Analysedocument  
Monitor van de stad

Ontdekstation013 & Bibliotheken Midden-Brabant

Fontys Hogeschool | Tilburg

|  |  |
| --- | --- |
| **Auteurs**: | Cyrion van dongen, Janno Vos, Janiek van Pelt, Stan Akkermans, Fabiënne Leidekker, Daan Brouwer |
| **Locatie**: | Tilburg |
| **Versie:** | 1.1 |
| **Datum**: | 14 april 2022 |
|  |  |

|  |  |  |
| --- | --- | --- |
| Versie | Wijziging | Datum |
| 1.1 | Extra requirements toegevoegd | 14-04-2022 |

# Versiebeheer

Inhoud

[1 Versiebeheer 2](#_Toc101255635)

[3 Inleiding 4](#_Toc101255636)

[4 Context 4](#_Toc101255637)

[5 Requirements 5](#_Toc101255638)

[5.1 Functionele requirements 5](#_Toc101255639)

[5.2 Acties 6](#_Toc101255640)

[5.3 Non functional requirements 6](#_Toc101255641)

[5.4 MoSCoW-Methode 7](#_Toc101255642)

[5.5 Definition of Done 8](#_Toc101255643)

[5.6 Documentatie kwaliteit 8](#_Toc101255644)

[5.7 Ease of use 8](#_Toc101255645)

[5.8 Schaalbaarheid 8](#_Toc101255646)

[6 Use case 9](#_Toc101255647)

[7 Use case diagram 14](#_Toc101255648)

[8 Double Diamond 15](#_Toc101255649)

[8.1 Wat fase 15](#_Toc101255650)

[8.1.1 UC01: Account aanmaken 15](#_Toc101255651)

[8.1.2 UC02: Inloggen 16](#_Toc101255652)

[8.1.3 UC03: Uitloggen 16](#_Toc101255653)

[8.1.4 UC04: Aanmaken persoonlijk meetpunt 16](#_Toc101255654)

[8.1.5 UC05: Verwijder persoonlijk meetpunt 17](#_Toc101255655)

[8.1.6 UC06: Wijzigen persoonlijk meetpunt 17](#_Toc101255656)

[8.1.7 UC07: Huidige metingen pand 18](#_Toc101255657)

[8.1.8 UC08: Huidige metingen gebied 18](#_Toc101255658)

[8.1.9 UC09: Huidige metingen meetpunt 19](#_Toc101255659)

[8.1.10 UC10: Gegevensdistributie bekijken 19](#_Toc101255660)

[8.1.11 UC11: Meetpunten van een gebied bekijken 20](#_Toc101255661)

[8.2 BPMN Diagrammen 21](#_Toc101255662)

[9 Domeinmodel 25](#_Toc101255663)

# Inleiding

In dit document wordt al beschreven wat voor functionaliteiten in de applicatie komen, krijgen deze functionaliteiten een prioriteit en worden er ook beperkingen gegeven aan deze prioriteiten zodat ze binnen de scope blijven en het een duidelijke requirement blijft. Met behulp van Use Cases krijgen deze requirements meer vorm door ze in een context te plaatsen. Ook wordt er met een Use Case diagram getoond wat mogelijk is voor welke actoren binnen de applicatie. Hiernaast worden er ook Non Functionals opgesteld om de kwaliteiten van de applicatie te beschrijven en wordt er al een kleine schets gemaakt door middel van een domeinmodel.

# Context

Daarvoor willen we inzicht creëren in de luchtkwaliteit in de openbare ruimte en in huizen en gebouwen. De meetgegevens willen we op een aansprekende wijze presenteren. De opdrachtgever Ontdekstation 013 werkt aan projecten samen met jongeren om hen te interesseren in de techniek. Vanuit de betrokkenheid bij Klimaatadaptatie, Energietransitie en schone lucht willen wij samen met de Bibliotheek Midden-Brabant een project rondom het meten van luchtkwaliteit in de regio Tilburg vormgeven.

In Tilburg staan op het moment verschillende sensoren die verschillende waardes kunnen meten. Aan ons de taak om een gebruikersvriendelijke website te maken met een goed functionerende back-end. Het is voor dit project erg belangrijk dat dit zo modulair mogelijk wordt opgezet zodat er nog makkelijk dingen aangepast kunnen worden. Bij voorkeur moet de applicatie ook werken in combinatie met het TTN LoRa netwerk in de regio tilburg. Hierbij zal ook gebruik gemaakt moeten worden van open source standaarden.

# Requirements

## Functionele requirements

FR-01 Als gebruiker wil ik de huidige metingen in kunnen zien van een gebied.  
 K-01.1 Verschillende soorten data zijn op te halen

FR-02 Als gebruiker wil ik per meetpunt de gemeten data in kunnen zien.

K-02.1 Ieder meetpunt is individueel

FR-03 Als gebruiker wil ik kunnen inloggen.

B-03.1 De gebruiker moet gegevens invullen

FR-04 Als gebruiker wil ik kunnen uitloggen.

B-04.1 De gebruiker moet ingelogd zijn

FR-05 Als ingelogde gebruiker wil ik data van mijn eigen pand in kunnen zien.

B-06.1 De gebruiker moet ingelogd zijn

FR-06 Als ingelogde gebruiker wil ik een meetpunt aan kunnen maken.

B-06.1 De gebruiker moet de gewenste gegevens invullen

B-06.2 De gebruiker moet ingelogd zijn

K-06.3 Opties om in te vullen zijn de naam, registratiecode, of het publiekelijk zichtbaar is, de hoogte, directie en welke sensoren erop staan

FR-07 Als ingelogde gebruiker wil ik een zelfgemaakt meetpunt kunnen verwijderen.

B-07.1 Er moeten meetpunten zijn

B-07.2 De gebruiker moet ingelogd zijn

FR-08 Als ingelogde gebruiker wil ik een zelfgemaakt meetpunt kunnen wijzigen.

B-08.1 Er moet minimaal één meetpunt zijn

B-08.2 De gebruiker moet ingelogd zijn  
 K-08.3 Alle gegevens kunnen worden aangepast, behalve de registratiecode

FR-09 Als gebruiker wil ik een nieuw account aan kunnen maken.

B-09.2 De gebruiker moet een wachtwoord en gebruikersnaam aangeven

FR-10 Als gebruiker wil ik gegevensdistributie kunnen zien via een heatmap.

FR-11 Als gebruiker wil ik de meetpunten binnen een gebied kunnen bekijken.

K-11.1 Alleen stations die niet als privé staan kunnen worden bekeken

## Acties

Acties zijn requirements die niet zozeer door een persoon uitgevoerd moeten worden, maar wel nodig zijn om het systeem werkend te hebben.

FA-01 Als gebruiker wil ik meldingen krijgen als er iets mis is met mijn station.

B-12.1 De gebruiker moet een account hebben

B-12.2 De gebruiker moet eigen meetpunten hebben

FA-02 Als beheerder wil ik dat de data van stations ontvangen wordt via het LoRa netwerk.

FA-03 Als gebruiker wil ik de windrichting op de kaart kunnen zien.

## Non functional requirements

|  |  |  |
| --- | --- | --- |
| ID | Categorie ISO 25010 | Omschrijving |
| NFR-01 | Compatibility | De back-end van de applicatie wordt in de taal Java geschreven met behulp van SpringBoot. |
| NFR-02 | Compatibility | Er wordt gebruik gemaakt van een MYSQL Database. |
| NFR-03 | Performance efficiency | De applicatie wordt gehost op … (nog invullen, bv fhict server). |
| NFR-04 | Compatibility | De front-end van de applicatie wordt gemaakt met behulp van ReactJS. |

## MoSCoW-Methode

Om een duidelijk beeld te geven van de prioritering binnen dit project wordt er binnen dit hoofdstuk gebruik gemaakt van de MoSCoW-methode. Deze methode geeft een inzicht in welke taken de groep hoog prioriteert, en welke laag.

|  |  |
| --- | --- |
| Functionele requirement(s) | Prioritering |
| FR01 - Als gebruiker wil ik de huidige metingen in kunnen zien van een gebied. | Must have |
| FR02 - Als gebruiker wil ik per meetpunt de gemeten data in kunnen zien. | Must have |
| FR03 - Als gebruiker wil ik kunnen inloggen. | Must have |
| FR04 - Als gebruiker wil ik kunnen uitloggen. | Must have |
| FR05 - Als ingelogde gebruiker wil ik data van mijn eigen pand in kunnen zien. | Should have |
| FR06 - Als ingelogde gebruiker wil ik een meetpunt aan kunnen maken. | Must have |
| FR07 - Als ingelogde gebruiker wil ik een zelfgemaakt meetpunt kunnen verwijderen. | Should have |
| FR08 - Als ingelogde gebruiker wil ik een zelfgemaakt meetpunt kunnen wijzigen. | Should have |
| FR09 - Als gebruiker wil ik een nieuw account aan kunnen maken. | Must have |
| FR10 - Als gebruiker wil ik gegevensdistributie kunnen zien via een heatmap. | Should have |
| FR11 - Als gebruiker wil ik de meetpunten binnen een gebied kunnen bekijken. | Could have |
| FA01 - Als ingelogde gebruiker wil ik meldingen krijgen als er iets mis is met mijn station. | Should have |
| FA02 - Als beheerder wil ik dat de data van stations ontvangen wordt via het LoRa netwerk. | Must have |
| FA03 - Als gebruiker wil ik de windrichting op de kaart kunnen zien. | Should have |

|  |  |
| --- | --- |
| Non-functionele requirement(s) | Prioritering |
| NF01 - De back-end van de applicatie wordt in de taal Java geschreven met behulp van SpringBoot. | Must have |
| NF02 - Er wordt gebruik gemaakt van een MYSQL Database. | Must have |
| NF03 - De applicatie wordt gehost op … (nog invullen, bv fhict server). | Must have |
| NF04 - De front-end van de applicatie wordt gemaakt met behulp van ReactJS. | Must have |
| NF05 - De app moet via telefoon beschikbaar zijn. | Could have |

## Definition of Done

* De issue moet gereviewed worden met een ander groepslid
* Wanneer deze gereviewed is moet de front/back-end getest worden
* De issue moet gemerged worden met de dev branch
* Check handmatig of de code goed werkt in combinatie met je ci/cd
* Check met een docent of de story gemerged kan worden vanuit dev naar originele dev

## Documentatie kwaliteit

Omdat dit project voor een langere tijd in ontwikkeling gaat zijn (en door verschillende proftaak teams) is het van belang dat de overdraagbaarheid goed is. Eén van de opties die helpen om de overdraagbaarheid te verbeteren is om documentatie te schrijven die duidelijk, straight-to-the-point en up-to-date is. Ook is het belangrijk dat keuzes en sourcecode gedocumenteerd is.

Alle documentatie van dit project wordt opgeslagen in de publieke OneDrive. Alle documentatie heeft dezelfde huisstijl en bevat altijd de context van de opdracht.

De sourcecode is gedocumenteerd door overal dezelfde code-stijl te gebruiken, goede functienamen en lastige stukken code met commentaar te beschrijven. Om deze code-stijl recht te houden wordt er gebruik gemaakt van SonarQube. Ook kan code alleen gemerged worden wanneer er minimaal één persoon de pull-request beoordeeld heeft.

## Ease of use

Bij het uitwerken van het project wordt rekening gehouden met de doelgroep. Omdat deze doelgroep niet altijd veel technische kennis heeft, moet hier rekening mee gehouden worden in het design van de applicatie en zijn functionaliteiten. Zo wordt er op gelet dat er niet al te vaak klikken nodig zijn om een bepaalde functionaliteit aan te roepen. De acties moeten zo duidelijk mogelijk zijn met zo min mogelijk woorden, en er zal ook geen informatie-overflow mogen zijn. Extra informatie voor zij die het willen, zal bereikbaar zijn binnen verschillende opties. Daarnaast wordt erop gelet dat pagina's niet te onoverzichtelijk worden voor gebruikers. In een later stadium zal dit dan ook getest worden door mensen met minder technische kennis.

## Schaalbaarheid

Een leerdoel in dit project is om software de bouwen die goed schaalbaar en uitbreidbaar is.

# Use case

|  |  |
| --- | --- |
| **Naam** | UC01: Account aanmaken |
| **Requirements** | FR-09 |
| **Samenvatting** | De actor wil een account aanmaken |
| **Actor** | Gast |
| **Aanname** | De actor is ingelogd |
| **Beschrijving** | 1. De actor wil een account aanmaken. 2. Het systeem vraagt naar de gegevens om een account te maken. 3. De actor voert de gevraagde gegevens in 4. Het systeem controleert de gegevens [1] 5. Het systeem maakt een account aan. |
| **Uitzondering** | 1. De ingevoerde gegevens zijn al in gebruik, ga terug naar stap 3 |
| **Resultaat** | Een nieuw account is aangemaakt en de actor wordt doorverwezen naar de inlogpagina. |

|  |  |
| --- | --- |
| **Naam** | UC02: Inloggen |
| **Requirements** | FR-03 |
| **Samenvatting** | De actor wil kunnen inloggen |
| **Actor** | Gebruiker, Admin |
| **Aanname** | Er zijn bestaande admin en gebruiker accounts (…) |
| **Beschrijving** | 1. De actor wil inloggen. 2. Het systeem vraagt naar de gegevens om in te loggen. 3. De actor vult zijn/haar inloggegevens in. 4. Het systeem controleert de ingevulde gegevens.[1] 5. De actor is ingelogd in zijn/haar account. |
| **Uitzondering** | 1. De actor heeft een onjuiste inloggegevens ingevoerd, terug naar stap 1 |
| **Resultaat** | Het systeem logt de actor normaal in, als admin codes zijn ingevoerd verwijst het door naar de admin pagina. |

|  |  |
| --- | --- |
| **Naam** | UC03: Uitloggen |
| **Requirements** | FR-04 |
| **Samenvatting** | De actor wil kunnen uitloggen |
| **Actor** | Gebruiker, Admin |
| **Aanname** | De actor is ingelogd |
| **Beschrijving** | 1. De actor wil uitloggen. 2. Het systeem logt de gebruiker uit. 3. De actor is uitgelogd. |
| **Uitzondering** |  |
| **Resultaat** | De actor is uitgelogd en wordt doorverwezen naar de hoofdpagina. |

|  |  |
| --- | --- |
| **Naam** | UC04: Aanmaken persoonlijk meetpunt |
| **Requirements** | FR-06 |
| **Samenvatting** | De actor wil een nieuw meetpunt aanmaken |
| **Actor** | Gebruiker |
| **Aanname** | B-06.1 De actor is ingelogd |
| **Beschrijving** | 1. De actor geeft aan om een meetpunt aan te willen maken 2. Het systeem toont de juiste pagina 3. De actor vult de gevraagde data in 4. De actor selecteert “Bevestig” [1] 5. Het systeem toont een bevestiging |
| **Uitzondering** | 1. De ingevoerde data is onjuist, ga terug naar stap 2 |
| **Resultaat** | Het meetpunt is aangemaakt en de actor wordt doorverwezen naar de (…) pagina. |

|  |  |
| --- | --- |
| **Naam** | UC05: Verwijderen persoonlijk meetpunt |
| **Requirements** | FR-07 |
| **Samenvatting** | De actor wil een meetpunt kunnen verwijderen |
| **Actor** | Gebruiker |
| **Aanname** | B-07.2 De actor is ingelogd |
| **Beschrijving** | 1. De actor gaat naar het meetpunt dat hij/zij wil verwijderen. 2. Het systeem vraagt of dit meetpunt definitief wil verwijderen. 3. De actor wil het meetpunt definitief verwijderen. 4. Het systeem verwijdert het meetpunt. |
| **Uitzondering** |  |
| **Resultaat** | Het meetpunt is en de actor wordt doorverzen naar de (…) pagina. |

|  |  |
| --- | --- |
| **Naam** | UC06: Wijzigen persoonlijk meetpunt |
| **Requirements** | FR-08 |
| **Samenvatting** | De actor wil een meetpunt aanpassen |
| **Actor** | Gebruiker |
| **Aanname** | B-08.2 De actor is ingelogd |
| **Beschrijving** | 1. De actor gaat naar het meetpunt waar hij/zij een aanpassing in wil maken. 2. Het systeem toont de gegevens van het gekozen meetpunt. 3. De actor vult de gegevens in die hij/zij wil aanpassen. [1] 4. Het systeem past het meetpunt aan. |
| **Uitzondering** | 1. De ingevoerde data is onjuist, ga terug naar stap 2 |
| **Resultaat** | Het meetpunt is aangepast en de actor wordt doorverwezen naar de (…) pagina. |

|  |  |
| --- | --- |
| **Naam** | UC07: Data van eigen pand inzien |
| **Requirements** | FR-05 |
| **Samenvatting** | De actor wil van zijn eigen pand de huidige metingen inzien |
| **Actor** | Gebruiker |
| **Aanname** | B-05.1 De actor is ingelogd |
| **Beschrijving** | 1. De actor geeft aan de metingen van zijn eigen pand te willen zien. 2. Het systeem toont de gegevens van het pand |
| **Uitzondering** |  |
| **Resultaat** | De huidigen metingen van het pand van de actor zijn zichtbaar. |

|  |  |
| --- | --- |
| **Naam** | UC08: Huidige metingen gebied |
| **Requirements** | FR-01 |
| **Samenvatting** | De actor wil van een gebied op de kaart de huidige metingen inzien |
| **Actor** | Gebruiker |
| **Aanname** |  |
| **Beschrijving** | 1. De actor geeft aan de metingen van een gebied te willen zien. 2. Het systeem toont de huidigen metingen die horen bij het gebied dat gekozen is door de actor. |
| **Uitzondering** |  |
| **Resultaat** | De huidigen metingen van het door de actor gekozen gebied zijn zichtbaar. |

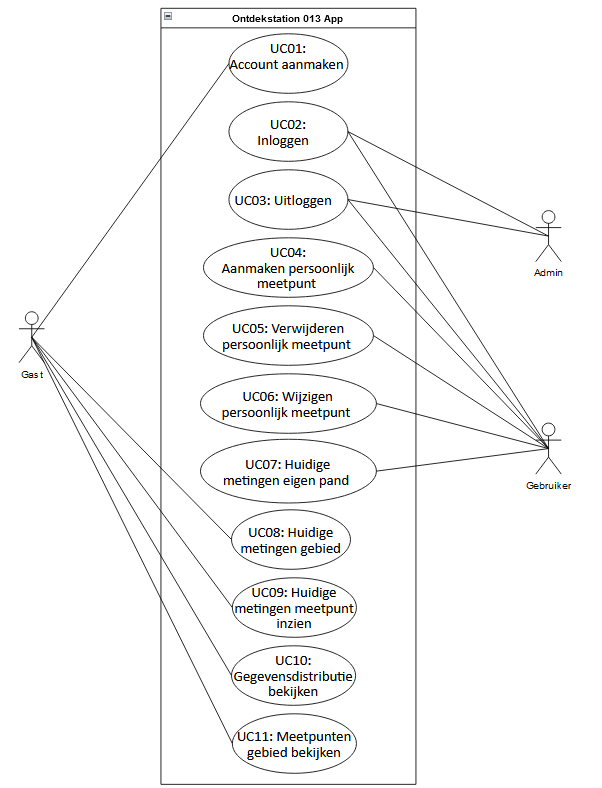
|  |  |
| --- | --- |
| **Naam** | UC09: Huidige metingen meetpunt inzien |
| **Requirements** | FR-02 |
| **Samenvatting** | De actor wil van een meetpunt de huidige metingen inzien |
| **Actor** | Gebruiker |
| **Aanname** |  |
| **Beschrijving** | 1. De actor geeft aan de metingen van een meetpunt te willen zien. 2. Het systeem toont de huidigen metingen die horen bij het meetpunt dat gekozen is door de actor. |
| **Uitzondering** |  |
| **Resultaat** | De huidige metingen van het door de actor gekozen meetpunt zijn zichtbaar. |

|  |  |
| --- | --- |
| **Naam** | UC10: Gegevensdistributie bekijken |
| **Requirements** | FR-10 |
| **Samenvatting** | De actor wil distributie van gegevens bekijken op de kaart d.m.v. een heatmap |
| **Actor** | Gebruiker |
| **Aanname** |  |
| **Beschrijving** | 1. De actor geeft aan van welke data hij de heatmap wil zien 2. Het systeem legt de heatmap over de kaart heen |
| **Uitzondering** |  |
| **Resultaat** | De hoeveelheid van een meetgegeven is zichtbaar per gebied op de kaart. |

|  |  |
| --- | --- |
| **Naam** | UC11: Meetpunten van een gebied bekijken |
| **Requirements** | FR-11 |
| **Samenvatting** | De actor wil de specifieke gegevens bekijken van de meetpunten binnen een gebied |
| **Actor** | Gebruiker |
| **Aanname** |  |
| **Beschrijving** | 1. De actor geeft het gewenste gebied aan 2. De actor geeft aan de meetpunten te willen bekijken 3. Het systeem laat de openbare meetpunten van het gebied zien 4. De actor geeft aan welk meetpunt hij wil bekijken 5. Het systeem laat de gegevens van dit meetpunt zien |
| **Uitzondering** |  |
| **Resultaat** | De actor kan de gegevens van de individuele meetpunten inzien |

# Use case diagram

Om een duidelijk overzicht te geven van welke use cases er bij welke actor horen, is er een use case diagram gemaakt. Hierin staat met lijnen aangegeven welke use case er bij welke actor hoort. Dit is ook terug te vinden in de use cases zelf.



# Double Diamond

## Wat fase

In dit gedeelte van het proces kijken we naar verschillende andere instanties en projecten die onderdelen van onze Use Cases aanpakken. Hierbij proberen we te kijken in hoeverre het toepasbaar is voor ons en wat er eventueel niet zou werken.   
Eventueel kijken we ook naar hoe de Heuristics van Nielsen geïmplementeerd zijn, om zo te zien of we een betere optie kunnen vinden voor onszelf.

### UC01: Account aanmaken

Voor het onderzoek naar hoe andere sites het registreer proces doen hebben we niet alleen gekeken naar sites die gaan over het inzien van data zoals sensor community of meetjestad, maar ook behance gebruikt om inspiratie op te doen over meerdere manieren van registratie en technieken die hiermee te maken hebben.

Bij de voorbeelden die we bekeken hebben zijn een paar punten die veel voorkomen: Als eerste zie je vaak dat er de optie is om in te loggen via Google, Facebook of een ander account. Dit maakt het registreer proces nog makkelijker en flexibeler omdat je een account op meerdere manieren kan aanmaken zoals met een account wat de gast al eerder ergens anders heeft gemaakt.

Als tweede zie je meerdere manieren om een gebruiker duidelijk te maken aan welke waardes voldaan moet worden met een email-adres of wachtwoord, zoals met een zin boven of onder het invulveld en een melding tijdens of na het invullen. De voorkeur ligt hier bij een melding om dit aan te geven om het scherm tijdens registreren niet vol te zetten met extra zinnen of data, deze wordt dan alleen te zien wanneer het nodig is.

Als laatste is er ook een knop of link om naar het inlog scherm te gaan, dit is er om de gebruiker in controle te houden en het proces flexibel en gebruiksvriendelijk te houden. Wanneer een gebruiker dus per ongelijk bij registreren uitkomt in plaats van login of andersom, kan de gebruiker altijd nog makkelijk en snel naar de gewenste pagina.

Bij het aanmaken van een account vragen wij van de gast alleen een naam, email-adres en een wachtwoord. Het wachtwoord moet een minimaal aantal karakters hebben en het email-adres moet een @ hebben en een eind deel (.nl , .com enz.)

Om het voor de gebruiker zo duidelijk mogelijk te houden willen we op de site altijd dezelfde woorden gebruiken voor accountinformatie, bijvoorbeeld wanneer we naam gebruiken wordt het niet op de volgende pagina gebruikersnaam en daarna weer username.

Ze moeten deze waarden hebben om te voorkomen dat er fouten komen als het email-adres gebruikt moet worden door het systeem om bijvoorbeeld e-mails te sturen met informatie over je meetstation. De waardes voor het wachtwoord zijn voor de veiligheid van de gebruiker en zijn/haar data.

### UC02: Inloggen

Voor het onderzoek naar hoe andere sites het login proces doen hebben we dezelfde sources gebruikt als bij het registreren om echt goed te kijken hoe de twee processen samenlopen. Hieronder vallen dus sensor community, meetjestad en behance.

Bij alle sites waar we gekeken hebben is het inlog proces zo klein/simpel mogelijk met alleen een username of email-adres en een wachtwoord, ook wordt hier bij de sites waar gebruikt gemaakt werd van registreren met andere accounts gebruikt gemaakt van inloggen met andere accounts. Dit wordt dan op een minimalistische manier getoond door de logos van de andere services boven of naast het standaard inlogformulier te zetten.

Voor onze site willen we het ook simpel houden als het gaat om inloggen, je vult je email-adres en wachtwoord in om in te loggen. Tijdens het inlog proces laten we ook foutmeldingen zien wanneer de combinatie van email-adres en wachtwoord niet matchen door middel van meldingen na de inlog poging.

### UC03: Uitloggen

Voor het onderzoek naar hoe andere sites het uitlog proces doen hebben we dezelfde sources gebruikt als bij het registreren en inloggen om echt goed te kijken hoe de processen samenlopen. Hieronder vallen dus sensor community, meetjestad en behance.

Eigenlijk kiezen alle sites ervoor om de uitlog/exit knop rechtsboven te zetten. Dit is omdat gebruikers tegenwoordig intuïtief hier naartoe gaan om uit te loggen. De gebruiker gaat hier als eerste kijken omdat die de knop daar verwacht, het is dus voor de gebruiker makkelijk te vinden.

Wij willen het dus ook op deze plek zetten om de gebruiker zo makkelijk mogelijk door onze app te kunnen laten navigeren. Om de applicatie ook flexibel en gebruiksvriendelijk te maken wordt er gevraagd of je zeker wilt uitloggen, dit doen we om te voorkomen dat je per ongeluk uitlogt en het hele inlogproces opnieuw moet doorlopen.

### UC04: Aanmaken persoonlijk meetpunt

Voor het onderzoek naar hoe andere sites een proces doen wat vergelijkbaar is met het registreren van een meetpunt, hebben we niet alleen gekeken naar sites die gaan over het inzien van data zoals sensor community of meetjestad, maar ook behance gebruikt om inspiratie op te doen over meerdere manieren van registratie en technieken die hiermee te maken hebben.

Dit proces is het beste te vergelijken met het registreren van een gebruiker. We willen hier dus op vergelijkbare punten letten om het voor de gebruiker zo gemakkelijk, simpel en foutloos mogelijk te laten gaan.

Niet te veel of te uitgebreide informatie vragen bij het registreren, een gebruiker wil niet een uur bezig zijn met het registreren van een punt of hard moeten zoeken naar de gevraagde info. Wij willen dus alleen de naam die je geeft aan het station, de soort sensoren die erin zitten (hiervoor kun je kiezen uit alle ondersteunde opties), de richting, de hoogte van de sensor en de optie of het privé is of niet. Dit zijn de enige belangrijke onderdelen die de gebruiker zelf moet bepalen.

Er moet net als bij het registreren gelet worden op dat de informatie die ingevuld wordt de gevraagde waarde heeft. De voorkeur ligt hier bij een melding om dit aan te geven om het scherm tijdens registreren niet vol te zetten met extra zinnen of data, zo houden we het registreerveld rustig en overzichtelijk.

We helpen ook de gebruiker om onderdelen te herkennen door steeds dezelfde namen te gebruiken op het overzicht van een meetpunt als bij het registreren. Zoals de hoogte (centimeters) en de richting (Noord, Oost, Zuid, West) zullen voor de gebruiker altijd op dezelfde manier weergegeven worden.

Om ook dit gedeelte van de site zo flexibel mogelijk te houden voor de klant kun je al de ingevulde informatie later weer veranderen of het hele punt verwijderen. Ook tijdens het registreren kun je doormiddel van een cancel knop terug naar de vorige pagina als je je bedenkt.

### UC05: Verwijder persoonlijk meetpunt

De gebruiker wil een persoonlijk meetpunt verwijderen.

Op Sensor Community is het vrij lastig om een sensor (meetpunt) te verwijderen. Informatie over het verwijderen van een sensor is bijna niet te vinden. Gebruikers dienen eerst aan te geven dat het punt inactief is en vervolgens een mail te sturen. Dit is zwaar tegenstrijdig met de punten 6 (Recognition rather than recall) en 10 (Help and documentation) van de heuristics van Nielsen. De gebruiker zou zich niet moeten afvragen hoe een relatief simpele opdracht moet en hier op een forum een vraag over moeten stellen. Voor onze applicatie zullen we duidelijke iconen gebruiken die meteen weggeven wat er mee kan gebeuren en zullen we duidelijke informatie geven op de FAQ-pagina.

### UC06: Wijzigen persoonlijk meetpunt

De gebruiker wil een persoonlijk meetpunt wijzigen.

Op MeetJeStad.nl is hier geen informatie over te vinden en ook op Sensor Community blijkt het wijzigen van een sensor een lastige taak te zijn. Sensor Community heeft bij bestaande meetpunten een lastige en lelijke userinterface. Het is voor gebruikers ook moeilijk om informatie te vinden over hoe je nou precies een nieuwe sensor toevoegt (tegenstrijdig met punt 6 en 10 van de Heuristics van Nielsen). Op de pagina waar je een sensor toe zou kunnen voegen staan veel woorden met bijna dezelfde betekenis. Bij punt 4 (Consistency and standards) van de Heuristics van Nielsen, wordt duidelijk gemaakt dat voor gebruikers een duidelijk onderscheid moet worden gemaakt tussen verschillende functies. Zo zijn woorden als configuratie en maken lastig te onderscheiden voor de meeste mensen. Voor onze applicatie zullen we dan ook duidelijk onderscheid maken over de woordkeuze. Wederom zullen we ook duidelijke iconen gebruiken en informatie op het FAQ plaatsen.

### UC07: Huidige metingen pand

<<Overgeslagen voor beginfase project, eventueel voor latere semesters>>

### UC08: Huidige metingen gebied

De gebruiker wil van een huidig gebied (grid) de metingen inzien.

Zowel de sites MeetJeStad.nl en Sensor Community hebben op de homepagina een kaart waarop alle meetpunten te zien zijn. MeetJeStad.nl heeft verschillende kaarten die eerst geopend dienen te worden voordat je ze kunt gebruiken. Deze brengen je dan naar een nieuwe pagina waar de kaart het gehele scherm beslaat. Dit is onhandig en visueel niet aantrekkelijk. De manier waarop de punten worden aangegeven zegt niet veel over wat er gebeurt. Je ziet alleen een thermostaat op de precieze locatie. Sensor Community maakt beter gebruik van de kaart op haar homepage. Deze kaart is meteen te gebruiken zonder een nieuwe pagina te hoeven openen. De kaart heeft verschillende functies zoals filters en zoom in knoppen, echter is de legenda erg onduidelijk en zal de gemiddelde gebruiker in eerste oogopslag geen idee hebben wat er nu precies gemeten wordt. Het aangeven van de meetpunten is wel beter dan die van MeetJeStad.nl maar is ook erg specifiek. Men zou misbruik kunnen maken van deze precieze locaties. Beide sites geven de data over een grotere locatie onhandig weer. Bijvoorbeeld door als je uitzoomt een thermostaat met data voor de gehele omgeving van een stad te laten zien, dit is niet representatief. Punt 7 (Flexibility and efficiency of use) beschrijft de manieren waarop verschillende gebruikers willen werken. Een beginnende gebruiker gebruikt bijvoorbeeld meer basale functies dan de geavanceerdere functies die een langer bestaande gebruiker gebruikt. Onze applicatie zou dit beter kunnen doen dan MeetJeStad.nl en Sensor Community door meer functies te bieden in de kaart, zodat zowel nieuwe als langdurige gebruikers gemakkelijk kunnen navigeren. Punt 8 (Aesthetic and minimalist design) beschrijft het visueel aantrekkelijk maken van de site. Voor onze opdracht is dit zeer belangrijk. We zouden dit kunnen implementeren door de ontdekte punten uit de doelgroep analyse toe te passen op de site en de kaart van Tilburg in grids te verdelen zodat de precieze locaties niet zichtbaar zijn met als doel dat de gebruiker meer privacy heeft. Deze grids zullen verdeeld zijn in wijken, om de scheidslijn duidelijker te houden. Verder zouden we in een container de metingen zichtbaar kunnen maken en met een legenda simpele termen gebruiken.

### UC09: Huidige metingen meetpunt

De gebruiker wil van een meetpunt de metingen inzien.

Sensor Community geeft deze data aan door kleine hexagonen op de precieze locaties. Zoals al eerder beschreven, wil de gebruiker wel de keuze hebben om te beslissen of het meetpunt openbaar te zien is of niet, dit om de privacy van de gebruiker te behouden. Als je op een punt klikt, krijg je een container te zien met veel technische termen. Een gemiddelde gebruiker zou niet snel begrijpen wat hier nou bedoeld wordt en wat er gemeten is. MeetJeStad.nl maakt ook hier gebruik van de thermostaat als punt. Deze wordt verdeeld in meerdere als je inzoomt op een specifieke plek. Na ingezoomd te hebben kun je op één punt klikken en krijg je netjes metingen over bepaalde data te zien. Wel staan deze gegevens in het Engels. In beide gevallen krijg je niet snel een goed beeld van wat er gaande is en wat er precies wordt gemeten.   
Punt 4 (Consistency and standards) geeft aan dat je consistent dient te blijven in je app. Dus bijvoorbeeld geen data in het Engels weer te geven als de rest van je site Nederlands gebruikt. Punt 8 (Aesthetic and minimalist design) beschrijft de visuele aspecten voor de websites en zouden in beide gevallen beter kunnen door überhaupt de gebruiker de keuze te geven om eigen meetpunten openbaar te maken of niet en deze mooier in de kaart te verwerken. Mensen willen graag mooie kaarten zien en geen punten of thermostaten. Voor onze applicatie zullen we de kaart wederom in grids verdelen en voor publieke meetpunten pinnen gebruiken. Deze meetpunten zullen met grafieken en simpele data de metingen laten zien.

### UC10: Gegevensdistributie bekijken

De gebruiker wil op de kaart een hittemap zien waarin de gekozen data wordt aangegeven.

MeetJeStad.nl laat zoals eerder al aangegeven de data zien per meetpunt als een eigen thermostaat. Dit is niet zozeer vergelijkbaar. Sensorcommunity laat zijn data in hexagonen zien. Deze nemen de gemiddelde data van alle stations binnen dit gebied, en geven het aan met een kleur. De betekenissen van deze kleuren staan in de legenda/thermostaat naast de kaart. Bij inzoomen zullen de hexagonen zich splitsen in kleinere hexagonen, met dus minder stations erin waardoor de informatie specifieker zal worden. Dit kan gedaan worden tot er in een enkele hexagoon maar één station zit. Hoewel dit maar voor een enkel type meting is, komt het wel meer overeen met wat wij verwachten.

Punt 6 (Recognition rather than recall) vraagt van ons dat de opties duidelijk zichtbaar en begrijpelijk zijn. Hierbij is het dus belangrijk voor ons om de te kiezen opties kort maar krachtig neer te zetten. Bij een geselecteerde optie zal de legenda zich dan ook moeten aanpassen om de informatie-flow goed te laten gaan. Punt 8 (Aesthetic and minimalist design) helpt hierbij ook mee om deze informatie zo duidelijk mogelijk te houden met zo min mogelijk woorden. Het feit dat deze informatie in een heatmap zichtbaar is in plaats van via getallen, helpt hier ook aan mee.

### UC11: Meetpunten van een gebied bekijken

De gebruiker wil de gegevens van specifieke meetpunten bekijken van een gebied.

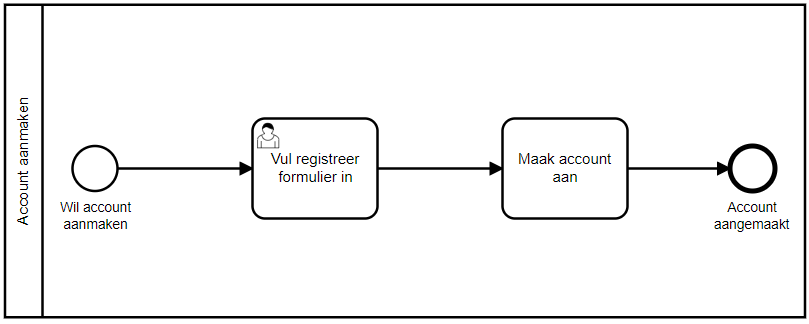
MeetJeStad.nl heeft hiervoor eigenlijk de stap van het gebied overgeslagen. Hierbij kan een gebruiker direct de gegevens van een meetpunt vinden. Dit kan gedaan worden door op het punt te klikken, waarna een pop-up de laatste meettijd, temperatuur en luchtvochtigheid geeft. Binnen deze pop-up is een link naar nog specifiekere gegevens van het station.  
Sensorcommunity heeft echter wel gebieden, in de vorm van de hexagonen. Door een gebied te selecteren, zal er een tabel verschijnen met alle sensoren en hun meetdata daarbinnen. Deze tabel kan zo klein blijven gezien de sensoren zelf niet veel anders meten dan dat ene gegeven. Door op een bepaalde sensor in de tabel te klikken, kan hierna een 24-uurs geschiedenis gezien worden van die specifieke sensor.

Het hele punt van de specifieke stations bekijken valt hierbij onder punt 7 (Flexibility and efficiency of use). Door de diepere data van individuele station op een andere pagina dan de kaart te plaatsen, kunnen de mensen die er dieper op in willen gaan hun eigen plek vinden. Zo blijft de informatie-overload weg van de “standaard” gebruikers. Punt 3 (User control and freedom) vraagt hierbij wel dat deze gebruikers, ook als ze per ongeluk op de details-pagina zijn gekomen, makkelijk terug kunnen. Door een simpele ‘terug naar kaart’ knop toe te voegen kan de gebruiker zo weer verder gaan waar hij gebleven was.

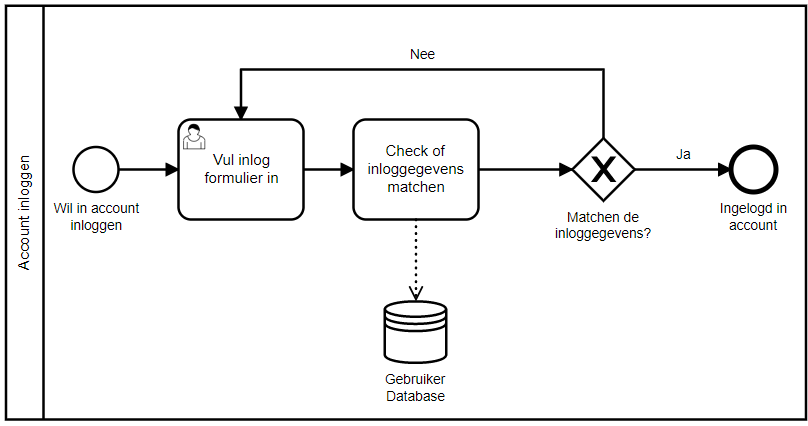
## BPMN Diagrammen

Deze diagrammen omschrijven het Business Proces van een use case.

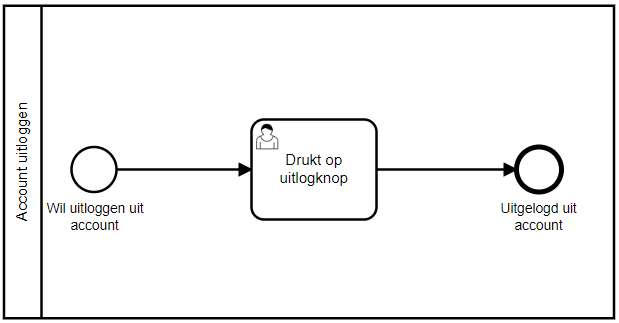
UC01:



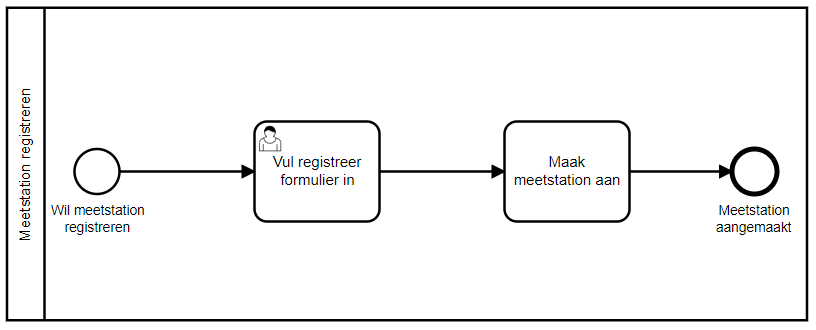
UC02:



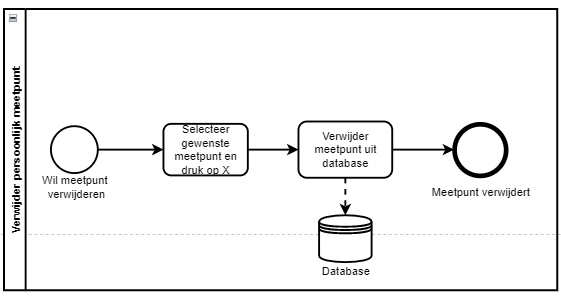
UC03:



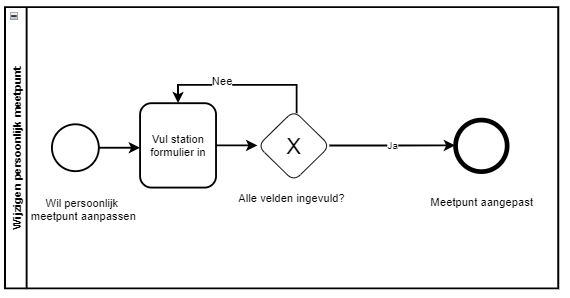
UC04:



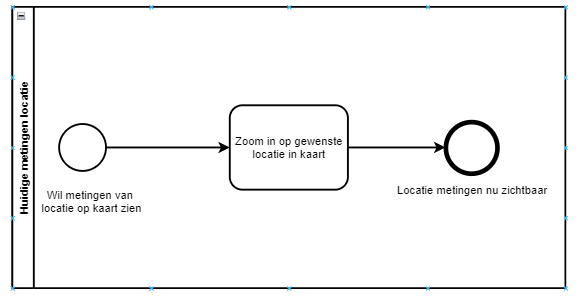
UC05:



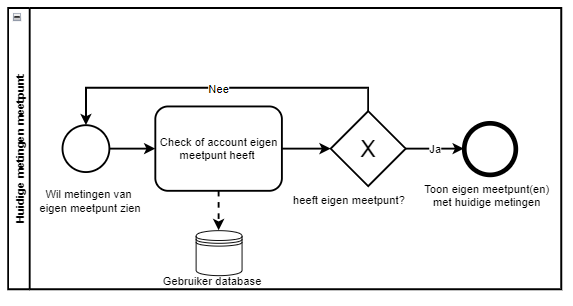
UC06:



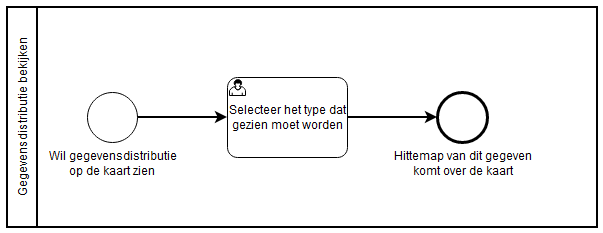
UC08:

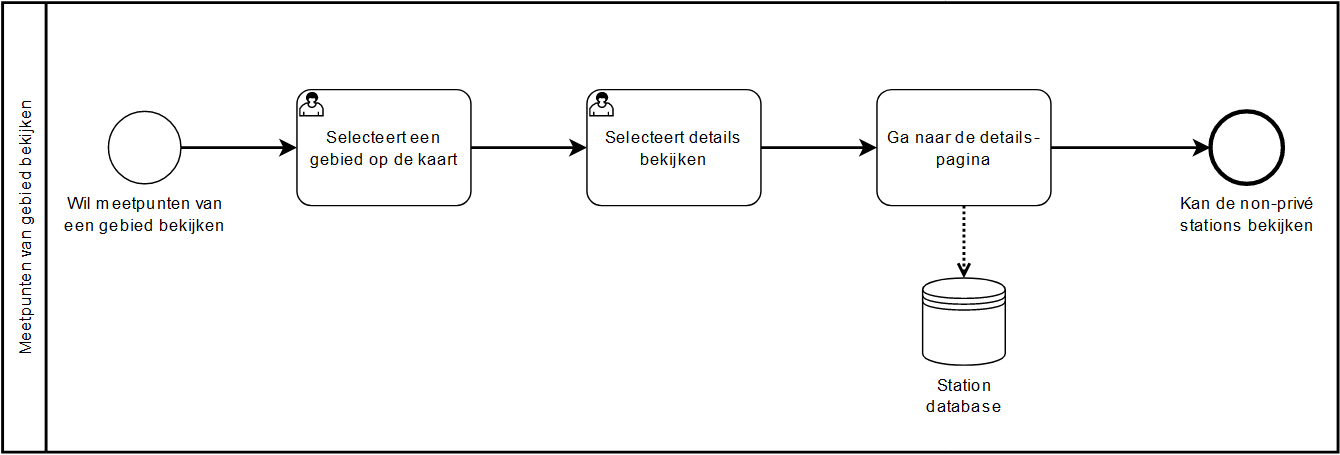


UC09:

’

UC10:

  
  
  
UC11:



# Domeinmodel

In onderstaande model staan op een hoog niveau alle mogelijke actoren en classes genoteerd, met de daarbij horende relaties.

