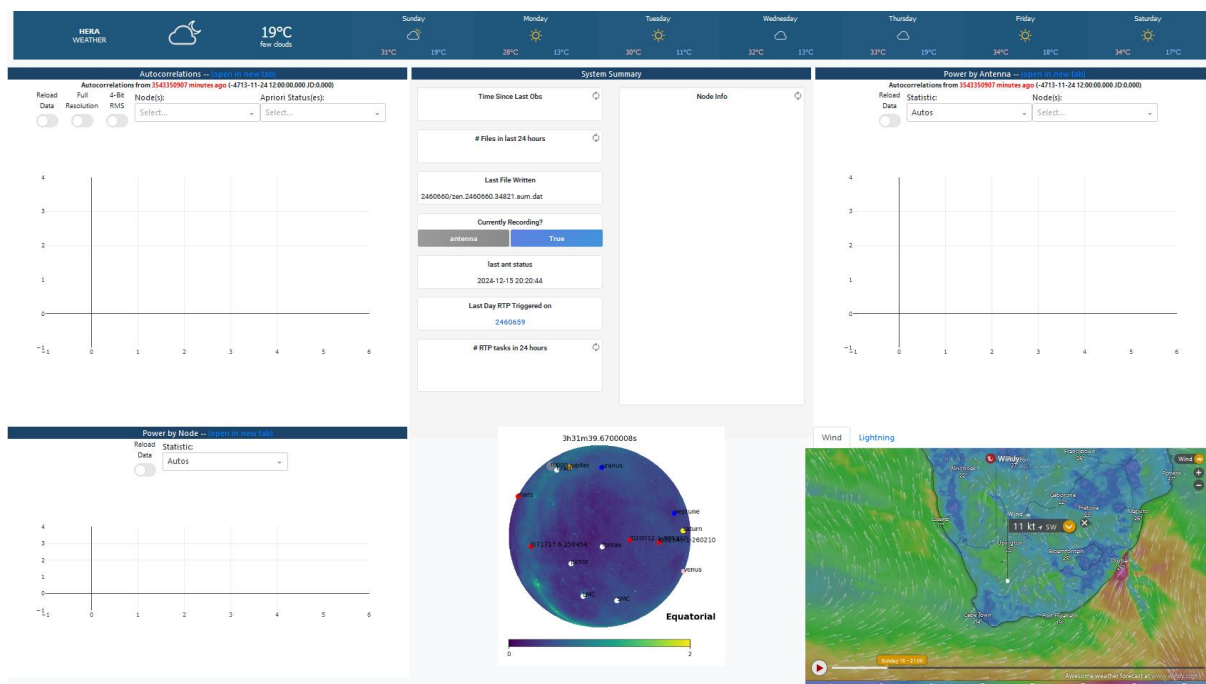


Plotly is een **datavisualisatiebibliotheek** dat volstaat om interactieve visualisaties te creëren. Het ondersteunt een breed scala aan grafiektypes, integreert naadloos met tools zoals Pandas, en biedt uitgebreide aanpassingsmogelijkheden aan. Hierdoor is het ideaal voor het weergeven van datasets voor free-users. ---Link: [Plotly documentatie](#). (Plotly, sd)

4.5.3.1. Waarom Plotly

Plotly is een open-source integratie, wat betekent dat het gratis toegankelijk is voor gebruikers. Het biedt ondersteuning voor meerdere programmeertalen, waaronder **Python**, **Javascript**, **R**, **etc**, waardoor het breed inzetbaar is in diverse omgevingen.

Plotly meest onderscheidende eigenschap is de mogelijkheid om **interactieve grafieken** te maken. Deze grafieken stellen gebruikers in staat om direct met data te interacteren, bijvoorbeeld door in te zoomen, gegevenspunten te selecteren of over specifieke onderdelen te hovern om extra details te zien. Hierdoor krijgen de gebruikers een beter beeld van context bij het analyseren van gegevens.



Daarnaast ondersteunt Plotly een breed scala aan **visualisatietypes**, van eenvoudige lijndiagrammen en staafdiagrammen tot geografische kaarten en tijdreeksanalyses. Deze veelzijdigheid maakt het mogelijk om visualisaties te creëren die zijn afgestemd op specifieke rapportagedoelen.

Een ander voordeel van Plotly is de mogelijkheid om visualisaties te integreren in **webgebaseerde dashboards**, met behulp van frameworks zoals **Dash**. Dit maakt het eenvoudig om krachtige, interactieve BI-dashboards te bouwen die toegankelijk zijn via een webbrowser. Hierdoor kunnen belanghebbenden in real-time gegevens bekijken en analyseren, ongeacht hun locatie.

Plotly is een uitstekende keuze voor dit project vanwege de ondersteuning voor verschillende programmeertalen, waaronder **Python** en **JavaScript**. Dit maakt het flexibel inzetbaar en geschikt voor diverse technische toepassingen. Bovendien is de documentatie van Plotly zeer uitgebreid, wat een groot voordeel biedt tijdens de implementatie. De vele beschikbare voorbeelden, zoals GeoSpatial-visualisaties en API-dashboards, geven al een goed beeld van wat mogelijk is met onze data en vormen een solide basis voor verdere ontwikkeling.

4.5.4. Chart.js

Chart.js is een open-source JavaScript-bibliotheek die wordt gebruikt om eenvoudige maar aantrekkelijk uitziende grafieken te maken in webapps. Het is lightweight, flexible en ideaal voor projecten die snelle visualisaties vereisen zonder een uitgebreide technische setup.

4.5.4.1. Waarom Chart.js

Het is een eenvoudige intuïtieve API, waardoor het toegankelijk is voor zowel beginners als ervaren developers. Grafieken kunnen snel worden opgezet met minimale configuratie, terwijl de mogelijkheid bestaat om ze verder aan te passen voor complexere toepassingen



Chart.js biedt een breed scala aan grafiektypen. Daarnaast ondersteunt het ook gecombineerde grafieken (bijvoorbeeld een lijn- en staafgrafiek in één), wat het ideaal maakt voor het visualiseren van diverse gegevens in één overzicht. De grafieken zijn standaard interactief. Gebruikers kunnen hoveren om details te bekijken, gegevenspunten selecteren, of met behulp van plugins functionaliteiten zoals zoomen en pannen toevoegen. Deze interactiviteit maakt het verkennen van data eenvoudiger.



Met Chart.js is het eenvoudig om stijlen aan te passen, zoals kleuren, lettertype, labels en animaties. De bibliotheek ondersteunt ook plugins waarmee extra functionaliteiten, zoals annotaties of live datastreams, kunnen worden toegevoegd. De visualisaties worden efficiënt worden weergegeven, zelfs bij grote dataset. De compacte aard van de bibliotheek zorgt ervoor dat het weinig belasting oplevert voor webapplicaties. (Chart.js, sd)

Conclusie: Chart.js is een uitstekende keuze voor eenvoudige, aantrekkelijke visualisaties in webapplicaties, terwijl Plotly beter geschikt is voor geavanceerde, interactieve dashboards en complexe data-analyse.

5. Deployment

5.1. FARM stack

Dit onderdeel bespreekt de keuze voor de FARM-stack (FastAPI, React en MongoDB) als ook de technologische basis voor dit project, gericht op het realiseren van schaalbare, asynchrone communicatie en flexibele data-opslag. FastAPI is gekozen voor de back-end vanwege de hoge snelheid, prestaties en de uitstekende integratiecapaciteiten met machine learning-modellen. In combinatie met MongoDB, wordt de databaselaag ondersteund door ODMantic, wat asynchrone mogelijkheden biedt en de integratie van ongestructureerde en geospatiale gegevens vereenvoudigt. De front-end is ontwikkeld met React, dat de basis biedt voor een dynamische gebruikersinterface en de mogelijkheid om een Progressive Web App (PWA) te realiseren, die zowel online als deels offline werkt. De keuzes voor deze technologieën zijn zorgvuldig onderbouwd, met een focus op hun geschiktheid voor de projectdoelen en hun voordelen ten opzichte van alternatieven.

5.1.1. Back-end (FastAPI)

De back-end verwerkt gegevens en biedt een API waarmee de front-end kan communiceren. Voor dit project heeft CropWatchers gekozen voor FastAPI, een modern Python-webframework dat uitstekend presteert bij AI-services en asynchrone toepassingen. FastAPI is ook strategisch gekozen vanwege de ervaring van studenten van het keuzetraject Artificial Intelligence met deze technologie. FastAPI biedt een optimale balans tussen snelheid, gebruikersgemak en functionaliteit wat essentieel is voor dit project.

5.1.1.1. Waarom FastAPI?

- **Asynchrone verwerking:** FastAPI biedt snelle prestaties en efficiënte verwerking van grote hoeveelheden data, wat essentieel is voor AI en machine learning-applicaties.
- **Eenvoudige integratie:** De technologie is zeer geschikt voor Python-gebaseerde toepassingen, waardoor het eenvoudig is om machine learning-modellen te integreren.
- **Automatische documentatie:** FastAPI genereert automatisch API-documentatie volgens de OpenAPI-specificatie, wat de ontwikkeling versnelt.
- **Schaalbaarheid:** Zowel voor kleine als grootschalige toepassingen is FastAPI uitstekend geschikt, wat de toekomstige uitbreiding van het project mogelijk maakt.

5.1.1.2. FastAPI VS Django VS Flask

Criterium	Gewicht	FastAPI	Django	Flask
Architectuur	-	ASGI	MVT	WSGI
Prestaties	20%	9	6	6
Gebruikersvriendelijkheid	20%	8	5	5
Documentatie	20%	9	7	7
Leercurve	10%	8	5	5
Schaalbaarheid	20%	9	9	7
Teamervaring	10%	6	1	1
Gewogen totaal	100%	8,5	6,1	5,7

(Lee, 2024)

5.1.1.3. Gebruik van een ORM/ODM

ODMantic is een asynchrone Object-Document Mapper (ODM) die speciaal is ontworpen voor MongoDB en naadloos integreert met FastAPI. Dit maakt het een uitstekende keuze voor dit project, aangezien het asynchrone mogelijkheden biedt en de integratie met MongoDB vereenvoudigt. De object-georiënteerde aanpak maakt de databaselaag eenvoudiger, waardoor de ontwikkeltijd korter wordt en de code makkelijker te onderhouden is.

(Smith, 2024)

5.1.2. Front-end (React)

De front-end biedt de gebruikersinterface. Voor dit project heeft CropWatchers gekozen voor React. Deze keuze is gebaseerd op de ervaring van studenten van het keuzetraject Application Development en de beschikbaarheid van handige bibliotheken zoals React-Leaflet die een eenvoudige integratie van interactieve kaarten mogelijk maakt. Daarnaast wordt gekeken naar de implementatie van een Progressive Web App (PWA). Met een PWA kan dezelfde codebase worden gebruikt om een app-achtige ervaring te bieden op verschillende apparaten. Dit biedt meerdere voordelen: de applicatie kan deels offline functioneren, is eenvoudig te installeren via de browser en elimineert de noodzaak om afzonderlijke apps te ontwikkelen voor iOS en Android. Dit maakt het een tijdbesparende oplossing en vergroot ook de toegankelijkheid voor mobiele gebruikers. Indien er na de initiële implementatie nog tijd beschikbaar is, kan de applicatie in een later stadium worden uitgebreid met een native mobiele app voor extra functionaliteiten. Naast React kan er ook gebruik gemaakt worden van Tailwind CSS om de ontwikkeling van de gebruikersinterface te versnellen.

5.1.2.1. Waarom React:

- **Componentgebaseerd:** Code is herbruikbaar, wat het onderhoud en uitbreidingen vereenvoudigt.
- **Hoge snelheid en prestaties:** Dankzij de virtuele DOM is React bijzonder efficiënt en geschikt voor dynamische interfaces.
- **Progressive Web App (PWA):** Met een PWA-architectuur kan de applicatie op verschillende apparaten draaien, zowel online als deels offline zonder dat aparte native apps nodig zijn voor iOS en Android.
- **Uitgebreid ecosysteem:** De grote community en beschikbare bibliotheken, zoals React-Leaflet voor interactieve kaarten, zorgen voor een snelle ontwikkeling van de interface.

5.1.2.2. React VS Angular:

Criterium	Gewicht	React	Angular
Architectuur	-	Componentgebaseerd	Model-View-Controller (MVC)
Prestaties	20%	9	7
Gebruikersvriendelijkheid	20%	8	6
Documentatie	20%	9	9
Leercurve	10%	8	5
Schaalbaarheid	20%	8	9
Teamervaring	10%	8	9
Gewogen totaal	100%	7,7	6,5

(geeksforgeeks, 2024)

5.1.3. Database (MongoDB)

MongoDB is gekozen voor zijn flexibiliteit en schaalbaarheid wat ideaal is voor het opslaan van ongestructureerde of semi-gestructureerde (NoSQL) gegevens. De geospatiale ondersteuning is van belang voor het verwerken van locatie gebonden gegevens zoals percelen/ landbouwgebieden. MongoDB biedt ingebouwde ondersteuning voor ruimtelijke queries en analyses, wat het bijzonder geschikt maakt voor het werken met geodata.

5.1.3.1. Waarom MongoDB:

- **Flexibele structuur:** JSON-achtige documenten.
- **Geospatiale ondersteuning:** Biedt ondersteuning voor ruimtelijke query's en analyses wat noodzakelijk is voor het opslaan van percelen.
- **Schaalbaarheid:** Geschikt voor grote hoeveelheden aan gegevens en hoge verkeersvolumes.

5.1.3.2. MongoDB VS PostgreSQL VS MySQL:

Criterium	Gewicht	MongoDB	PostgreSQL	MySQL
Type	-	NoSQL	SQL	SQL
Schaalbaarheid	20%	9	7	7
Flexibiliteit	20%	9	5	5
Geospatiale ondersteuning	20%	8	8	6
Data-opslag	10%	9	7	7
Prestaties bij grote hoeveelheden data	20%	9	8	8
Geschiktheid voor hoge verkeersvolumes	10%	9	7	7
Gewogen totaal	100%	8,8	7,0	6,6

(DB-Engines, n.d.)

5.2. Hosting

5.2.1. Introductie tot de Big 3 Cloud Providers

Voor dit project hebben we onderzocht welke cloudprovider het meest geschikt is. De drie grootste cloudplatformen, bekend als de Big 3, zijn:

- **Amazon Web Services (AWS)**

AWS is de marktleider en biedt een breed scala aan diensten, waaronder krachtige containerbeheeropties zoals ECS en EKS, en flexibele opslagmogelijkheden. Het platform staat bekend om zijn schaalbaarheid en uitgebreide ecosysteem.

- **Microsoft Azure**

Azure biedt sterke integratie met Microsoft-producten en is aantrekkelijk voor organisaties die al gebruikmaken van Microsoft-technologie. De containerdiensten (zoals AKS) en opslagmogelijkheden zijn goed geïntegreerd in het platform.

- **Google Cloud Platform (GCP)**

GCP is een pionier in Kubernetes en biedt krachtige containerdiensten zoals GKE, met een sterke focus op open source en gebruiksgemak. GCP is ook vaak aantrekkelijk voor projecten met een beperkt budget, dankzij flexibele credits en serverless opties.

5.2.2. Vergelijking: Weighted Ranking Method (WRM)

We hebben een WRM toegepast op zes belangrijke criteria om de platforms objectief te evalueren:

Criterium	Gewicht (%)	AWS	Azure	GCP
Kosten en Credits	25%	5	4	3
Gebruiksgemak	20%	5	4	4
Schaalbaarheid	20%	5	4	4
Eenvoudige Database-integratie	15%	5	4	3
Beveiliging en Compliance	10%	4	5	4
Beschikbaarheid van Tools en Diensten	10%	5	4	4

Cloud Provider	Gewogen totaal
AWS	4.85
Azure	4.25
GCP	3.95

- **Kosten en Credits (25%)**

AWS biedt een aantrekkelijke combinatie van lage kosten voor een proof of concept en \$50 aan startcredits. Dit maakt het bijzonder geschikt voor jouw budget.

- **Gebruiksgemak (20%)**

AWS heeft een gebruiksvriendelijke console en biedt uitgebreide documentatie en tutorials, waardoor het eenvoudiger is om infrastructuur op te zetten.

- **Schaalbaarheid (20%)**

AWS heeft bewezen mogelijkheden voor automatische schaalbaarheid via diensten zoals Auto Scaling Groups en Lambda, wat handig is voor toekomstige productie.

- **Eenvoudige Database-integratie (15%)**

MongoDB Atlas werkt goed samen met AWS, en er zijn weinig verbindingproblemen dankzij ingebouwde integratiemogelijkheden.

- **Beveiliging en Compliance (10%)**

Azure scoort hier iets hoger vanwege de sterke focus op enterprise compliance, maar AWS biedt uitstekende security tools zoals WAF, GuardDuty en IAM.

- **Beschikbaarheid van Tools en Diensten (10%)**

AWS heeft een uitgebreide suite aan tools (bijvoorbeeld CloudWatch en CloudFormation) die je kunt gebruiken voor monitoring en automatisering.

5.2.3. Conclusie

Na vergelijking van de drie providers, kiezen we voor **Amazon Web Services (AWS)** voor de hosting van onze webapplicatie. AWS biedt een optimale balans tussen gebruiksgemak, schaalbaarheid, en flexibele opslagopties, ondanks dat de kosten iets hoger zijn.

Veronderstellingen achter deze keuze:

- **Budget:** Hoewel AWS iets duurder is, ga ik ervan uit dat ik kosten kan beperken door gebruik te maken van spot instances. De beschikbaarheid van \$50 in credits helpt hierbij.
- **Toekomstige Schaalbaarheid:** AWS's bewezen infrastructuur biedt de beste mogelijkheden voor toekomstige groei.

Met deze keuze zijn we voorbereid op een schaalbare en betrouwbare hostingomgeving die binnen de projectdoelen past.

5.3. Infrastructuur

Dit onderdeel geeft een overzicht van de AWS-infrastructuur die we hebben opgezet voor het PoC van dit project. De infrastructuur is ontworpen om een solide basis te bieden. We maken gebruik van een Virtual Private Cloud (VPC) waarin we verschillende netwerklagen implementeren, zoals een public subnet voor de frontend, een private subnet voor de backend en een veilige verbinding met MongoDB Atlas voor de database. Dit zorgt ervoor dat de applicatie goed gescheiden is van externe netwerken, terwijl de communicatie tussen de verschillende onderdelen veilig blijft.

De opzet biedt voldoende bescherming tegen ongewenst verkeer door middel van beveiligingsgroepen, network ACLs en juiste netwerkinstellingen. Ook de VPC Peering zorgt ervoor dat de verschillende netwerkomgevingen effectief kunnen communiceren, zonder toegevingen te doen aan de beveiliging of netwerkkintegriteit.

Hoewel we de schaalbaarheid van het systeem niet volledig hebben geoptimaliseerd, is deze opzet klaar voor een beperkte productieomgeving waar de belangrijkste aspecten zoals beveiliging en netwerkbeheer goed geregeld zijn. Dit maakt het een solide keuze voor het proof of concept, waarbij de applicatie functioneert in een gecontroleerde, veilige en betrouwbare omgeving.

5.3.1. VPC en Subnets

De applicatie draait binnen een Virtual Private Cloud (VPC), die een beveiligde en geïsoleerde omgeving biedt voor de applicatie. Binnen de VPC hebben we twee subnetten:

- **Public Subnet:** Dit subnet is toegankelijk via het internet en bevat de componenten van de frontend, zoals amazon S3 voor statische bestandshosting en CloudFront voor wereldwijde content delivery.
- **Private Subnet:** dit subnet bevat de backend, waarin de FastAPI applicatie draait op een EC2-instance. Deze componenten zijn niet direct toegankelijk vanaf het internet om de beveiliging te verbeteren.

5.3.2. Internet Gateway en NAT Gateway

- **Internet Gateway (IGW):** Dit maakt de communicatie met het internet mogelijk voor de resources in het public subnet.

5.3.3. Routing en Beveiliging

- Route Tables: Verkeer wordt gestuurd via specifieke route tables. Verkeer van het public subnet gaat via de IGW naar het internet. Verkeer vanuit het private subnet wordt via de NGW gestuurd.
- Security Groups: deze fungeren als virtuele firewalls. We beperken de toegang:
- De frontend security group staat alleen HTTP/HTTPS-verkeer toe van internetgebruikers.
- De backend security group staat alleen verkeer toe van de frontend op de API-poorten.
- De MongoDB Atlas security group staat alleen verbindingen toe van de backend, zodat de database veilig is.

5.3.4. MongoDB Atlas

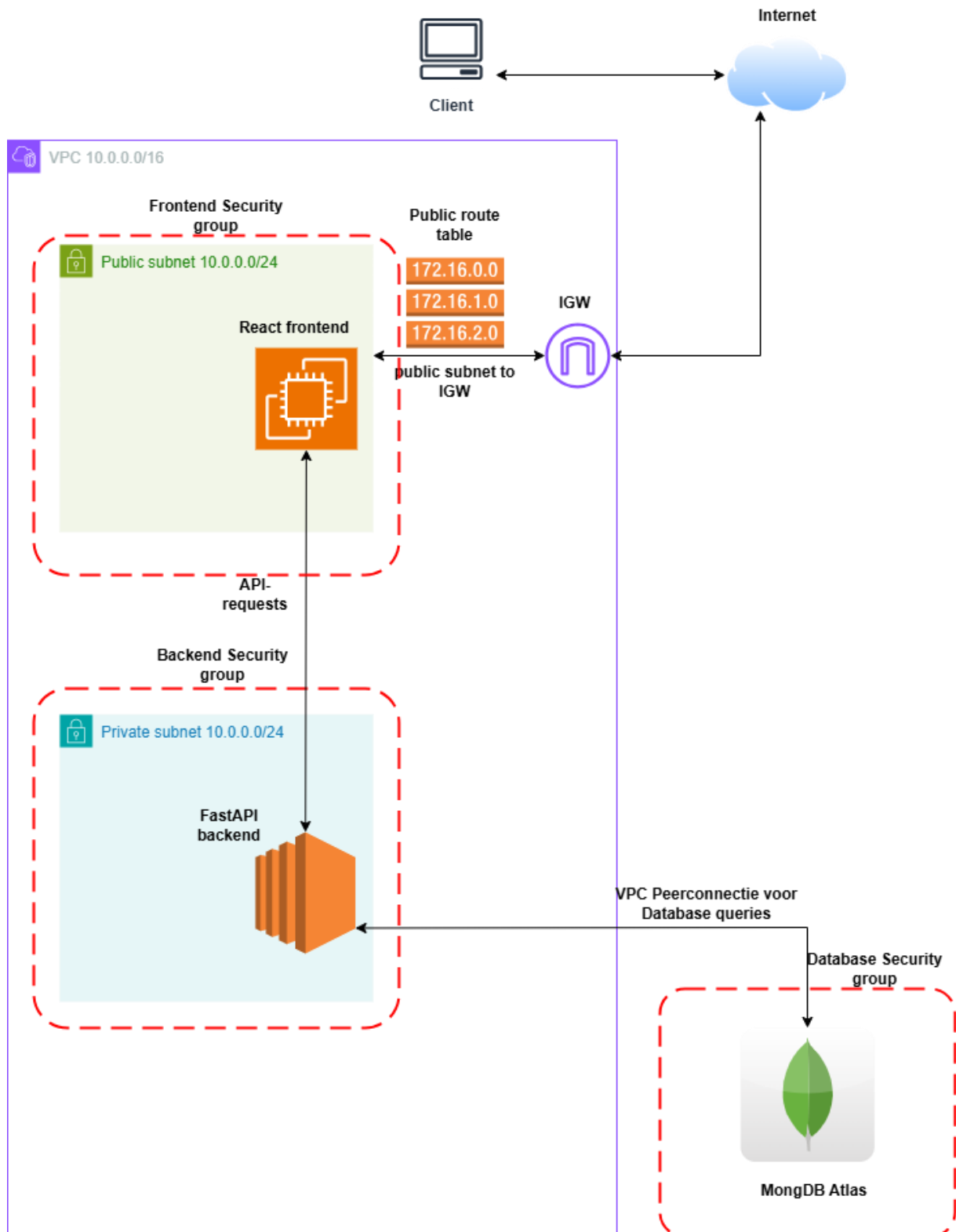
De MongoDB Atlas-database wordt extern beheerd en is verbonden met de EC2-instance via een veilig, versleutelde verbinding. Door het gebruik van VPC-peering kan de EC2-instance communiceren met de database zonder dat het verkeer via het openbare internet gaat, wat de veiligheid verbetert.

5.3.5. Schaalbaarheid en Beschikbaarheid

De infrastructuur is opgezet met de mogelijkheid om te schalen:

- de EC2-instanties kunnen geconfigureerd worden voor automatische schaalbaarheid via Auto Scaling Groups als dat nodig is.

5.3.6. Diagram



5.4. Kostenanalyse

5.4.1. VPC en Subnets

Het aanmaken van een VPC en subnets is gratis en brengt geen kosten met zich mee.

5.4.2. EC2 Instanties

Voor beide instanties kiezen we voor een t3.micro, deze hebben een on-demand kost van \$0.0104 per uur. Ervan uitgaand dat een maand 720 uur telt (24uur x 30 dagen) kost elk van deze instances \$7.49 per maand. Samen komt dit uit op \$14.98 per maand.

5.4.3. MongoDB Atlas

Voor onze database gebruiken we de free tier waarmee we toegang hebben tot ongeveer 0.5Gb. voor een proof of concept is dit zeker genoeg en dit zorgt er ook voor dat we hier dus geen extra kosten hebben.

5.4.4. Data Transfer

De eerst Gb aan data transfer is gratis, daarna kost dit \$0.60 per Gb. Voor onze proof of concept is dit nogmaals een kost waar we dus geen rekening mee moeten houden omdat we verwachten dat we onder 1Gb aan data transfers kunnen blijven.

5.4.5. Productie omgeving

De voorgaande berekening is op basis van een proof of concept. Als we beginnen nadenken over een omgeving die klaar is om te gebruiken in productie ziet het kostenplaatje er al snel anders uit. Dit valt echter buiten scope voor dit project maar we willen hier toch graag een indicatie geven voor welke zaken we dan extra nodig hebben en hoe het uiteindelijke kostenplaatje er dan kan uitzien. Onze ec2 instanties moeten automatisch schaalbaar zijn en meer bezoekers op piekmomenten kunnen verwerken. We hebben nood aan een load balancer en moeten gebruik maken van de betaalde versie van MongoDB Atlas om de grotere hoeveelheid data op te kunnen slaan. Verder hebben we ook een AWS WAF(Web Application Firewall) nodig, AWS CloudWatch en AWS guardDuty. Dit zijn AWS services die de veiligheid en logging/monitoring van de applicatie verhogen. Deze extra opties geven ons een nieuw kostenplaatje van \$129.88.

5.4.6. Conclusie

Voor een proof of concept hebben we hier een zeer goedkope opstelling kunnen samenstellen die in totaal slechts \$14.98 per maand kost. En dat op voorwaarde dat de instanties 24/7 draaien wat zelfs niet het geval zal zijn. Dit zorgt ervoor dat we met onze \$50 learner credits op AWS ruimschoots toekomen.

Het is belangrijk te beseffen dat deze kosten slechts schattingen zijn, en de uiteindelijke kosten afhankelijk zijn van daadwerkelijk gebruik en variabelen in de cloudomgeving. Cloudomgevingen bieden flexibiliteit, maar vereisen ook een voortdurende monitoring van gebruik en kosten om binnen budget te blijven.

5.5. Gitlab CI/CD

Bij het ontwikkelen van dit project vinden we eenvoud en snelheid belangrijk, alsook een gestructureerde workflow om een soepele samenwerking te garanderen en fouten te minimaliseren. Daarom stellen we een werkwijze voor waarbij we GitLab CI/CD gebruiken om automatisering te combineren met een duidelijke branch-structuur.

Met deze aanpak werken de ontwikkelaars zelfstandig, testen we wijzigingen grondig in een staging-omgeving en zorgen we voor een gecontroleerde overgang naar productie. Dit onderdeel beschrijft de voorgestelde workflow, van branch-management tot automatische deployments in AWS.

5.5.1. Workflow Overzicht

We werken met een aantal logische stappen die het ontwikkelproces organiseren. Elke stap heeft een duidelijke plek in de samenwerking tussen het team en het deployen van de applicatie:

- **Persoonlijk werk in aparte branches**
- Elke ontwikkelaar werkt in zijn of haar eigen tak, bijvoorbeeld dev-bw of dev-gb. Dit voorkomt dat ze elkaars werk per ongeluk beïnvloeden.
- **Samenvoegen en testen in de staging-omgeving**
- Als een ontwikkelaar klaar is, worden de wijzigingen naar de test branch gestuurd. Dit is een centrale plek waar we alles samenvoegen en automatisch testen.
- **Gecontroleerde productie-deployment**
- Pas als alles goed getest is, worden de wijzigingen handmatig doorgestuurd naar de main branch. Dit triggert een automatische deployment naar de productieomgeving.

Deze aanpak houdt het simpel, maar geeft toch voldoende controle over wat er naar productie gaat.

5.5.2. Branching Strategie

We gebruiken drie soorten branches om het werk op te delen:

- **Main (ook wel production):**
- Dit is de branch die altijd de stabiele versie van de applicatie bevat. Alles wat hier staat, wordt direct naar productie gedeployed.
- **Test (ook wel staging):**
- Hier komen alle wijzigingen samen voordat ze naar productie gaan. Het is de ideale plek om te testen en te controleren of alles goed werkt in een omgeving die lijkt op productie.
- **Persoonlijke branches (bijvoorbeeld dev-bw):**
- Elke ontwikkelaar werkt in een eigen tak. Hier wordt nieuwe functionaliteit ontwikkeld of worden bugs opgelost.

Door deze indeling kunnen we het werk eenvoudig structureren en mogelijke problemen vroegtijdig opsporen.

5.5.3. CI/CD Pipelines

Automatisering bespaart tijd en minimaliseert fouten. Met GitLab CI/CD zorgen we ervoor dat een groot deel van het proces automatisch verloopt. Hier is hoe dat eruitziet:

Persoonlijke branches (dev-<initialen>):

- Elke keer dat een ontwikkelaar code toevoegt aan zijn of haar branch, wordt deze automatisch gecontroleerd. Denk aan:
- Codekwaliteit (linting).
- Basisfunctionaliteit (unittests).
- Als er problemen zijn, ziet de ontwikkelaar direct wat er misgaat.

Testbranch (test):

- Wanneer een persoonlijke branch wordt samengevoegd met test, gebeurt er meer:
- Integratietests controleren of de nieuwe code goed samenwerkt met bestaande functies.
- De applicatie wordt automatisch gedeployed naar een staging-omgeving in AWS.
- In de staging-omgeving kan het team gezamenlijk testen.

Mainbranch (main):

- Wanneer alles in test goed werkt, kunnen we handmatig beslissen om de wijzigingen naar main te sturen.
- Een merge naar main start automatisch een productie-deployment naar AWS.

Dit proces combineert snelheid met betrouwbaarheid.

5.5.4. AWS Deployment Strategie

Voor de deployment gebruiken we AWS. We maken gebruik van twee omgevingen:

- **Staging-omgeving:**
- Hier testen we nieuwe wijzigingen voordat ze live gaan. Dit is een veilige plek om fouten op te sporen en op te lossen.
- **Productie-omgeving:**
- Dit is de live-omgeving waar eindgebruikers toegang hebben tot de applicatie.

5.5.4.1. Hoe werkt de deployment?

- **Automatisch:**
- Zodra wijzigingen naar test worden gestuurd, deployen we naar staging.
- **Manueel:**
- Bij een merge naar main vragen we eerst een handmatige goedkeuring. Pas daarna wordt de applicatie gedeployed naar productie.

Met deze scheiding tussen staging en productie zorgen we ervoor dat fouten worden opgevangen voordat gebruikers ze zien.

5.5.5. Stappenplan Workflow

Hier is hoe een typische werkcyclus eruitziet:

- **Ontwikkeling in persoonlijke branches:**
- Elke ontwikkelaar werkt aan zijn of haar eigen branch (dev-<initialen>).
- Code wordt automatisch getest en gecheckt bij elke commit.
- **Testen in staging:**
- Zodra een feature klaar is, maakt de ontwikkelaar een merge request naar test.
- De wijzigingen worden samengevoegd en automatisch gedeployed naar staging.
- Het team test gezamenlijk in een realistische omgeving.
- **Productie-deployment:**

- Als de tests in staging succesvol zijn, wordt de test branch manueel gemerged naar main.
- De productieomgeving wordt automatisch bijgewerkt.

Door deze aanpak blijft het proces helder en voorspelbaar.

5.5.6. Monitoring en Onderhoud

Om de workflow soepel te houden, zijn er een paar praktische onderhoudsacties:

- **Pipeline-meldingen:**
- We stellen GitLab in om meldingen te sturen als een pipeline faalt. Zo kunnen we problemen snel oplossen.
- **Opschonen van branches:**
- Persoonlijke branches worden verwijderd zodra een feature succesvol is geïntegreerd in test.
- **Staging cleanup:**
- Oude resources in de staging-omgeving worden automatisch opgeruimd om kosten en complexiteit te verminderen.

5.5.7. Conclusie en Voordelen

Deze workflow is ontworpen om de balans te vinden tussen eenvoud en controle, met duidelijke voordelen:

- **Snelheid:** Ontwikkelaars kunnen zelfstandig werken zonder elkaar in de weg te zitten.
- **Kwaliteit:** Door automatisering worden fouten vroeg opgespoord en getest.
- **Betrouwbaarheid:** Staging-tests en handmatige goedkeuring voorkomen fouten in productie.

Met deze aanpak kan ons team effectief samenwerken en tegelijkertijd het Proof of Concept succesvol realiseren.

6.Planning

6.1. Week 1 – 13/01/2025

In de eerste week van de realisatiefase van het project gaat Cropwatchers het project opzetten op hun laptops. Vervolgens worden de homepagina's en de dashboards aangemaakt. Er wordt ook gezorgd dat een gebruiker zich kan registreren en zijn profiel kan bekijken. Tegelijkertijd wordt er begonnen aan de dataverzameling voor het project en wordt de basis gelegd voor de Chatbot. Hiernaast wordt ook de basis Cloud infrastructuur opgezet.

6.2. Week 2 – 20/01/2025

In de tweede week van het project, wordt ervoor gezorgd dat een landbouwer een veld kan registreren en opvragen. Er wordt eveneens gezorgd dat een landbouwer kan chatten met een chatbot. Een consultant moet ook toegang krijgen tot een veld en hij/zij moet de gegevens van landbouwer kunnen opvragen en advies kunnen geven. Verder wordt er begonnen aan de use case 'Schadeclaim aanvragen'. Daarnaast worden er ook visualisaties gemaakt die verduidelijking geven aan de persona's. Hiernaast worden ook de verbindingen binnen de infrastructuur getest en wordt de beveiliging verbeterd.

6.3. Week 3 – 10/02/2025

In de week van 10 februari, wordt er gezorgd dat een schadeclaim kan aangevraagd worden en dat deze ook aangepast kan worden. Er wordt ook gezorgd dat de status van een schadeclaim kan aangepast worden. Vervolgens wordt er voor gezorgd dat een overheidsmedewerker de gegevens over landbouwschade kan opvragen per gemeente. Verder wordt de infrastructuur gefinetuned, gekeken waar eventuele problemen kunnen opspelen en er wordt aan kostenbeheer gedaan om te kijken of er nog extra besparingen nodig zijn.

6.4. Week 4 – 17/02/2025

In de week van 17 februari, gaat Cropwatchers hun project finaliseren. Nadien wordt het project gepresenteerd voor een jury.

7. Conclusie

Dit project bestaat dus uit 4 belangrijke onderdelen, namelijk persona's, eisenanalyse, data verzameling en deployment.

Eerst werd er geanalyseerd welke persona's er aan bod moesten komen. Dit zijn eigenlijk de gebruikers die het systeem gaan gebruiken. Hiervan werden de IT-vaardigheden en befoefen en de uitdagingen bestudeerd.

Verder werd er in het deel 'eisenanalyse' aan de hand van de persona's een Use Case Diagram en een Entity Relationship Diagram opgesteld. Hierdoor werd duidelijk wat alle eisen waren van de gebruikers. Door middel van prototypes werd er geanalyseerd hoe de eisen van de gebruikers zouden kunnen voldaan worden.

Nadien gingen we over naar het data verzameling hoofdstuk. Hierin werd uitgelegd hoe de kaartintegratie van Geopunt ging gebruikt worden (React Leaflet) elke services van OpenEO er gebruikt gaan worden. Vervolgens werd er in dit deel een analyse gedaan voor de LLM (LLaMA met Groq API), de scraper (Custom Search JSON API) en de visualisatietool (Plotly).

Ten slotte gingen we over naar het deployment hoofdstuk. Hierin bespreken we welke technologieën gebruikt worden om onze applicatie te laten werken zoals beschreven in de FARM-stack. Vervolgens vergelijken we de verschillende hostingopties en werd er een infrastructuur ontworpen. In het onderdeel gitlab ci/cd werd er nagedacht over automatisatie van de code en een branching strategie waardoor we een vlotte en efficiënte samenwerking kunnen behouden.

8. Bibliografie

- Amazon Web Services, Inc. (2024). *docs*. Opgehaald van [aws.amazon.com](https://docs.aws.amazon.com/): <https://docs.aws.amazon.com/>
- Chart.js. (sd). *Chart.js documentation*. Opgehaald van Chart.js: <https://www.chartjs.org/>
- DB-Engines. (sd). *DB-Engines*. Opgehaald van DB-Engines: <https://db-engines.com/en/>
- Face, H. (sd). *BERT*. Opgehaald van Hugging Face: https://huggingface.co/docs/transformers/model_doc/bert
- Face, H. (sd). *T5*. Opgehaald van Hugging Face: https://huggingface.co/docs/transformers/model_doc/t5
- geeksforgeeks. (2024, 09 23). *angular-vs-reactjs*. Opgehaald van geeksforgeeks: <https://www.geeksforgeeks.org/angular-vs-reactjs/>
- Geoapify. (2019, 04 12). *leaflet-vs-openlayers*. Opgehaald van geoapify: <https://www.geoapify.com/leaflet-vs-openlayers/>
- Google. (2024, December 12). *Overview of language models*. Opgehaald van Google Cloud: <https://cloud.google.com/vertex-ai/generative-ai/docs/language-model-overview#palm-api>
- Google. (sd). *bert*. Opgehaald van Github: <https://github.com/google-research/bert>
- Google. (sd). *Custom Search JSON API*. Opgehaald van Programmable Search Engine: <https://developers.google.com/custom-search/v1/overview>
- Google. (sd). *text-to-text-transformers*. Opgehaald van Github: <https://github.com/google-research/text-to-text-transfer-transformer>
- Hensels, A., & Verbruggen, B. (2024). Canvas Cursus Cloud Engineering.
- Lee, J. (2024, 06 23). *choosing-the-right-python-web-framework-django-vs-fastapi-vs-flask*. Opgehaald van devtoys: <https://devtoys.io/2024/06/23/choosing-the-right-python-web-framework-django-vs-fastapi-vs-flask/>
- Llama, M. (sd). *The Llama Family*. Opgehaald van Hugging Face: <https://huggingface.co/meta-llama>
- Meta. (2024). *Graph API*. Opgehaald van Meta: <https://developers.facebook.com/docs/graph-api/>
- Meta. (sd). *Get started with Llama*. Opgehaald van Meta: <https://www.llama.com/docs/overview>
- Meta. (sd). *Instagram API with Instagram Login*. Opgehaald van Meta: <https://developers.facebook.com/docs/instagram-platform/instagram-api-with-instagram-login>
- Meta. (sd). *Llama*. Opgehaald van Meta: <https://www.llama.com>
- MongoDB. (sd). *What is retrieval-augmented generation (RAG)?* Opgehaald van MongoDB: <https://www.mongodb.com/resources/basics/artificial-intelligence/retrieval-augmented-generation>
- MongoDB, Inc. (2024). *docs*. Opgehaald van MongoDB: <https://www.mongodb.com/docs/>
- OpenAI. (sd). *OpenAI developer platform*. Opgehaald van OpenAI: <https://platform.openai.com/docs/overview>
- Plotly. (sd). *Dash Python User Guide*. Opgehaald van Plotly: <https://dash.plotly.com/>
- react-leaflet. (sd). *react-leaflet.js.org*. Opgehaald van react-leaflet.js.org: <https://react-leaflet.js.org/docs/start-introduction/>
- SerpApi. (sd). *Google News API*. Opgehaald van SerpApi: <https://serpapi.com/google-news-api>
- Smith, M. (2024, 04 23). *8-fastapi-mongodb-best-practices*. Opgehaald van mongodb: <https://www.mongodb.com/developer/products/mongodb/8-fastapi-mongodb-best-practices/>
- TikTok. (sd). *Research Tools*. Opgehaald van TikTok for developers: <https://developers.tiktok.com/products/research-api/>
- X. (sd). *X API*. Opgehaald van X Developer Platform: <https://developer.x.com/en/docs/x-api>