



Faculteit Ingenieurswetenschappen en Architectuur

Schakelprogramma tot Master of Science in de  
Industriële Wetenschappen: Informatica

Vakoverschrijdend project (E767014)

Projectgroep 3

# VOP Verkeerscentrum: Software-Analyse



Academiejaar 2015-2016

De Bock Jelle  
Floré Brent  
Stofferis Jeroen  
Vandemoortele Simon  
Vervenne Jan





Faculteit Ingenieurswetenschappen en Architectuur

Schakelprogramma tot Master of Science in de  
Industriële Wetenschappen: Informatica

Vakoverschrijdend project (E767014)

Projectgroep 3

# VOP Verkeerscentrum: Softwareanalyse

Academiejaar 2015-2016

De Bock Jelle  
Floré Brent  
Stofferis Jeroen  
Vandemoortele Simon  
Vervenne Jan

## Contents

Inleiding .....	5
Analyse .....	6
1 User Stories.....	7
2 Use cases diagram .....	7
3 Use cases .....	9
3.1 Use Case 1: Meld abnormaliteiten .....	9
3.2 Use Case 2: Bekijk reistijden .....	10
3.3 Use Case 3: Controleer data .....	11
3.4 Use Case 4: Bekijk actuele verkeerssituatie.....	12
3.5 Use Case 5: Vergelijk verkeersinformatie .....	13
3.6 Use Case 6: Data ophalen en hergebruiken.....	14
3.7 Use Case 7: Oorzaken verkeersproblemen bekijken .....	15
Ontwerp .....	1
4 Inleiding .....	2
5 Providerapplicatie.....	2
5.1 State diagram.....	2
5.2 Design Class Diagram (DCD).....	3
Detailoverzicht scrapers .....	4
Detailoverzicht domein .....	4
Detailoverzicht repository/controller .....	5
Legende .....	5
5.3 Algoritmes.....	7
API call maken .....	7
5.4 Generators .....	7
5.5 Scrapers.....	8
5.6 Repositories .....	8
6 Webapplicatie.....	9
6.1 Algoritmes.....	9
Home pagina bezoeken .....	9
7 Databank.....	10
7.1 Entity-relationship diagram (ERD) .....	10
Testen .....	11
8 Testplan .....	12
8.1 Scope definitie .....	12
8.2 Unit testen .....	12
Uitvoeren.....	12
Scenario's .....	12

Voorziene testen .....	12
8.3    Integration testen .....	13
Bekijk reistijden .....	13
Bekijk actuele verkeerssituatie.....	13
Vergelijk verkeersinformatie .....	14
Waypoints traject aanpassen .....	14
8.4    Usability testen .....	14
Bekijk reistijden .....	15
Bekijk actuele verkeerssituatie.....	15
Vergelijk verkeersinformatie .....	15
Waypoints traject aanpassen .....	15
Installatie en onderhoud .....	17
9    Installatie project op geconfigureerde omgeving .....	18
Retrospective.....	19
10    Realisaties .....	20
11    Mogelijke verbeteringen .....	20
12    Realisaties per persoon .....	21
Appendix.....	23
13    Configuratie van server en installatie project .....	24
13.1    Installatie .....	24
Server.....	24
GlassFish .....	31
Desktop.....	36
Deployment .....	37
14    Analyse providers en reistijden .....	39
14.1    Inleiding .....	39
14.2    Het project .....	40
Doel .....	40
Werking .....	40
Trajecten.....	41
14.3    Providers .....	42
Google Maps.....	42
Bing Maps .....	44
Coyote Systems .....	46
Waze .....	49
Here .....	51
TomTom .....	55
14.4    Samenvatting .....	58



## Inleiding

Deze softwareanalyse is er gekomen naar aanleiding van het vakoverschrijdend eindproject. Hierbij is het de bedoeling om in teamverband een reëel project uit te werken. Het project dat ons toegewezen werd is in opdracht van stad Gent.

Dit document is het resultaat van de software analyse. Het is specifiek gericht tot de vereisten van de te ontwerpen software en de daarbij horende functionaliteiten. Het bevat onder meer het ontwerpdocumenten van het ontwikkelingsproces.

Het doel van deze analyse is om de vereisten van de klant te gaan vertalen in deelproblemen en deze zo beknopt mogelijk te formuleren. Het herwerken van deze bundel van vereisten is essentieel om een goeie oplossing te bekomen.

Hoofdstuk één behandelt de user story's. Deze pagina bevat een korte beschrijving van de vereisten, gecombineerd met hun gewicht en prioriteit. De twee volgende hoofdstukken, use case diagram en use cases, beschrijven dan weer concrete handelingen die de gebruikers uitvoeren op het systeem. Hoofdstuk vier en vijf bevatten respectievelijk het klassediagram en het databaseschema. Het document eindigt met enkele aandachtspunten voor de volgende sprint.

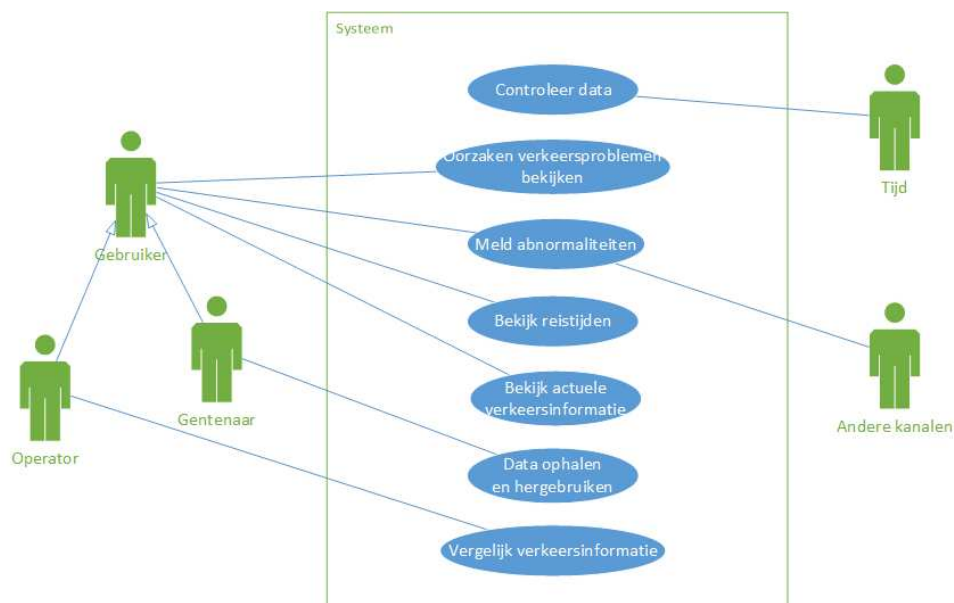
Analyse



## 1 User Stories

Prioriteit	Gewicht	Opgenomen als issue	User	Story
1	10	x	Gebruiker	De reistijden van trajecten afkomstig van verschillende providers kunnen vergelijken.
2	3	x	Gebruiker	De historiek van de reistijden kunnen vergelijken.
3	4	x	Operator	De actuele verkeerssituatie raadplegen op een kaart.
4	4	x	Operator	De ideale reistijden van trajecten vergelijken met de huidige reistijden.
5	5		Ontwikkelaar	Ophalen van verschillende gegevens via een API.
6.1	6	X	Operator	Raadplegen van een dashboard om trajecten op verschillende momenten, verschillende trajecten,... te vergelijken.
6.2	3		Operator	Verwittiging ontvangen bij ernstige gebeurtenissen.
6.3	2		Operator	Instellen na hoeveel minuten een boodschap verstuurd wordt als er vertragingen zijn.

## 2 Use cases diagram





### 3 Use cases

#### 3.1 Use Case 1: Meld abnormaliteiten

<b>Use Case naam</b>	Meld abnormaliteiten
<b>Use case nummer</b>	1
<b>Toepassing</b>	Back-end
<b>Beschrijving</b>	Een abnormaliteit wordt automatisch gemeld via de beschikbare kanalen.
<b>Primaire Actoren</b>	het systeem
<b>Stakeholders</b>	De gebruikers van de communicatiekanalen en operator.
<b>Precondition</b>	nvt
<b>Trigger</b>	Een bepaalde hoeveelheid vertraging werd overschreden op een welbepaald traject.
<b>Basisverloop</b>	1. Het systeem detecteert een abnormale vertraging.
	2. Het systeem selecteert de beschikbare kanalen die verwittigd moeten worden.
	3. Het systeem stuurt een melding naar de beschikbare kanalen.
	4. Het systeem rapporteert zijn tussenkomst.
<b>Alternatief verloop</b>	nvt
<b>Domeinregels</b>	1. De drempel waarop een vertraging abnormaal wordt kan door een operator ingesteld worden.
	2. Mogelijke kanalen zijn
	- Twitter
	- Opvallende plaats op de website
	- Native apps

### 3.2 Use Case 2: Bekijk reistijden

<b>Use Case naam</b>	Bekijk reistijden
<b>Use case nummer</b>	2
<b>Toepassing</b>	front end
<b>Beschrijving</b>	De actor bekijkt de reistijden die voor hem relevant zijn.
<b>Primaire Actoren</b>	Gebruiker, operator
<b>Precondition</b>	nvt
<b>Postcondition</b>	De actor heeft een overzicht van de reistijden.
<b>Trigger</b>	nvt
<b>Basisverloop</b>	1. De actor wenst een reistijd te bekijken.
	2. Het systeem retourneert een overzicht van de beschikbare trajecten.
	3. De actor kiest het gewenste traject.
	4. Het systeem retourneert het overzicht van het geselecteerde traject.
	5. De actor bekijkt de gewenste reistijden.
<b>Alternatief verloop</b>	5A. De actor wenst uitgebreide informatie van het traject.
	6. Het systeem retourneert de beschikbare opties.
	7. De actor kiest de gewenste opties.
	8. Het systeem retourneert de uitgebreide informatie van het geselecteerde traject.
	9. De actor bekijkt de uitgebreide informatie.
<b>Domeinregels</b>	1. Beschikbare uitgebreide opties zijn:
	- Historische reistijden, te selecteren voor een bepaalde periode
	- Filter reistijden op basis van provider
	2. De reistijden worden gepresenteerd in het volgende formaat: [hh:mm:ss]

### 3.3 Use Case 3: Controleer data

<b>Use Case naam</b>	Controleer data
<b>Use case nummer</b>	3
<b>Toepassing</b>	back-end
<b>Beschrijving</b>	Periodiek de data in de database controleren en corrigerende updates doorvoeren.
<b>Primaire Actoren</b>	systeem
<b>Preconditie</b>	nvt
<b>Postconditie</b>	De optimale reistijden zijn representatief voor de huidige verkeerssituatie op macroniveau.
<b>Trigger</b>	Elke week op middernacht.
<b>Basisverloop</b>	1. Het systeem verzamelt de reistijden van alle providers voor de trajecten.
	2. Het systeem herberekend de optimale reistijden voor de trajecten op basis van de reistijden.
	3. Het systeem corrigeert de optimale reistijden indien nodig.
<b>Alternatief verloop</b>	nvt
<b>Domeinregels</b>	1. Voor de herberekening van de optimale reistijd moet met volgende factoren rekening gehouden worden:
	* Voorzien de providers al dan niet een optimale reistijd voor de trajecten.
	* Enkel de realtime reistijden op een rustiger tijdstip (21u tot 6u) zijn bruikbaar voor de analyse.
	2. De optimale reistijd wordt berekend aan de hand van de gemiddelde van de bruikbare data

### 3.4 Use Case 4: Bekijk actuele verkeerssituatie

<b>Use Case naam</b>	Bekijk actuele verkeerssituatie
<b>Use case nummer</b>	4
<b>Toepassing</b>	front end
<b>Beschrijving</b>	De actor bekijkt de actuele verkeerssituatie
<b>Primaire Actoren</b>	Gebruiker, Operator
<b>Stakeholders</b>	/
<b>Precondition</b>	/
<b>Postcondition</b>	De actor beschikt over de actuele verkeerssituatie.
<b>Basisverloop</b>	1. De actor wenst de actuele verkeerssituatie te bekijken.
	2. Het systeem haalt de reistijden en gebeurtenissen van de routes op.
	3. Het systeem geeft de routes met bijhorende reistijden en eventuele gebeurtenissen weer.
<b>Alternatief verloop</b>	2A. Het systeem kan geen routes met bijhorende data ophalen.
	2A1. Het systeem geeft een foutboodschap weer.
	2A2. Einde use case zonder bereiken postconditie.
<b>Domeinregels</b>	1. Verkeerssituatie wordt weergegeven op een kaart.

### 3.5 Use Case 5: Vergelijk verkeersinformatie

<b>Use Case naam</b>	Vergelijk verkeersinformatie
<b>Use case nummer</b>	5
<b>Toepassing</b>	front end
<b>Beschrijving</b>	De actor vergelijkt actuele of historische verkeerssituaties
<b>Primaire Actoren</b>	Operator
<b>Stakeholders</b>	/
<b>Precondition</b>	/
<b>Postcondition</b>	De actor kan verschillende verkeerssituaties met elkaar vergelijken.
<b>Basisverloop</b>	1. De actor wenst verkeerssituaties te vergelijken.
	2. Het systeem geeft de mogelijke vergelijkingen weer.
	3. De actor selecteert een type vergelijking.
	4. Het systeem valideert en haalt een overzicht op van de mogelijke verkeerssituaties.
	5. De actor selecteert een aantal verkeerssituaties om te vergelijken.
	6. Het systeem valideert en haalt de detailgegevens op van de relevante verkeerssituaties
	7. Het systeem geeft de detailgegevens weer.
<b>Alternatief verloop</b>	4A. Het systeem kan geen relevante verkeerssituaties ophalen.
	4A1. Het systeem geeft een foutboodschap weer.
	4A2. Ga naar 3
	6A. Het systeem kan geen detailgegevens ophalen.
	6A1. Het systeem geeft een foutboodschap weer.
	6A2. Ga naar 5
<b>Domeinregels</b>	<b>Mogelijke types vergelijking:</b>
	1. Gegevens van verschillende momenten(uren, dagen, speciale perioden)
	2. Gegevens van verschillende trajecten
	3. Verkeerssituatie met bezettingsgraad bepaalde parking

### 3.6 Use Case 6: Data ophalen en hergebruiken

<b>Use Case naam</b>	Data ophalen en hergebruiken
<b>Use case nummer</b>	6
<b>Toepassing</b>	front end
<b>Beschrijving</b>	De actor maakt een API call om de verzamelde data te krijgen en te hergebruiken
<b>Primaire Actoren</b>	Gebruiker
<b>Stakeholders</b>	/
<b>Precondition</b>	/
<b>Postcondition</b>	De actor beschikt over de verzamelde verkeersdata.
<b>Basisverloop</b>	1. De actor wenst zelf een nieuwe toepassing te maken adhv de verkeersdata.
	2. De actor maakt een API call naar de gepaste URL naar gelang de gewenste data.
	3. De actor krijgt de data terug in JSON formaat.
<b>Alternatief verloop</b>	3A. Er is geen data beschikbaar die voldoet aan de eisen van de actor, dus wordt een correcte response teruggestuurd.
	3B. Er wordt een foutboodschap teruggestuurd.
<b>Domeinregels</b>	1. Verkeersituatie wordt weergegeven op een kaart.



### 3.7 Use Case 7: Oorzaken verkeersproblemen bekijken

<b>Use Case naam</b>	Oorzaken verkeersproblemen bekijken
<b>Use case nummer</b>	7
<b>Toepassing</b>	front end
<b>Beschrijving</b>	De actor maakt een API call om de verzamelde data te krijgen en te hergebruiken
<b>Primaire Actoren</b>	Gebruiker
<b>Stakeholders</b>	/
<b>Precondition</b>	/
<b>Postcondition</b>	De actor beschikt over de verzamelde oorzaken van verkeersproblemen.
<b>Basisverloop</b>	1. De actor wenst zelf een nieuwe toepassing te maken adhv de verkeersdata.
	2. De actor maakt een API call naar de gepaste URL naar gelang de gewenste data.
	3. De actor krijgt de data terug in JSON formaat.
<b>Alternatief verloop</b>	3A. Er is geen data beschikbaar die voldoet aan de eisen van de actor, dus wordt een correcte response teruggestuurd.
	3B. Er wordt een foutboodschap teruggestuurd.
<b>Domeinregels</b>	1. Verkeersproblemen worden weergegeven in tabelformaat



Ontwerp

## 4 Inleiding

Dit onderdeel van het dossier is voornamelijk gericht op de structuur van de applicaties zelf. Dit is nuttig voor programmeurs om zich snel in te kunnen werken of voor onderhoud op middellange termijn te kunnen vergemakkelijken. Het ecosysteem dat instaat voor het opvolgen van het verkeer in en rond Gent valt uit elkaar in enkele verschillende applicaties en ondersteunende componenten.

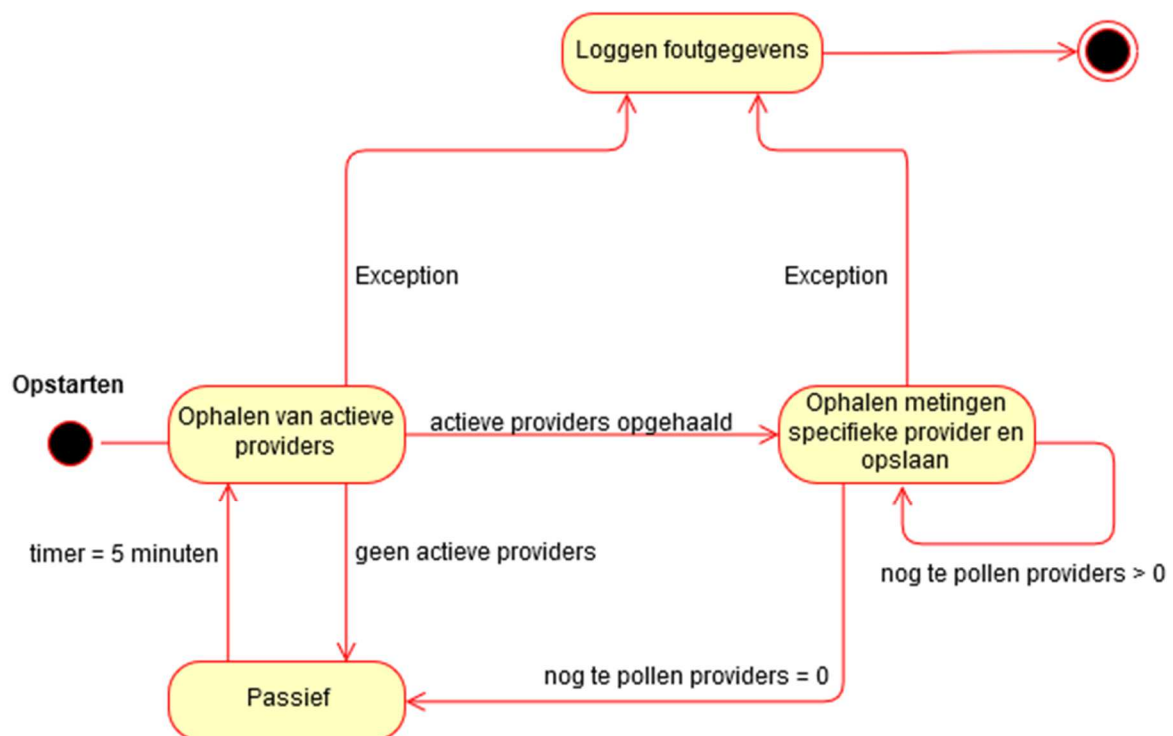
Allereerst is er de providerapplicatie. Deze applicatie staat in voor het periodiek opvragen van de verkeerssituatie en het wegschrijven van deze informatie naar een databank. De applicatie zelf staat in voor het wachten tot de volgende reeks gegeven in verband met het verkeer moeten worden opgehaald. Deze planning wordt dus niet vanuit het besturingsysteem opgelegd zoals met CRON bevoorbeeld.

Naast de providerapplicatie is er ook een webapplicatie. Deze kan gebruikt worden door mensen van het verkeerscentrum of inwoners van Gent om informatie op te halen in verband met de huidige verkeerssituatie. Hiernaast zijn er voor mensen van het verkeerscentrum nog meer mogelijkheden zoals het wijzigen van de op te volgen trajecten, geschiedenis van de verkeerssituatie vergelijken en meer.

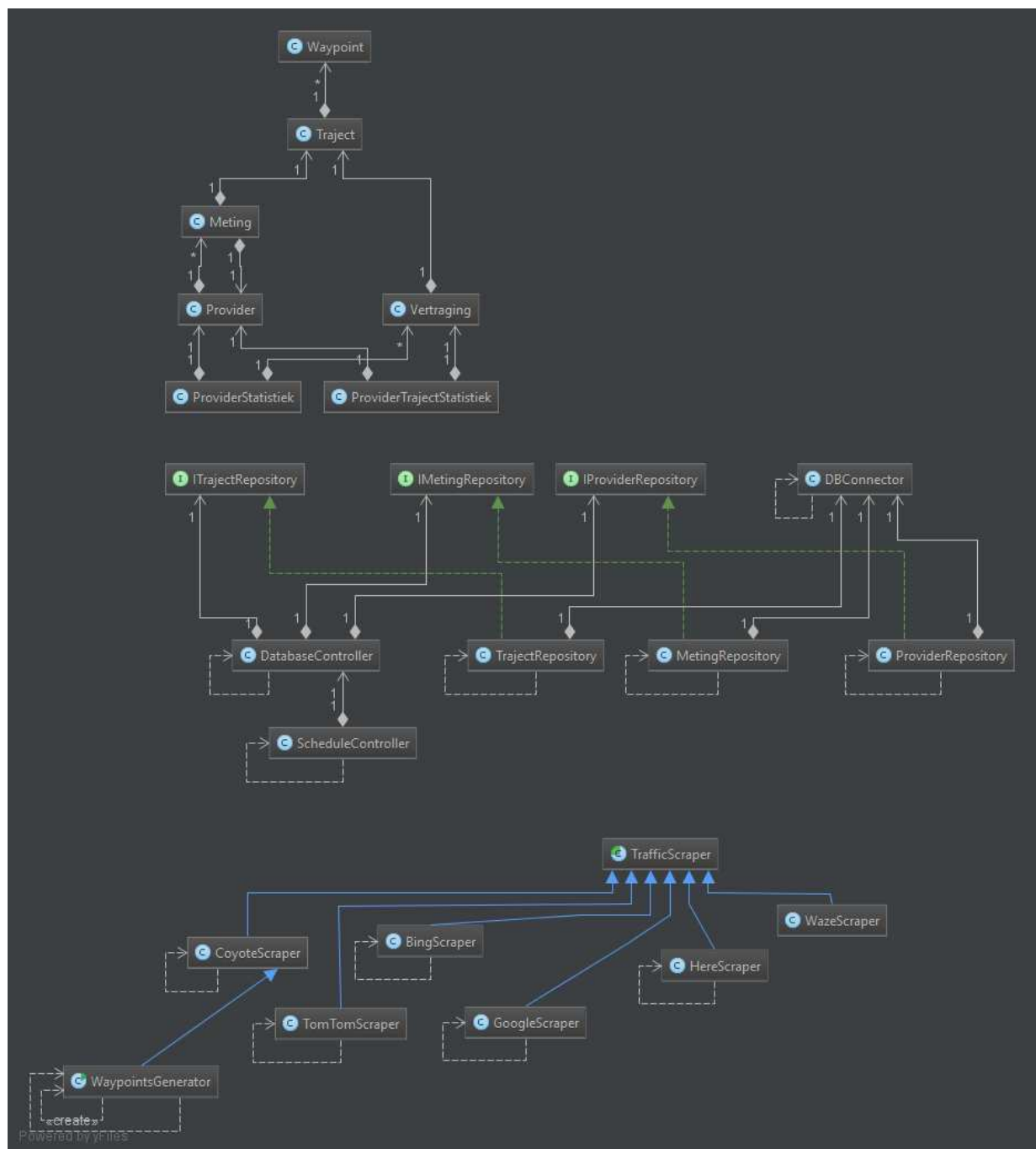
Natuurlijk is er ook een databank aanwezig achter de schermen waarin alle informatie wordt bijgehouden.

## 5 Providerapplicatie

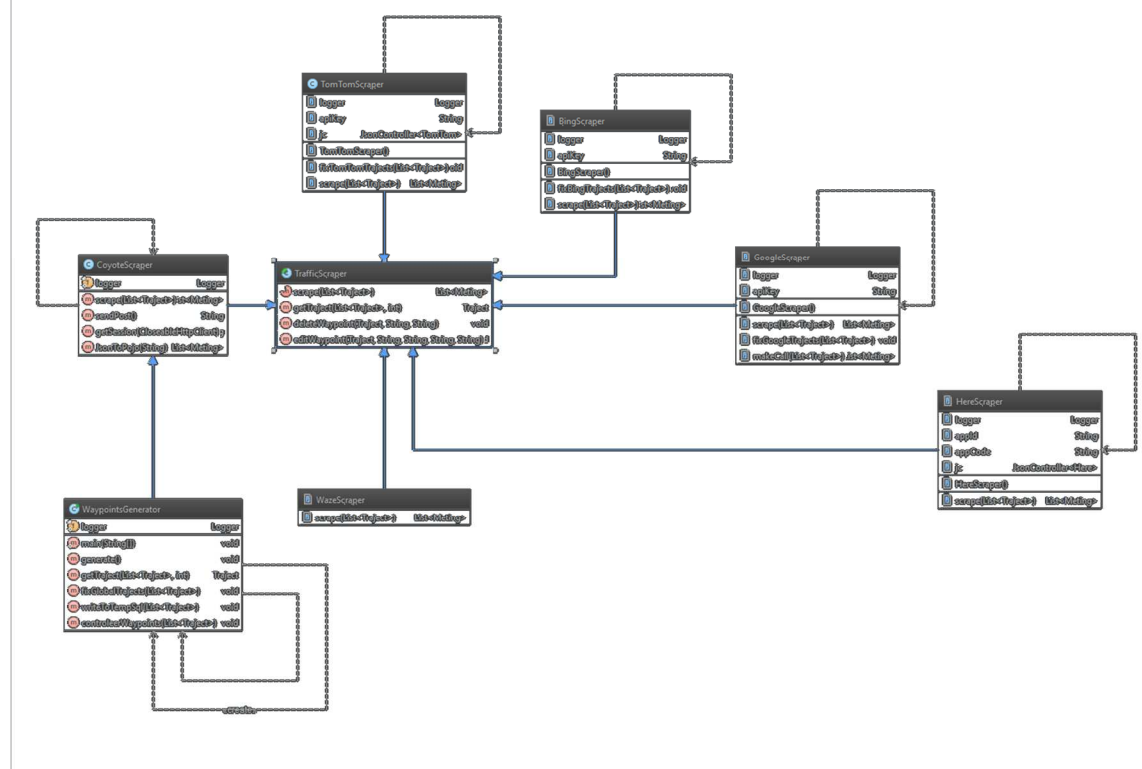
### 5.1 State diagram



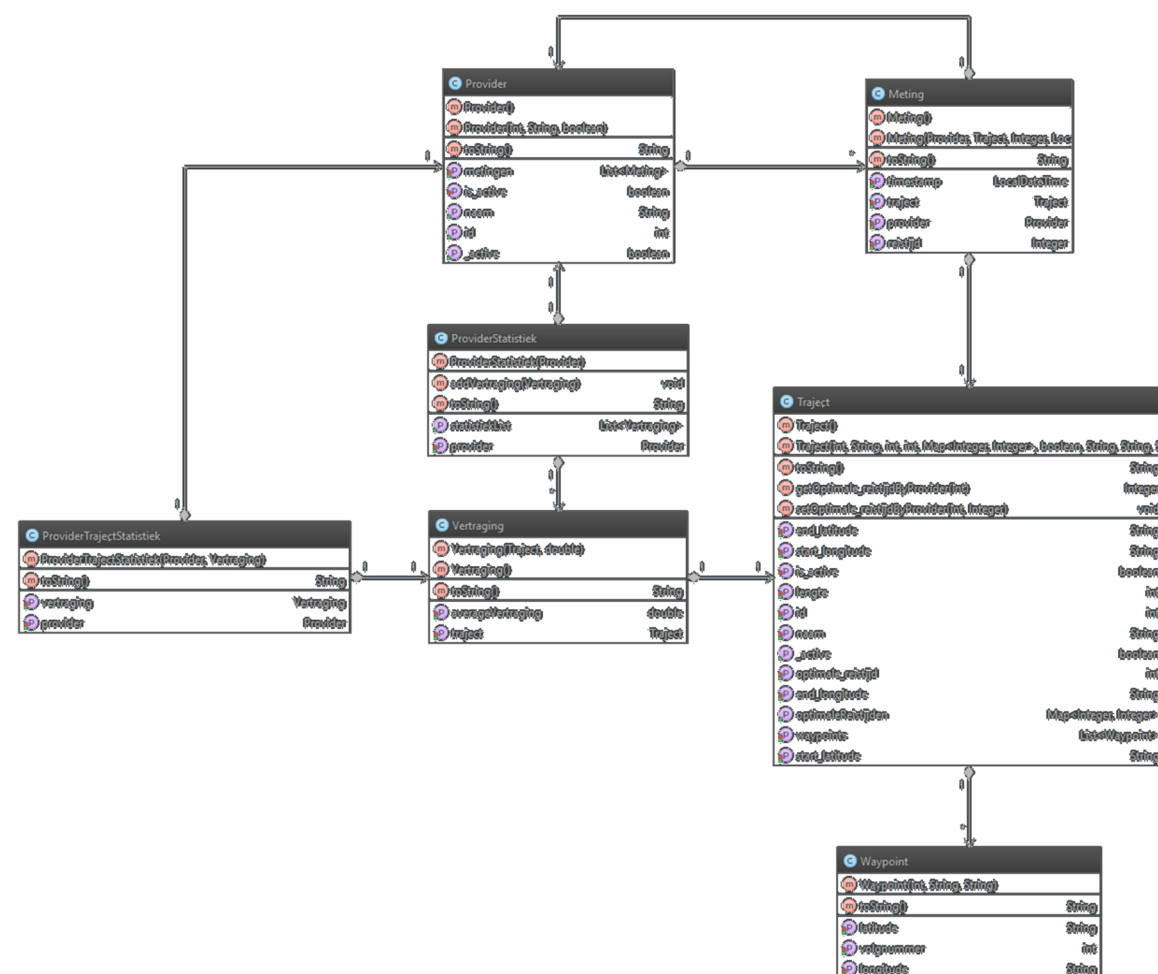
## 5.2 Design Class Diagram (DCD)



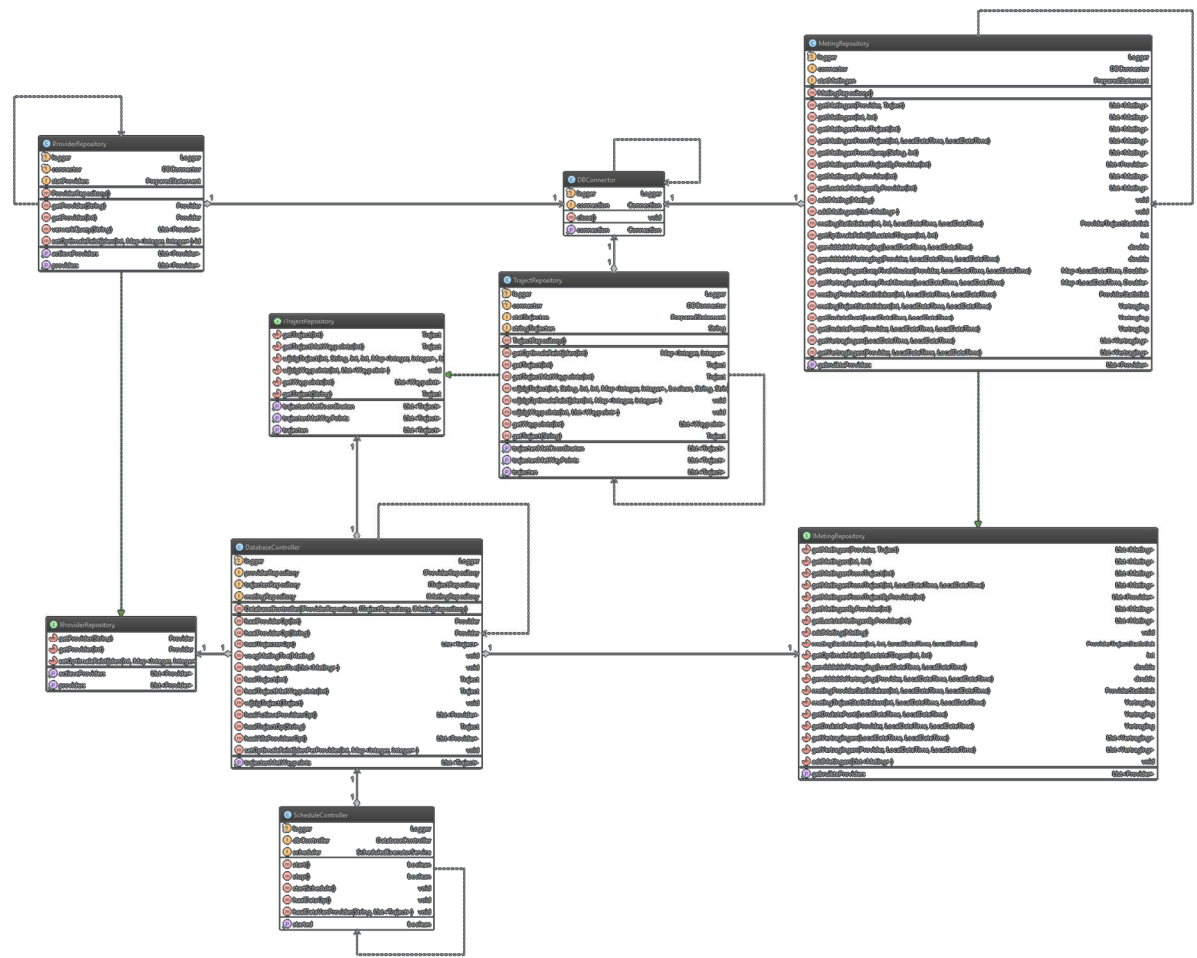
## Detailoverzicht scrapers



## Detailoverzicht domain





## Detailoverview repository/controller



## Legende

Icon	Description
	Class
	Abstract Java class
	Annotation
	Enumeration
	Exception
	Final Java class
	Interface
	Java class that contains declaration of the main() method.
	Test case
	Java class located out of the source root. Refer to the section <a href="#">Configuring Content Roots</a> for details.
	Java class <a href="#">excluded from compilation</a> .
	Method
	Abstract method
	Field

	Variable
	Property
	Property with getter
	Property with setter
	Parameter
	Element
	Directory
	Module
	Group of modules
	Package
	Source root
	Test root
	Excluded root
	Resources
	Test resources
	Generated source roots
	Generated test source roots
<b>Visibility modifiers</b>	
	Read-only class, e.g. from a jar of an external library.
	private
	protected
	package protected
	static
	public



## 5.3 Algoritmes

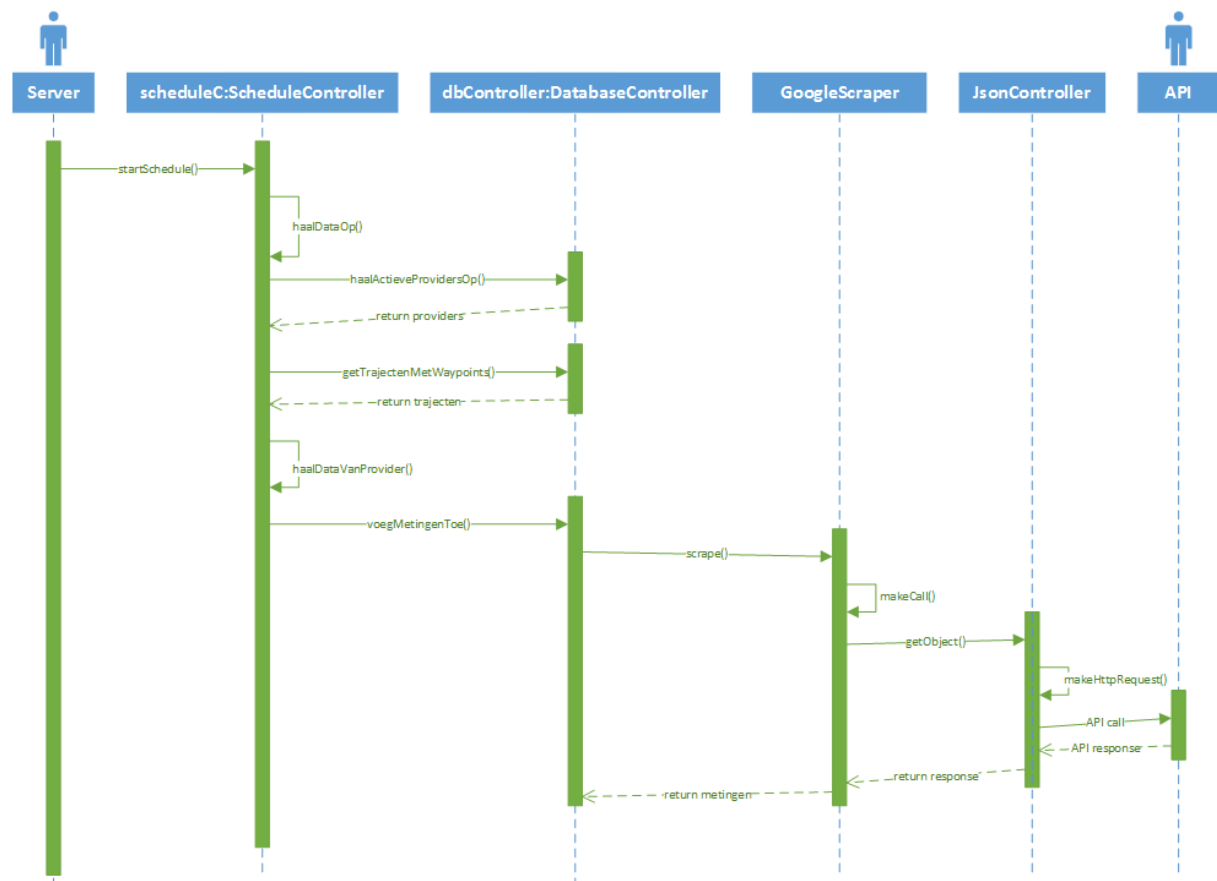
### API call maken

Een API call maken, en de database vullen met metingen van de verschillende routes, wordt gestart door de ScheduleController. Deze haalt eerst de benodigde data op om de API call te kunnen maken namelijk alle beschikbare providers naar waar een API call gemaakt kan worden, en alle trajecten waarvoor een API call gemaakt moet worden.

Eenmaal al de benodigde data verzameld is, kunnen metingen gemaakt worden en toegevoegd worden aan de database. Daarvoor wordt de DatabaseController en een Scraper opgeroepen. De Scraper maakt een call, waarbij een request wordt gestuurd naar de JsonController gestuurd wordt. De JsonController maakt op zijn beurt dan een API call naar de API en vangt zijn response op. Deze response wordt dan terug gestuurd naar de Scraper die de oproep deed.

De Scraper controleert dan op zijn beurt de response. Als de response correcte data bevat, dan wordt de meting toegevoegd in de database. Als de response een error of geen data bevat, dan wordt een lege meting toegevoegd in de database.

Voor elke provider wordt de DatabaseController elke 5 minuten opgeroepen en wordt de bovenstaande procedure doorlopen.



## 5.4 Generators

Voor het project zijn er twee generators aanwezig, deze zijn beschikbaar vanuit het admin paneel.

De eerste generator, de *OptimaleReistijdGenerator* wordt gebruikt om optimale reistijden te genereren per provider voor elk traject, zonder dat de effectieve reistijden gekend zijn. Zo kunnen er rekening houdend met het huidige uur, procentuele optimale reistijden aangemaakt worden voor het geval dat er geen data aanwezig is in de databank.

De tweede generator, de *WaypointsGenerator* wordt gebruikt om te zorgen dat voor elk traject correcte waypoints gebruikt worden. De generator overloopt de lijst met waypoints voor elk traject, en verwijdert de waypoints die niet kunnen gebruikt worden door de scrapers.

## 5.5 Scrapers

Er zijn verschillende scrapers aanwezig, voor elke provider één. De *BingScraper*, *CoyoteScraper*, *GoogleScraper*, *TomTomScraper* en *HereScraper* werken volledig vanuit java. Als een scraper opgeroepen worden vragen ze eerst de lijst met trajecten en hun waypoints op, en maken daarna een API call naar de REST API van de overeenkomstige provider om de data op te vragen. Daarna wordt de data weggeschreven naar de databank als deze aanwezig is, of wordt er een lege rij toegevoegd als er geen data teruggekregen wordt.

De *WazeScraper* wordt vanuit java opgeroepen, maar werkt vanuit perl. Voor de Waze provider was er geen API beschikbaar, dus wordt de web applicatie zelf gescraped voor data. Deze scraper werkt, maar het opvragen van de data duurt te lang, dus is deze niet geactiveerd.

Naast de scrapers om data over het traject op te vragen, is er ook een *HereIncidentScraper*. Deze wordt gebruikt om informatie over eventuele verkeersproblemen voor elk traject op te vragen die bijgehouden worden door Here.

Sommige van de scrapers bevatten ook nog een extra functie die sommige waypoints aanpast per provider, omdat de originele waypoints voor die provider niet op de weg liggen. Daarom worden die waypoints handmatig aangepast.

## 5.6 Repositories

Om de communicatie naar de database te voorzien, zijn er verschillende repositories beschikbaar. De *TrajectRepository*, *ProviderRepository*, *LoginRepository*, *IncidentRepository* en *MetingRepository* verzorgen elk de queries die te maken hebben met de trajecten, de providers, het inloggen, de incidenten en de metingen. Zo moet de query niet telkens opnieuw geschreven worden omdat slechts één enkele waarde veranderd.

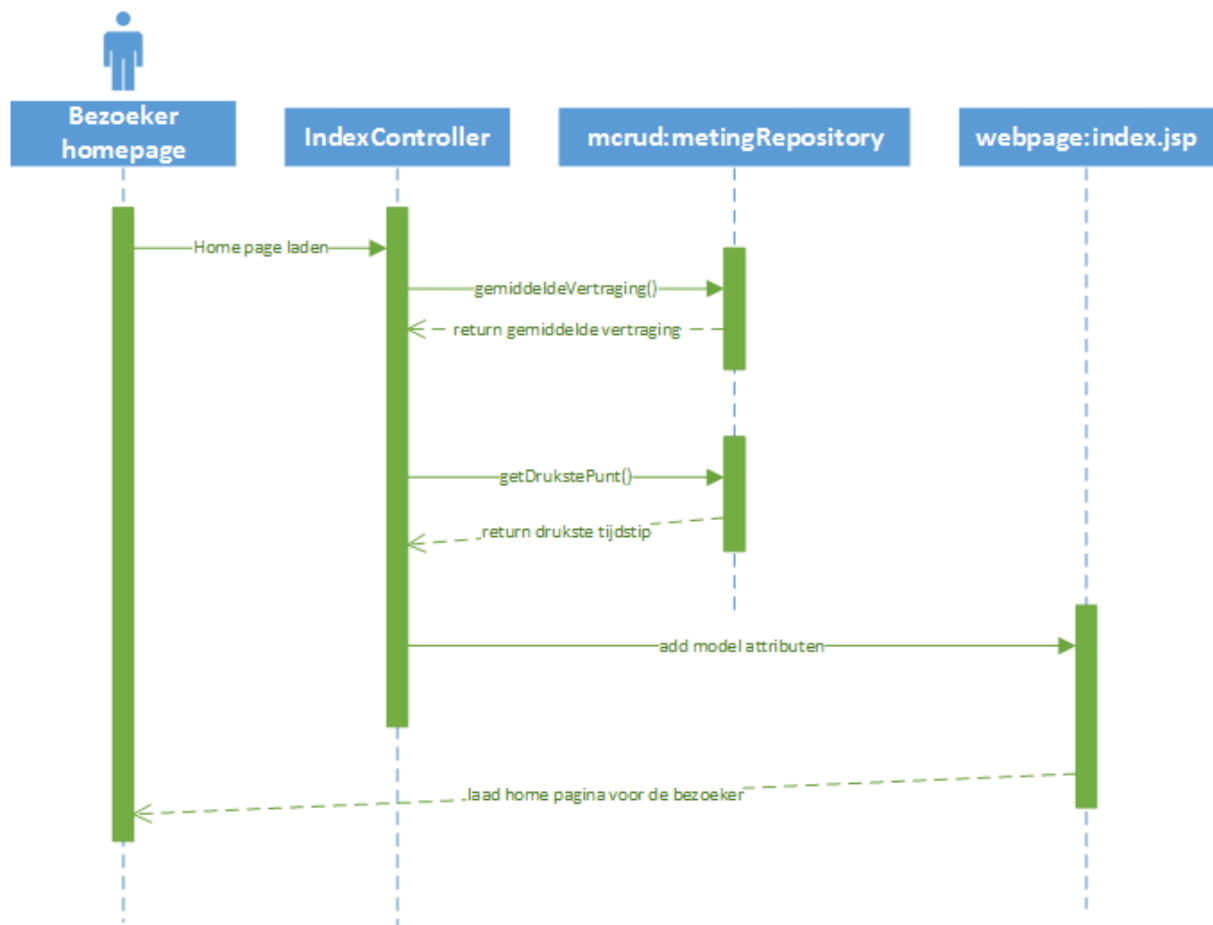
## 6 Webapplicatie

### 6.1 Algoritmes

#### Home pagina bezoeken

Als een bezoeker de home pagina wil bezoeken, dan wordt deze geladen door de IndexController.

De IndexController vraagt eerst via de metingRepository de gemiddelde vertraging en het drukste traject met het drukste punt op, zodat deze als attributen kunnen meegegeven worden aan de index javaserver page. Deze index.jsp wordt dan met deze attributen geladen door de IndexController, zodat de homepagina geladen kan worden voor de bezoeker.



## 7 Databank

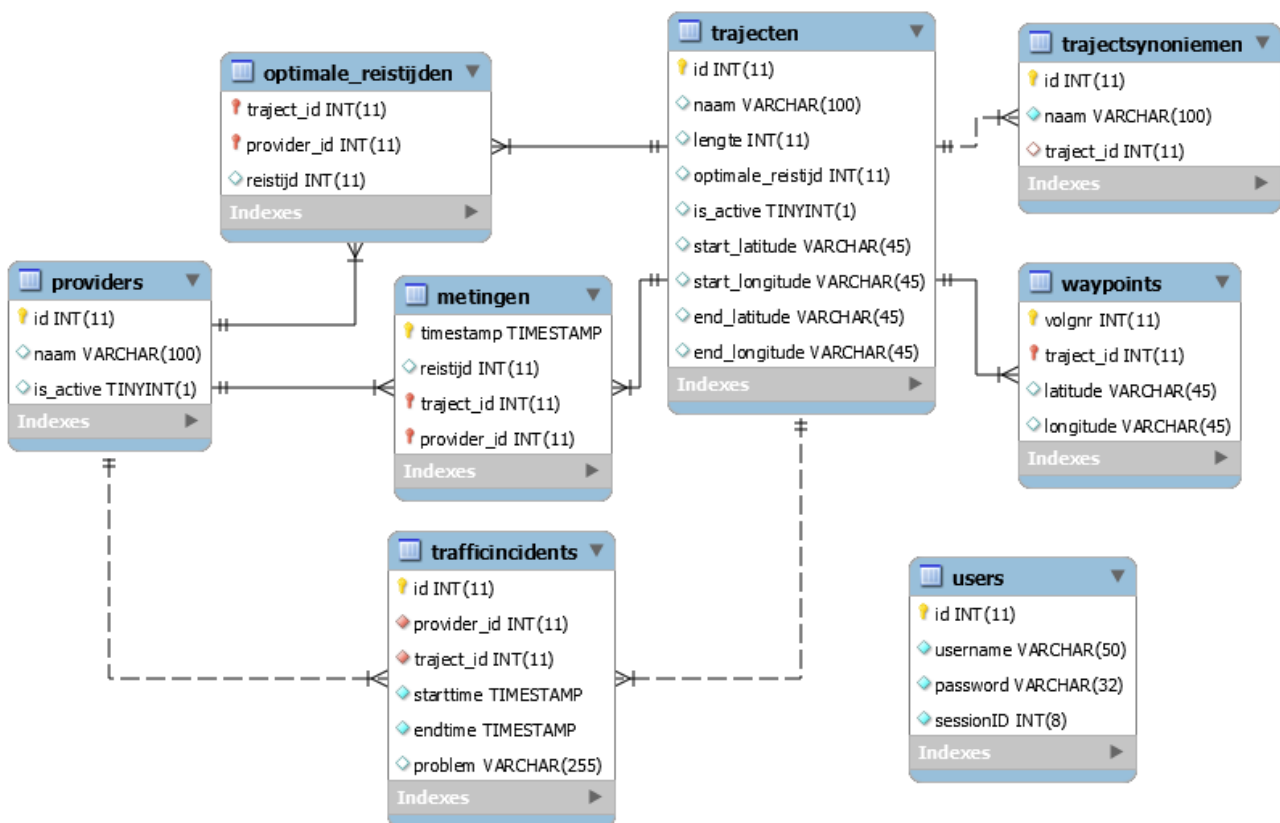
De databank bestaat uit zes tabellen. Een tabel voor de providers, drie tabellen voor de trajecten en één voor de metingen.

De tabel voor de providers houdt, zoals de naam al doet vermoeden, bij welke providers gebruikt worden. Er is eveneens een aanduiding mogelijk om ze al dan niet te activeren.

Voor de metingen wordt enkel de reistijd bijgehouden per traject en per provider, samen met een timestamp van de meting.

Trajecten zijn iets complexer en bestaat daarom uit vier tabellen. De eerste houdt voor elk traject de basisdata bij: coördinaten, naam, lengte en optimale reistijd. Deze optimale reistijd is een globale waarde voor alle providers per traject. In de tweede tabel wordt dan de optimale reistijd per traject en per provider bijgehouden. De derde tabel houdt per traject extra waypoints bij zodat deze gebruikt kunnen worden bij de API calls. Ze zijn een noodzaak voor het ophalen van correcte data.

### 7.1 Entity-relationship diagram (ERD)



Testen

## 8 Testplan

Het testplan bestaat uit 3 discrete delen. De unit testen, die automatisch worden uitgevoerd bij elke build van het programma; de integration testen die uitgevoerd worden op een lokale testomgeving voor deze wordt gedeployed in productie en de usability testen, die momenteel niet uitgevoerd zijn, maar waar wel een scenario voor aanwezig is om te overlopen en verder uit te breiden indien de geleverde software verder ontwikkeld en verfijnd zal worden.

### 8.1 Scope definitie

Tijdens de testfase worden meerdere onderdelen van de applicatie getest. De aandacht zal vooral gevestigd worden op de interne functionaliteit, bruikbaarheid en de interactie met gebruikers.

Om een testplan op te stellen is het belangrijk om de scope te definiëren. De scope kan opgesplitst worden in enkele categorieën: gebruiksvriendelijk, functionaliteit en abstractie (voor toekomstige ontwikkelaars).

Onder gebruiksvriendelijkheid vallen alle testen die betrekking hebben met de userinterface van de applicatie. De functionaliteit zal getest worden door unit en integration testen. Het testen van de abstractie is een proces dat continue uitgevoerd moet worden tijdens de ontwikkeling van de software. Het uittesten hiervan vereist dat de testers de broncode doornemen en op zoek gaan naar mogelijkheden om verder te abstraheren.

Wanneer een scope gedefinieerd wordt is het net zo belangrijk om te vermelden welke elementen geen onderdeel uitmaken van de scope. De scope wordt beperkt tot de ontwikkelde applicatie zelf. Interactie met onderliggende lagen zoals het besturingssysteem, GlassFish en MariaDB worden niet getest. Er wordt uitgegaan van een werkende omgeving waarop het programma kan draaien. Tot slot wordt het stresstesten van de applicatie eveneens uit de scope gehouden.

### 8.2 Unit testen

De unit testen controleren de individuele componenten van de software in een gecontroleerde omgeving. Zo wordt van elk essentieel component getest of het verwacht gedrag vertoont. Dit kan nuttig zijn om onverwachte gevolgen van aanpassingen snel op te kunnen sporen en verhelpen.

#### Uitvoeren

Nieuwe testen worden geschreven voor het schrijven van nieuwe code. De testen worden ook allemaal steeds uitgevoerd voor de laatste wijzigingen naar de masterbranch worden gecommit. Enkel indien alle testen slagen wordt nieuwe code op de masterbranch gezet. Op de branch van de programmeur is dit op korte termijn minder belangrijk. Hoewel dit de best practice is, werd dit niet correct en consistent toegepast.

#### Scenario's

Bij het testen worden steeds enkele gevallen gecontroleerd. Een correct geval, een foutief geval en enkele mogelijkheden die op de grenzen van deze situatie vallen. Denk hierbij aan randgevallen, extreem grote of kleine waarden, speciale tekens of bijzondere waarden. Een concreet voorbeeld hiervan zijn trajecten zonder waypoints of metingen zonder timestamp.

#### Voorziene testen

Momenteel zijn er enkel concrete testen geschreven voor de databaseController. De focus lag hierbij voornamelijk op de code die omgaat met metingen, providers en trajecten. Dit omdat de databaseController instaat voor het opbouwen en het invullen van de objecten. Het is belangrijk om te vernoemen dat er in de testen gebruik wordt gemaakt van gemoekte data objecten. De testen zijn dus volledig onafhankelijk van de databank en maken gebruik van constante en voorgedefinieerde data om enkel het opbouwen van objecten te testen. Verder zal de code in verband met metingen ook

nauwgezet worden getest. Tot slot moet de code die het domein vormt ook getest worden. Het gaat hier concreet om de klassen Traject, Meting, Provider, Waypoint, Vertraging,...

DatabaseControllerTest	
dbController	DatabaseController
prepareClass()	void
haalProviderOpMetId_Correct()	void
haalProviderOpMetId_BestaatNiet()	void
haalProviderOpMetNaam_Correct()	void
haalProviderOpMetNaam_BestaatNiet()	void
haalActieveProviders_Correct()	void
haalAlleProviders_Correct()	void
haalAlleTrajecten_Correct()	void
haalAlleTrajectenMetWaypoints_Correct()	void
haalTrajectId_Correct()	void
haalTrajectNaam_Correct()	void
wijzigTraject_Correct()	void
wijzigTraject_MetNULL()	void

### 8.3 Integration testen

De integration testen controleren in tegenstelling tot de unit testen niet de individuele componenten van de software, maar de software als geheel. Om dit te bereiken wordt een lijst van te testen functionaliteiten overlopen. De tester moet deze hele lijst kunnen doorlopen zonder noemwaardige problemen om ervan uit te kunnen gaan dat de software als geheel ook correct werkt.

Voor elke test wordt ervanuit gegaan dat de gebruiker start op de homepage van de webapplicatie.

Bekijk reistijden

#### 8.3.1.1 Verwachte functionaliteit

De huidige reistijden worden op een overzichtelijke manier aan de gebruiker weergegeven.

#### 8.3.1.2 Stappen

- De gebruiker klikt in de navigatiebalk op Trajecten
- Er wordt naar een pagina genavigeerd met een kruistabel van alle trajecten en de reistijden
- In de tabel zijn per provider de optimale en huidige reistijd ingevuld. Ook de verwachte vertraging wordt per provider weergegeven.
- Bij het scrollen naar beneden blijft de titelkolom zichtbaar bovenaan het scherm.
- Een specifiek traject wordt geselecteerd en wordt vervolgens weergegeven op de kaart op dezelfde pagina.
- Meerdere trajecten kunnen ook geselecteerd worden.
- Opnieuw selecteren van een reeds aangeklikt traject deactiveert het traject en verwijdert enkel dit traject van de kaart.

Bekijk actuele verkeerssituatie

#### 8.3.1.3 Verwachte functionaliteit

Op een kaart en in een lijst wordt de huidige verkeerssituatie weergegeven, met zowel kleurcodering als concrete reistijden.

#### 8.3.1.4 *Stappen*

- Op de pagina is een selectiebalk met de verschillende providers en een algemeen overzicht aanwezig.
- Voor elk tabblad horende bij een provider is er een: algemene vertraging, drukste punt, kaart met alle trajecten aangeduid volgens drukte weergegeven op de kaart.

Vergelijk verkeersinformatie

#### 8.3.1.5 *Verwachte functionaliteit*

De gebruiker de verkeerssituaties van verschillende momenten en plaatsen vergelijken op een eenvoudige manier(side-by-side)

#### 8.3.1.6 *Stappen*

- De gebruiker selecteert een startdatum met de datepicker. Validatie van de invoer.
- De gebruiker selecteert een einddatum met de datepicker. Validatie van de invoer.
- De gebruiker kiest een traject van de dropdown lijst.
- Een grafiek wordt weergegeven met de gevraagde tijdsperiode van het geselecteerde traject.

Waypoints traject aanpassen

#### 8.3.1.7 *Verwachte functionaliteit*

Een gebruiker met adminprivileges kan het verloop van trajecten aanpassen door waypoints te verplaatsen, toe te voegen of te verwijderen.

#### 8.3.1.8 *Stappen*

- Gebruiker navigeert naar het adminpaneel
- De lijst met alle trajecten wordt weergegeven.
- Gebruiker selecteert een te wijzigen traject.
- Een kaart met het traject en alle waypoints wordt weergegeven. Ook wordt een dynamische lijst van belangrijke gegevens zoals lengte, reistijd, start- en eindpunt weergegeven.
- De gebruiker kan waypoints toevoegen door op het traject zelf te klikken.
- De gebruiker kan waypoints verwijderen door er dubbel op te klikken
- De gebruiker kan waypoints verplaatsen door te klikken en te verslepen.
- De gegevens in de tabel worden aangepast bij elke wijziging van de kaart
- Als op wijzig traject wordt gedrukt worden de aanpassingen weggeschreven naar de databank.

## 8.4 Usability testen

In dit onderdeel wordt de GUI (grafische user interface) getest. Het uittesten van de grafische interface gaat hand in hand met de use cases. Iedere use case beschrijft namelijk de interacties die een gebruiker met het systeem kan maken. Indien de gebruiker hier niet in slaagt (bijvoorbeeld door fouten) of heel wat hinder ondervindt om tot een resultaat te komen, dan kan de test als mislukt beschouwd worden. Het toevertrouwen van zulke testen aan externen, die niet noodzakelijk IT expert zijn kan een meerwaarde bieden aan de gebruiksvriendelijkheid en algemene kwaliteit van het project.

Use Case 1: Meld abnormaliteiten	p8
Use Case 2: Bekijk reistijden	p9
Use Case 3: Controleer data	p10
use Case 4: Bekijk actuele verkeerssituatie	p11
use Case 5: Vergelijk verkeersinformatie	p12
use Case 6: Data ophalen en hergebruiken	p13



Hier staan enkele scenario's beschreven die een testpersoon met vergelijkbare kennis van de uiteindelijke gebruiker moet doorlopen. De testpersoon volgt de instructies op, gaat niet vooruitlopen op de gevraagde acties en zegt duidelijk wat hij of zij denkt tijdens het navigeren van de GUI en uitvoeren van de taken. Bij moeilijkheden (>30 seconden zoeken) wordt hulp gevraagd en genoteerd wat voor problemen zorgt.

#### Bekijk reistijden

- Klik op trajecten op de navigatiebalk.
- Geef de lengte en reistijdinformatie van traject X.
- Geef dit traject weer op de kaart onderaan de pagina.
- Vergelijk dit traject met traject Y.
- Geef traject y weer op de kaart
- Verwijder traject X van de kaart.

##### 8.4.1.1 Opmerkingen

.....

.....

##### 8.4.1.2 Moeilijkheden

.....

.....

#### Bekijk actuele verkeerssituatie

- Bekijk de algemene verkeerssituatie op kaart
- Zoek de algemene vertraging en het drukste punt
- Herhaal deze stappen voor provider X

##### 8.4.1.3 Opmerkingen

.....

.....

##### 8.4.1.4 Moeilijkheden

.....

.....

#### Vergelijk verkeersinformatie

- Navigeer naar overzicht grafieken.
- Selecteer een startdatum en einddatum.
- Kies traject X
- Bekijk de grafiek specifieker over een interval van 1uur

##### 8.4.1.5 Opmerkingen

.....

.....

##### 8.4.1.6 Moeilijkheden

.....

.....

#### Waypoints traject aanpassen

- Open het administratorpaneel
- Kies voor wijzig traject X
- Voeg een nieuw waypoint toe.
- Verwijder waypoint Y

- Verplaats een waypoint.
- Geef die lengte van het aangepaste traject
- Geef de coördinaten van begin- en eindpunt.
- Sla de wijzigingen op.

#### 8.4.1.7 *Opmerkingen*

---

---

#### 8.4.1.8 *Moeilijkheden*

---

---

# Installatie en onderhoud

## 9 Installatie project op geconfigureerde omgeving

Het project vereist een linux distributie met daarop glassfish en mariadb of MySQL geconfigureerd. Voor details omtrend het opzetten hiervan wordt verwezen naar de uitgebreide installatiehandleiding in de appendix. Hier wordt enkel de installatie en configuratie van het verkeersplatform zelf behandeld.

Alle bestanden die nodig zijn voor configuratie zitten in de map deployment. Het script dat in deze map aanwezig is kan ook gebruikt worden om deze bestanden op de server binnen te halen. Alternatief kan het commando `"svn export https://github.ugent.be/iii-vop2016/verkeer-3/trunk/deployment --force --username [GIT_USERNAME] --password [GIT_WACHTWOORD]"` gebruikt worden om op de huidige locatie de folder deployment met de nodige bestanden te downloaden. Hiervoor is subversion nodig.

1. Configuratie van de databank
  1. Het bestand *"Volledige database.sql"* bevat de query om de databank op te bouwen en aan te vullen. Deze versie is compatibel met MySQL en MariaDB. Deze query kan op de geprefereerde manier uitgevoerd worden (terminal of gui zoals MySQL workbench).
2. Installatie van de webapplicatie
  1. De webapp kan geïnstalleerd worden door middel van het *"deployment.sh"* script. Op de server kan het commando `"bash deployment.sh [GIT_USERNAME] [GIT_WACHTWOORD]"` uitgevoerd worden om de laatste versie op te halen en te deployen op glassfish. (op de productieomgeving is dit script te vinden onder de root home directory /verkeerscentrum/repo/deployment/)
  2. Het script zal vervolgens de gebruikersnaam en wachtwoord van de glassfish admin in te geven.
  3. Bij de eerste installatie kan het script *"eerste deployment.sh"* analoog gebruikt worden.
  4. Handmatig kan het bestand *"verkeer3gui.war"* ook op een persoonlijk geprefereerde manier geïnstalleerd worden.
3. Configuratie van de properties
  1. Het script haalt ook een config.properties file op van de remote repository en kopieert deze naar een locatie om gebruikt te worden binnen glassfish *"/opt/glassfish4/glassfish/domains/domain1/config/config.properties"*. In dit bestand kunnen gegevens zoals de url van de databank, wachtwoord en gebruikersnaam en API keys aangepast worden indien nodig.

# Retrospective

## 10 Realisaties

Use Case 1: Meld abnormaliteiten	Niet geïmplementeerd	Geschrappt in eindproduct om tijd vrij te maken voor meer gewenste features.
Use Case 2: Bekijk reistijden	Volledig in orde	
Use Case 3: Controleer data	Volledig in orde	
Use Case 4: Bekijk actuele verkeerssituatie	Volledig in orde	
Use Case 5: Vergelijk verkeersinformatie	Grotendeels in orde	Het is mogelijk om verschillende trajecten te vergelijken over hetzelfde moment en tussen verschillende providers. Een gewenste uitbreiding is het vergelijken van verschillende tijdstippen.
Use Case 6: Data ophalen en hergebruiken	Niet geïmplementeerd	Licentie van de dataproviders liet niet toe om de gegevens te delen of openbaar te maken. Bijgevolg was een api dan ook van lagere prioriteit.
Use Case 7: Oorzaken verkeersproblemen bekijken	Grotendeels in orde	De basis voor het ophalen van oorzaken van verkeersproblemen is gelegd. De provider Here werd toegevoegd en is volledig functioneel. Andere providers die gelijkaardige informatie aanbieden kunnen toegevoegd worden op dezelfde wijze.

Hiernaast is er ook een pagina gemaakt die aangeeft welke gebeurtenissen zich hebben voorgedaan op de weg. Op deze manier kan een vertraging van enkele weken terug ook gelinked worden aan bijvoorbeeld een ongeval. Ook is er een administratiepaneel aanwezig dat de configuratie en het beheer van de voorgaande features verzamelt.

## 11 Mogelijke verbeteringen

Op vlak van efficiëntie kan de code die in staat voor de communicatie met de databank een stuk beter. Zo worden er bijvoorbeeld nog op veel plaatsen metingen stuk per stuk weggeschreven, in plaats van dit in batches te doen.

Het systeem van gebeurtenissen op de wegen kan indien gewenst uitgebreid worden om automatisch gelinked te worden aan vertragingen in de buurt.

In de gui zijn er nog verschillende visuele bugs aanwezig zoals de kaarten of trajecten die soms niet laden. Dit komt af en toe voor maar er is nog geen oplossing voor gevonden.

## 12 Realisaties per persoon

Hieronder volgt een opsomming per persoon wat deze heeft bijgedragen aan het project. Sommige realisaties kunnen bij meerdere personen voorkomen indien deze samengewerkt hebben om dit onderdeel uit te werken.

### 12.1.1.1 *De Bock Jelle*

- Structuur van de domein code opstellen (klassenschema + eerste versie van de code)
- Opzetten van de structuur van de databank
- Opzetten van de structuur van de webapplicatie
- Scraper van Here uitwerken
- Implementatie van Leaflet maps gestuurd door Openstreet maps
- Highcharts grafieken in de webapplicatie implementeren
- Het design van de webapplicatie
- Filmpjes maken van de basishandelingen in de webapplicatie
- Artikels op de wikipagina maken
- Nalezen van het analysedocument (spellingsfouten)
- Eindpresentatie (slides maken en presenteren)

### 12.1.1.2 *Floré Brent*

- Softwareanalyse: document opstellen en uitwerken
- Opzetten scheduling
- Logging
- Scraper voor provider Coyote uitwerken in java
- Dal refactoreren en optimaliseren
- Aanpassen van bestaande trajecten
- Documentatie
- Dependency injection en unit testen
- Opstellen integration testen en usability testen
- Deployen van project
- Deployment script schrijven
- Presentatie 2
- Debuggen en verbeteringen van front end

### 12.1.1.3 *Stofferis Jeroen*

- analysedocument van de verschillende providers,
- TomTom route scraper,
- presentatie 1,
- perlscraper Waze,
- documentatie: installatie Fedora & CentOS server,
- documentatie: MariaDB,
- documentatie: manuele deployment,
- volledig userbeheer + loginmodule,
- abstracte scraper backend voor verkeersproblemen,
- Here provider voor verkeersproblemen.

### 12.1.1.4 *Vandemoortele Simon*

- Initiele logging functionaliteit
- Analysedocument van de verschillende providers
- Documentatie: providers, trajecten, etc.

- Controle van alle waypoints per provider (zorgen dat iedere provider de correcte waypoints meegeeft en dus de juiste route-informatie berekent)
- Javadoc
- Gebruik maken van een end-time bij de traffic-incidents
- Compatibiliteitsproblemen tussen MySQL en MariaDB oplossen
- Generatoren
- Queue-systeem in de ScheduleController
- Mogelijkheid om meerdere optimale reistijden per provider te hebben
- Databankoptimalisaties waar mogelijk
- GUI:
  - De trajecten-tabel
  - Het admin-paneel
  - Index: basislogica om het laten werken met afzonderlijke providers
  - Bugfixes (status, index, trajecten, admin, incidents, login)
- Bugfixes in de statistiekenqueries
- Performantieverbeteringen (Controllers, Scrapers, Repositories, views, js-files, ...)
- Presentatie 2 en eindpresentatie

#### 12.1.1.5 Vervenne Jan

- JSONScraper om API call te maken
- Google Scraper uitwerken
- Presentatie eerste sprint
- Waze scraper in perl:
  - Database connectie
  - Selenium server
  - Web Scraping
  - Script vanuit java
- Waypoints gebruiken bij HereScraper en updaten TomtomScraper
- Documentatie: Sequentiediagrammen, Klassediagrammen,...
- GUI:
  - Sorteren tabellen
  - Pagina trajecten vergelijken
  - Pagina twee periodes traject vergelijken



# Appendix

## 13 Configuratie van server en installatie project

Dit onderdeel van de software-analyse zal de installatie van de (productie)server grondig beschrijven. In wat volgt wordt in detail besproken hoe vanuit de broncode een werkende applicatie opgestart kan worden.

### 13.1 Installatie

Om het project te laten runnen zijn er enkele basiscomponenten nodig. Er zijn minimaal twee toestellen nodig om dit project te laten werken. Allereerst is er een server nodig. Dit kan zowel een fysieke hardware server, een desktop als een virtuele machine zijn. Om deze server zal een server OS draaien gebouwd rondom de Linux kernel. Een tweede component is een host toestel, dat bij voorkeur van een grafisch besturingssysteem gebruik maakt (Windows, Mac OS, Linux). Dit toestel zal eenmalig gebruikt worden om de broncode te compileren. Eenmaal dit gebeurt is kan het gecompileerde project overgeplaatst worden naar de server.

#### Server

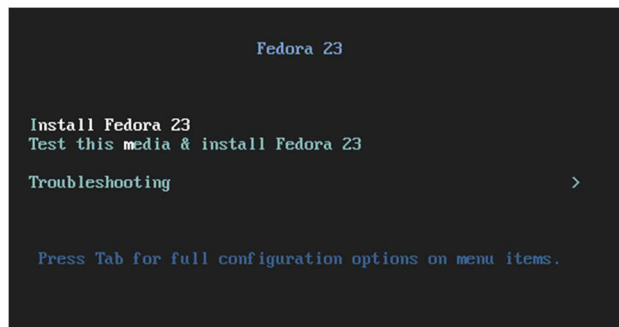
Vooraleer de installatie van start kan gaan moet een geschikte keuze gemaakt worden welk besturingssysteem er op de server zal draaien. De keuze is enerzijds afhankelijk van persoonlijke voorkeur, maar ook functionaliteit en compatibiliteit verschillen vaak van distributie tot distributie. Zoals reeds eerder aangehaald wordt er voor de serveromgeving gekozen voor een Linux georiënteerde distributie. Dit OS is bij voorkeur specifiek voor serverdoeleinden samengesteld. In deze installatiehandleiding zullen twee distributies in detail besproken worden: enerzijds Fedora en anderzijds CentOS. De reden hiervoor is dat een Fedora server aangeboden werd door de opleidingscoördinatoren van onze richting. Fedora is zeker geschikt als server maar deze heeft echter het nadeel van een snelle Life-Cycle. Dit wil zeggen dat een versie van Fedora sneller beschouwd wordt als 'oud' en deze hierdoor geen updates of beveiligingspatches meer ontvangt. In het algemeen wordt een bepaalde versie ondersteund tot één maand nadat twee nieuwe versies uitgekomen zijn. Er wordt ongeveer ieder half jaar een nieuwe versie gereleased. CentOS daarentegen heeft een veel langere 'life cycle'. De huidige versie (CentOS 7) werd gereleased in juni 2015 en zal ondersteund worden tot juni 2024.

Stabiliteit is eveneens een belangrijk aspect in dit project. Het is de bedoeling dat deze server (en het project) 24 uur per dag, 7 dagen per week draait. Hoe stabiel het besturingssysteem, hoe groter de kans dat dit vooropgesteld doel gehaald kan worden. Zowel CentOS als Fedora staan bekend om hun stabiliteit, maar dit is voor heel wat Linux distributies het geval. Toch zorgen de frequente releases bij Fedora ervoor dat het OS nipt de duimen moet leggen voor CentOS op het vlak van stabiliteit. Als een CentOS een label van 100% stabiel krijgt, dan komt dit neer op 99.5% stabiliteit bij Fedora. Uiteindelijk zijn beide server geschikt als server en ligt de finale keuze bij de eindgebruiker.

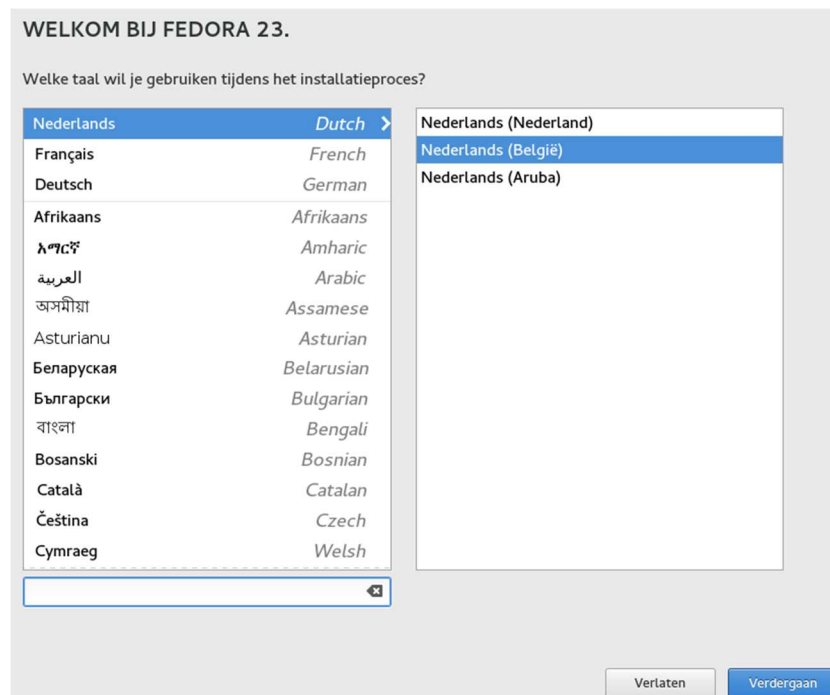
#### 13.1.1.1 Fedora

De installatie van Fedora is relatief eenvoudig. Net zoals bij Windows kan de installatie gebeuren op een grafische manier. Het besturingssysteem kan gedownload worden via de website <https://getfedora.com>. Hier is hierbij wel belangrijk om de serverversie te selecteren. De keuze voor 32-bit of een 64-bit systeem is afhankelijk van de fysieke server. Een algemene regel is om de 32-bit installatie uit te voeren op een systeem met 4GB ram of minder. Indien de CPU van het toestel 64-bit ondersteunt, wordt best gekozen voor de 64-bit installatie.

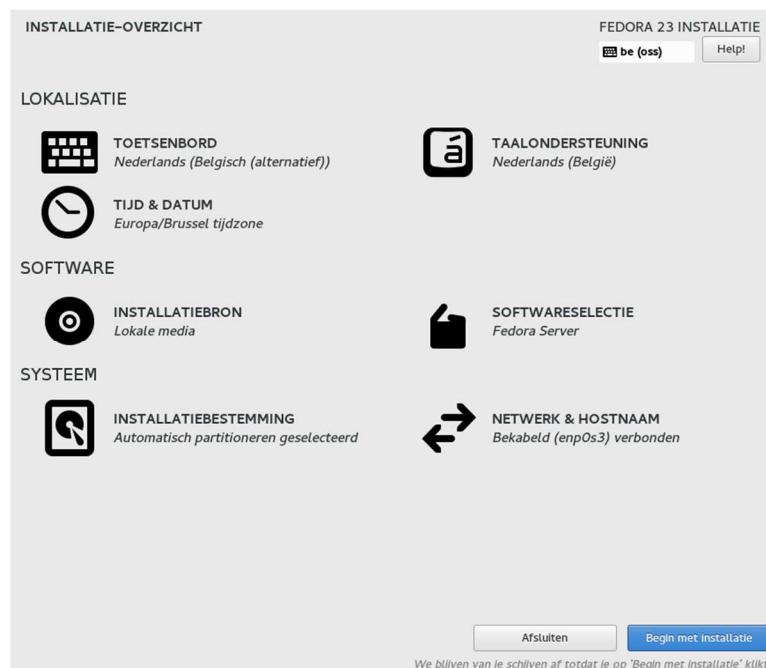
Wanneer het toestel opgestart wordt met de installatie-cd van Fedora wordt gevraagd aan de gebruiker of hij het OS meteen wil installeren, of dat hij het OS eerst even wil testen. Alhoewel het testen van Fedora handig kan zijn wordt hier toch voor de directe installatie gekozen. Klik hier dus op 'install'.



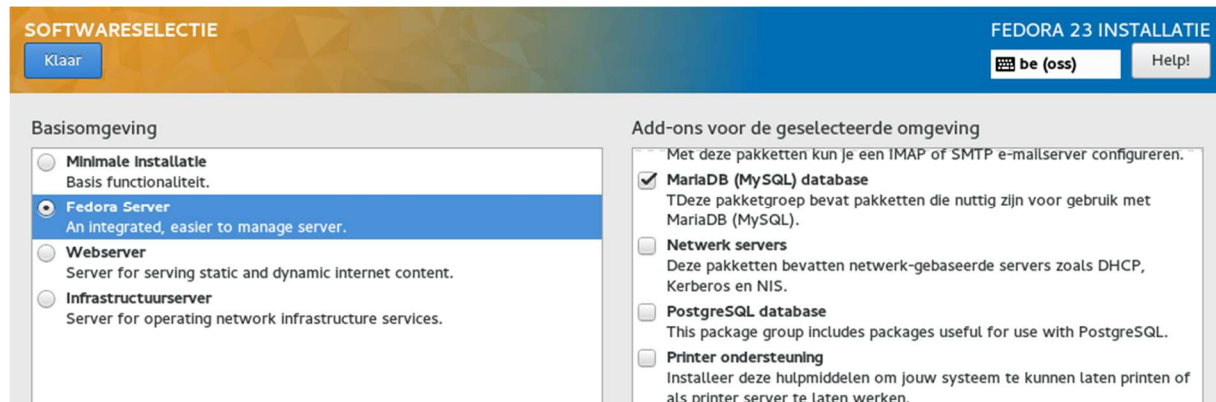
De eerste stap is een taal selecteren. Indien er internettoegang is, zal Fedora automatisch Nederlands selecteren. Klik vervolgens op “verdergaan”.



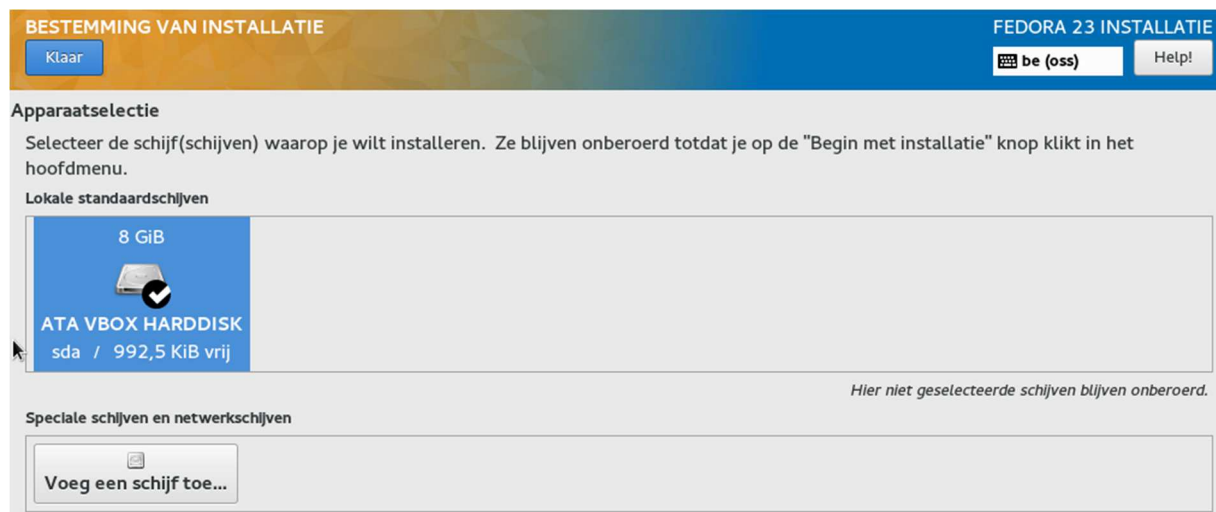
Vervolgens wordt een overzicht gegeven van de te installeren onderdelen. Hier is het belangrijk om onder softwareselectie en installatiebestemming volgende wijzigingen aan te brengen.



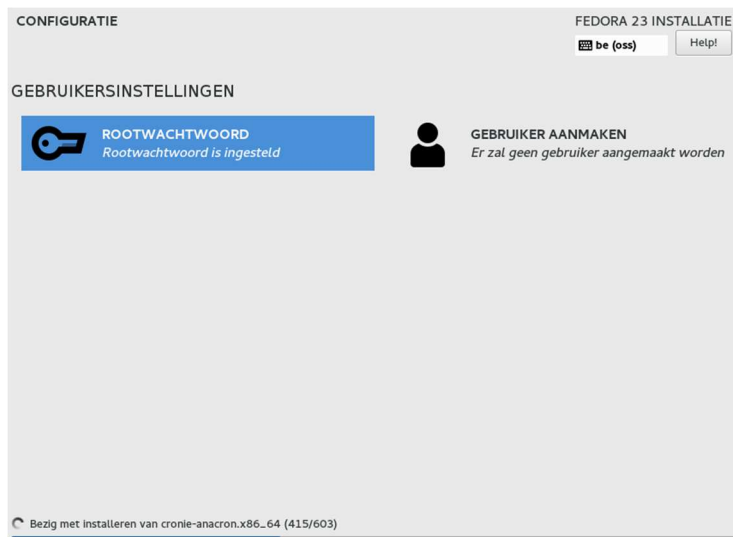
Onder softwareselectie moet het vinkje gemarkeerd worden naast MariaDB (MySQL) database. Deze database is nodig om de gegevens op te slaan die het project genereert. Indien gekozen wordt voor een ander type database (zoals MySQL), zal deze op eigen houtje geïnstalleerd moeten worden. Vervolgens klikt men linksboven op “klaar”.



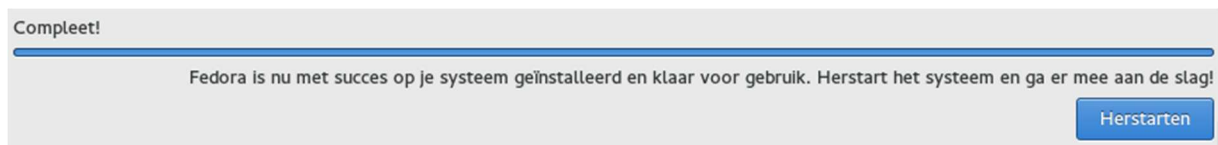
Nu men terug op het algemene overzichtsscherm zit, moet de installatiebestemming geselecteerd worden. Dit wordt bekomen door op het harde schijficoontje te drukken en vervolgens de harde schijf te selecteren. In onderstaande screenshot staan een voorbeeld van dit menu met de selectie van de harde schijf. Vervolgens klikt men op “klaar”.



In het overzichtsmenu wordt nu onderaan geklikt op “begin met installatie”. Dit zal de installatie starten. Tijdens de installatie wordt de mogelijkheid geboden om het wachtwoord van de root gebruiker in te stellen. Een nieuwe gebruiker aanmaken is niet direct nodig.



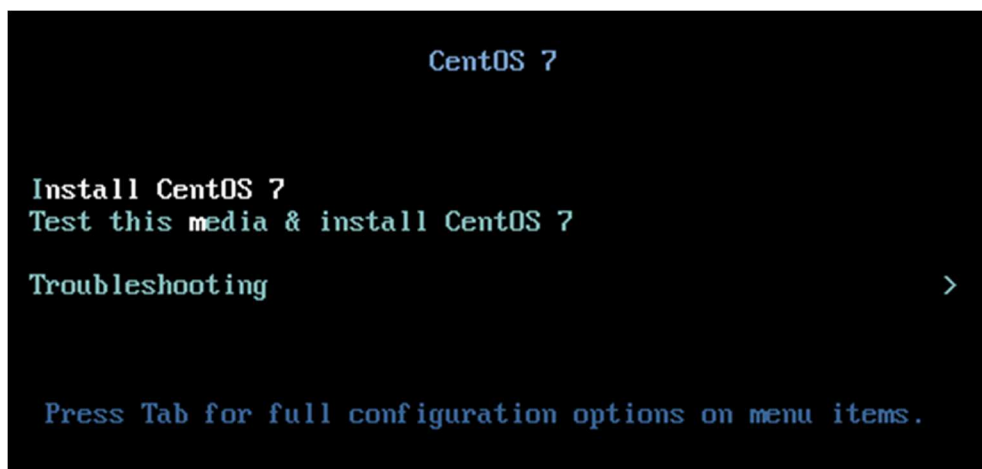
Wanneer het installatieproces voltooid is, is het tot slot nog belangrijk om de cd te verwijderen alvorens opnieuw op te starten. Klik vervolgens op “herstarten”.



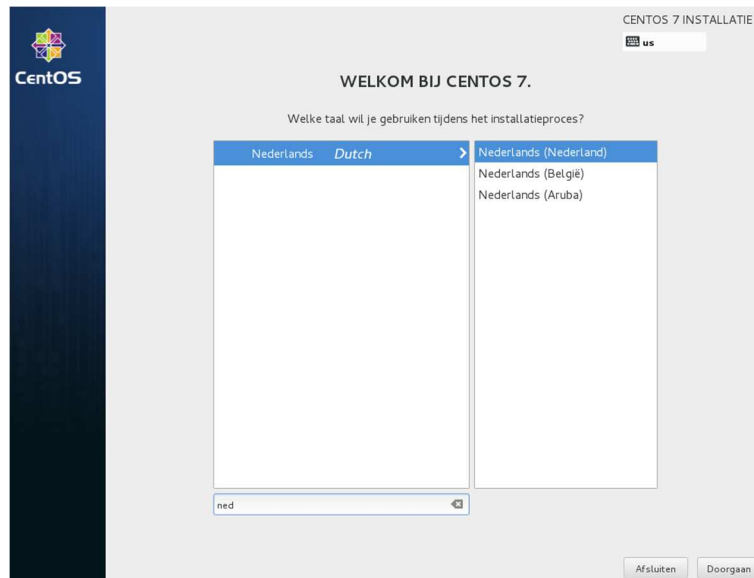
#### 13.1.1.2 CentOS

Het installeren van CentOS is zeer gelijkaardig met de installatie van Fedora, dit omdat beide besturingssystemen vanuit de populaire Red Hat distributie ontsproten zijn. Op het moment van schrijven is de meest recente, stabiele versie CentOS 7. Deze is te downloaden via <https://centos.org>. Hier moet de keuze gemaakt worden tussen de “minimal ISO”, “de dvd ISO” en de “full ISO”. De “dvd ISO” is het meest geschikt voor dit project maar indien de beheerder van dit toestel verkiest om zelf de packages te installeren die vereist zijn, is de “minimal ISO” voldoende. De keuze voor 32- of 64-bit vergt hier minder aandacht omdat tijdens de installatie automatisch een controle uitgevoerd wordt om welk type toestel het gaat. Afhankelijk van de processor wordt de juiste versie geïnstalleerd.

Wanneer de ISO opstart wordt gevraagd of de gebruiker de software meteen wil installeren of eerst even in demo wil runnen. We kiezen hier voor “install CentOS 7”.

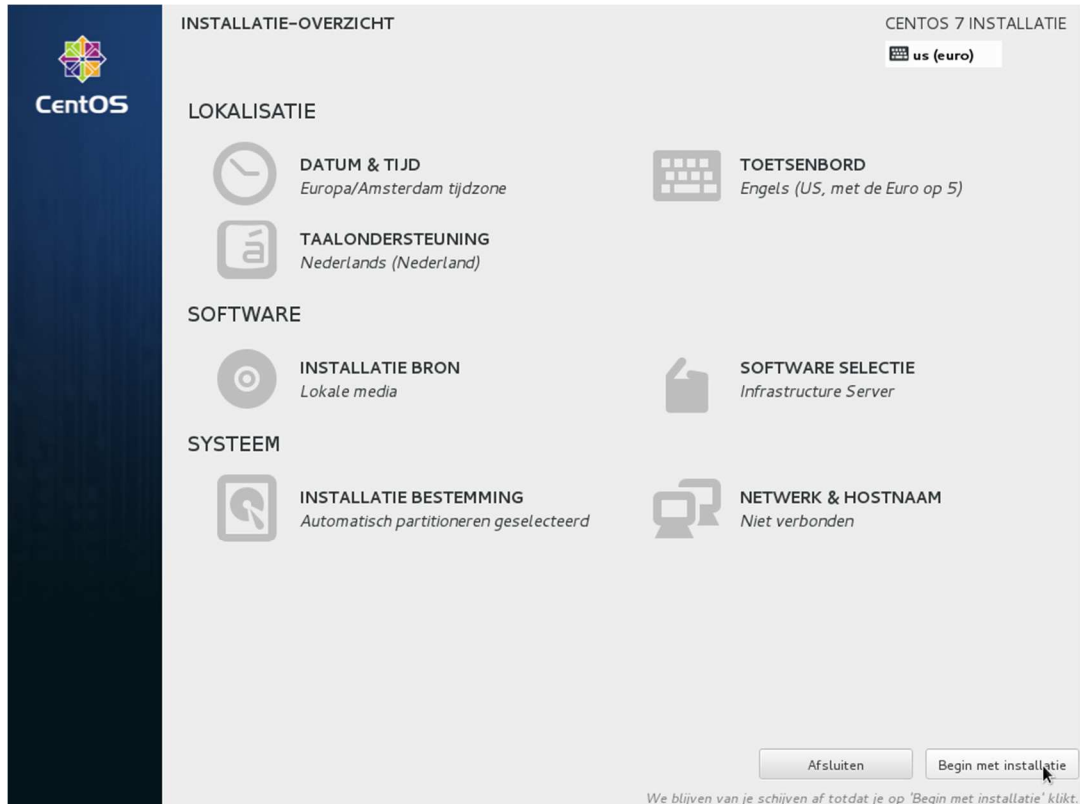


Het vervolg van de installatie zal verder op een grafische manier verlopen. Eerst moet een taal geselecteerd worden waarin het installatieproces zal verlopen. Na de selectie van de taal klikt men rechtsonder op “doorgaan”.

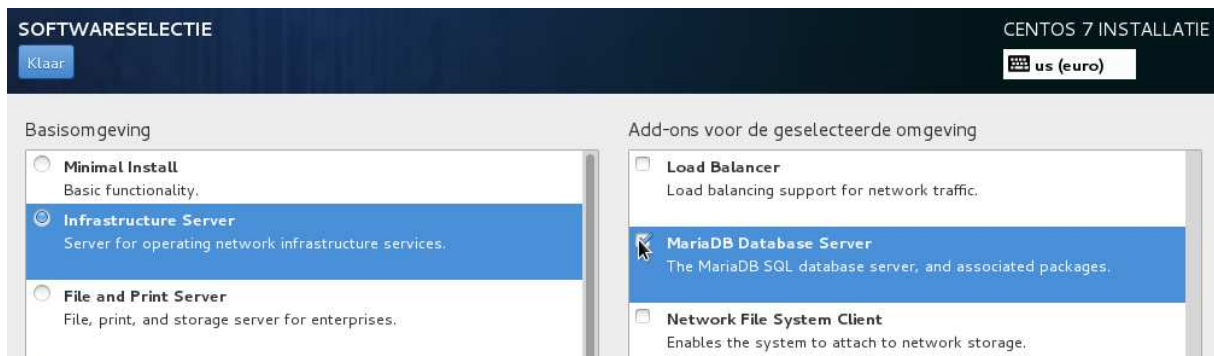


Het volgende scherm biedt een overzicht aan van alle installatieopties. Onder het submenu “lokalisatie” staan heel wat algemene instellingen rondom toetsenbord, taal, datum & tijd. Deze instellingen worden vaak via het netwerk correct ingesteld. De kans is groot dat de juiste opties al geselecteerd zijn.

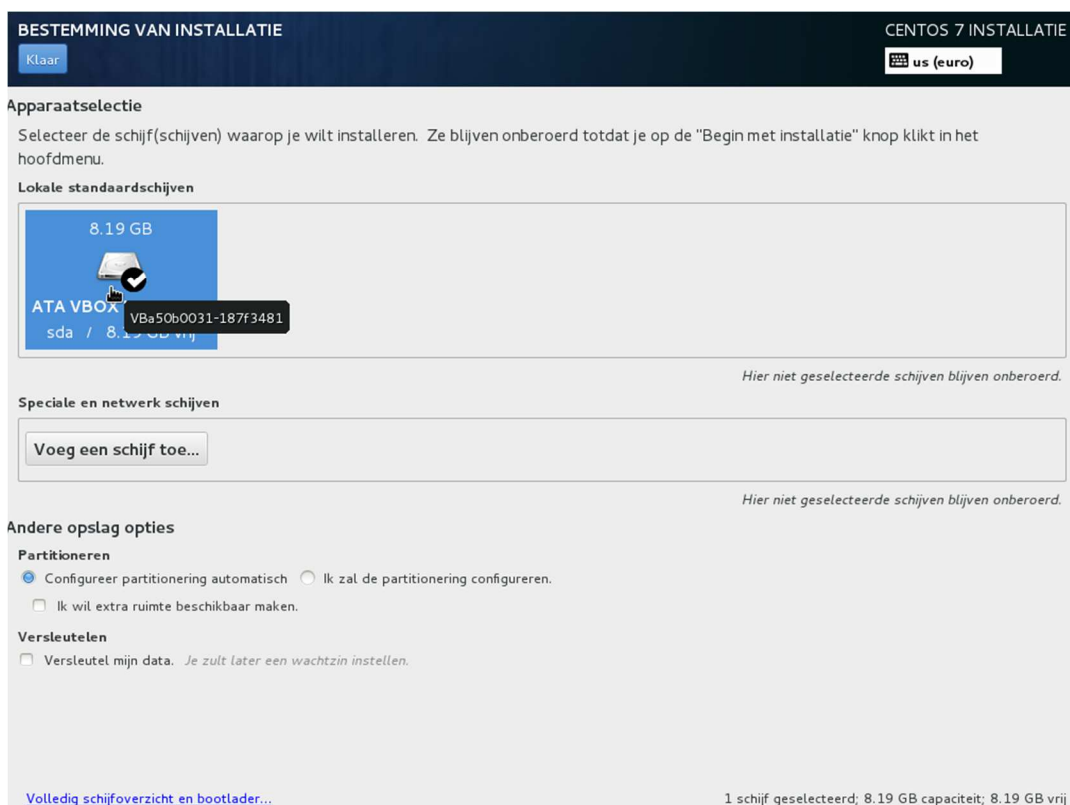
Onder het submenu software moeten wel enkele wijzigingen gemaakt worden. Klik hier op “software selectie”.



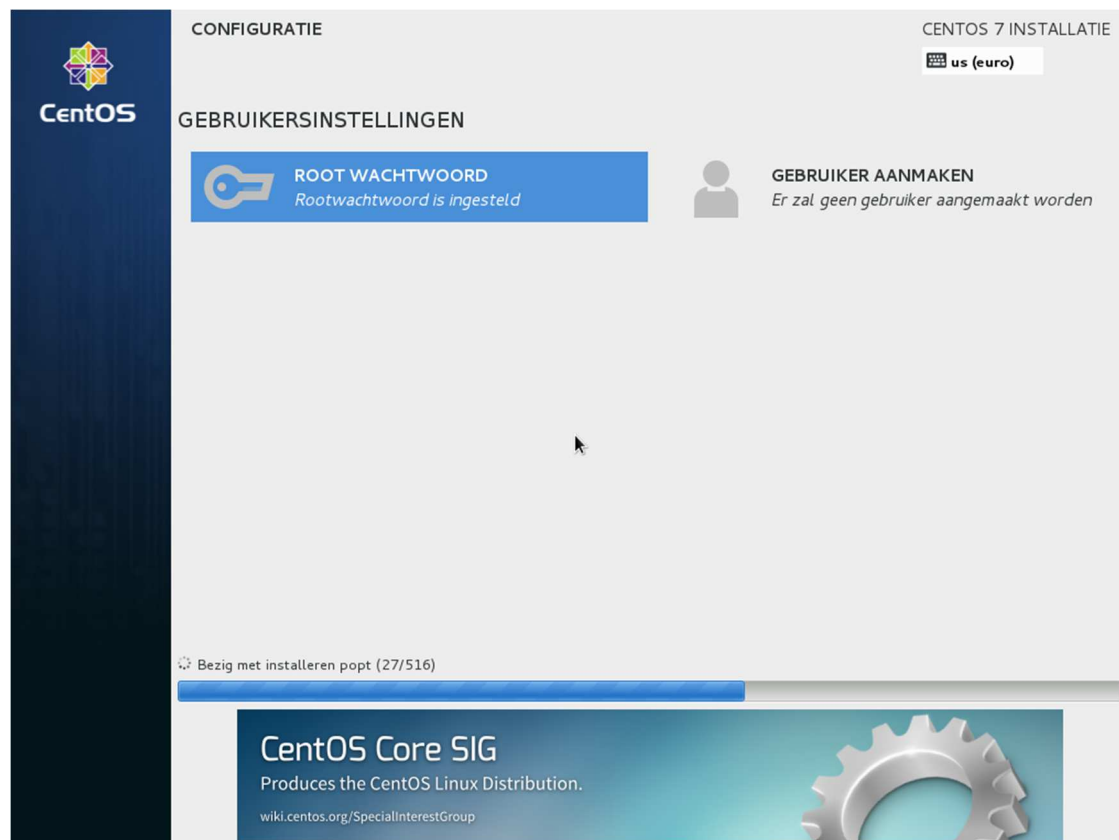
Standaard wordt een minimale installatie voorgesteld, deze kan gewijzigd worden naar een “infrastructure server” met de add-ons optie MariaDB Database Server in de rechter kolom. MariaDB is een databank die gebruikt wordt om de gegevens te bewaren dat het project verzamelt. Indien er gebruik gemaakt wordt van een externe database (bijvoorbeeld op een ander toestel) of een ander type database, moet deze optie niet geselecteerd worden. Vervolgens klikt men linksboven op “klaar”.



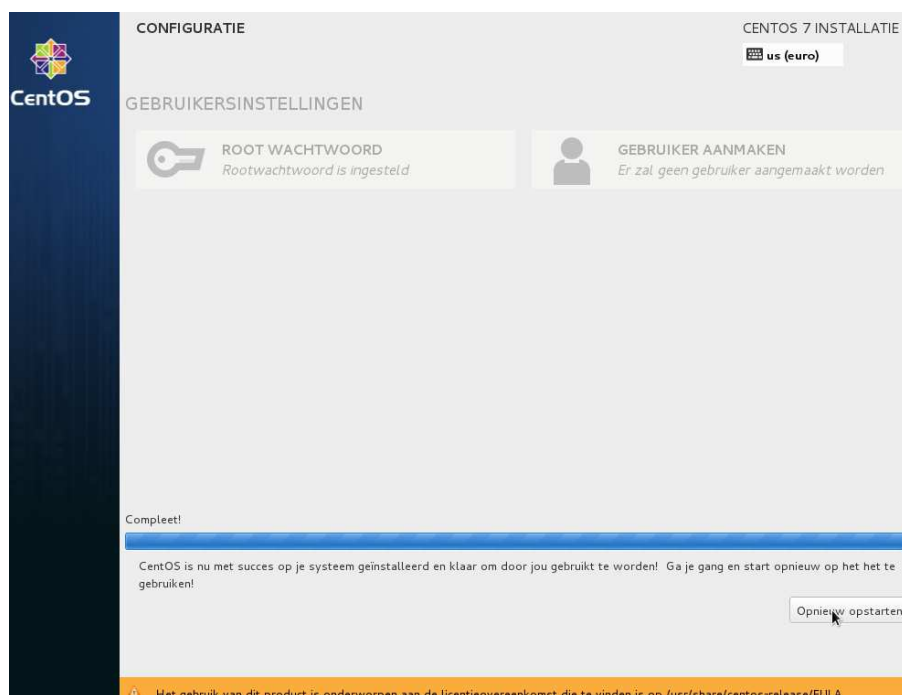
Terug bij het overzichtsmenu gekomen staat een waarschuwing bij bestemming van de installatie onder het submenu systeem. Wanneer met hier op klikt hoeft enkel de harde schijf geselecteerd te worden. Voor het aanmaken en configureren van partities, opstartprocedures,... zal de installatie wizard zelf zorgen. Vervolgens klikt met op “klaar”.



Terug bij het overzicht aangekomen wordt onderaan geklikt op “begin installatie”. Tijdens de installatie is het handig om het root wachtwoord in te stellen. Dit is vergelijkbaar met het administrator wachtwoord in Windows. Een extra gebruiker aanmaken is momenteel niet nodig.



Wanneer de installatie ten einde is, is het belangrijk om eerst de cd uit te werpen vooraleer op “Opnieuw opstarten” te klikken. Dit vermijdt dat de server opnieuw opstart vanaf de installatie cd.





## GlassFish

Het eerste softwarepakket dat geïnstalleerd moet worden is GlassFish. GlassFish is een applicatieserver die de Java webapplicatie aan de gebruikers zal presenteren.

Vooraleer GlassFish geïnstalleerd kan worden, is het noodzakelijk om de Java EE SDK te installeren.

Java is immer de basis laag van de webserver en het is de programmeertaal waarin de applicatie geschreven is.

Om een correct werkende Glassfish server te verkrijgen dienen volgende acties uitgevoerd te worden:

- programma's *wget* en *unzip* installeren
- gebruiker toevoegen met username *glassfish*
- het wachtwoord van de *glassfish* user wijzigen
- java jdk 8.45 downloaden
- java jdk 8.45 installeren

```
yum install wget unzip nano

adduser --comment 'Glassfish User' --home-dir /home/glassfish glassfish

passwd glassfish

wget --no-cookies --no-check-certificate
--header "Cookie: oraclelicense=accept-securebackup-cookie"
"http://download.oracle.com/otn-pub/java/jdk/8u45-b14/jdk-8u45-linux-
x64.rpm" -O jdk-8u45-linux-x64.rpm

yum install jdk-8u45-linux-x64.rpm
```

Nu de Java JDK geïnstalleerd is, kan GlassFish toegevoegd worden aan het systeem. Onderstaande commando's voeren de volgende zaken uit:

- inloggen met de user *glassfish*
- Glassfish 4.1 downloaden en uitpakken
- het gedownloade bestand wissen

```
su glassfish
```

```
wget --no-cookies --no-check-certificate
```

```
--header "Cookie: oraclelicense=accept-securebackup-cookie"
```

```
"http://download.oracle.com/glassfish/4.1.1/release/glassfish-4.1.1.zip" -O  
glassfish-4.1.1.zip
```

```
unzip glassfish-4.1.1.zip
```

```
rm -f glassfish-4.1.1.zip
```

```
exit
```

Nu GlassFish succesvol geïnstalleerd werd op de server blijft nog één stap over: het toevoegen van een service om GlassFish te starten. Het aanmaken van deze service komt neer op het aanmaken van een bestand met configuratieparameters. Onderstaand commando maakt dit bestand aan en opent het vervolgens.

```
nano /etc/systemd/system/glassfish.service
```

Nu het bestand geopend is, moet onderstaande configuratie toegevoegd worden.

```
[Unit]
```

```
Description = GlassFish Server v4.1.1
```

```
After = syslog.target network.target
```

```
[Service]
```

```
User=glassfish
```

```
ExecStart = /usr/bin/java -jar
```

```
/home/glassfish/glassfish4/glassfish/lib/client/appserver-cli.jar start-  
domain
```

```
ExecStop = /usr/bin/java -jar
```

```
/home/glassfish/glassfish4/glassfish/lib/client/appserver-cli.jar stop-  
domain
```

```
ExecReload = /usr/bin/java -jar
```

```
/home/glassfish/glassfish4/glassfish/lib/client/appserver-cli.jar restart-  
domain
```

```
Type = forking
```

```
[Install]
```

```
WantedBy = multi-user.target
```

Vervolgens drukt men de toetsencombinatie CTRL+O in om het bestand op te slaan.

Het is zeer handig om de Glassfish service automatisch te laten starten tijdens het opstarten van het besturingssysteem. Het starten van de service en deze activeren wanneer het systeem opstart kan met onderstaande commando's.

```
systemctl enable glassfish.service
systemctl start glassfish.service
```

Tot slot is het natuurlijk aangewezen dat deze service vlot bereikt kan worden vanaf een extern toestel. Daarom is het noodzakelijk om enkele poorten te openen. Standaard worden volgende poorten door GlassFish gebruikt:

- Administratie: 4848
- HTTP: 8080
- HTTPS: 8181

Poorten openen kan met onderstaande commando's:

```
firewall-cmd --zone=public --add-port=4848/tcp --permanent
firewall-cmd --zone=public --add-port=8080/tcp --permanent
firewall-cmd --zone=public --add-port=8181/tcp --permanent
systemctl restart firewalld
```

#### 13.1.1.3 MariaDB

Wanneer tijdens installatie van het OS niet gekozen werd voor automatische installatie van MariaDB wordt hier nog kort gedocumenteerd hoe MariaDB in enkele stappen geïnstalleerd kan worden. Het opzetten van een database is immers essentieel voor dit project. Zoals eerder aangehaald is het perfect mogelijk om een alternatieve database te gebruiken zoals MySQL of MSSQL. De installatie kan zowel gebeuren op het lokale toestel als op een externe server. Er moet sowieso een externe server gebruikt worden indien de keuze voor MSSQL gemaakt werd omdat deze enkel functioneert op een Windows server. Onderstaande commando's installeren MariaDB en voeren vervolgens een beveiligde installatie uit.

```
yum install mariadb-server
mysql_secure_installation
```

Vervolgens wordt de database aangemaakt, samen met een MariaDB user waarmee de applicatie zal verbinden met de databank. Tot slot wordt het databaseschema, dat de tabellen van de webapplicatie bevat, geïnstalleerd.

```
mysql -u root -p
create database vop;
grant all privileges on vop.* to vop@localhost identified by
'secretpassword';
flush privileges;
exit
mysql -u root -p password <./VolledigeDatabank.sql
```

*Opmerking:* Indien u merkt dat metingen wel gescraped worden maar dat deze niet op de GUI tevoorschijn komen, kan het zijn omdat de interne klok van MariaDB verkeerd staat. Het herstarten van deze service zou dit probleem moeten oplossen.

Tot slot moet de MariaDB service gestart worden en wordt ingesteld dat deze opstart samen met het systeem.

```
systemctl enable mariadb
systemctl start mariadb
```

#### 13.1.1.4 Vast IP-adres

Het toekennen van een vast IP-adres is belangrijk zodat het mogelijk is om steeds de gebruikersinterface te kunnen bereiken via hetzelfde IP-adres. Indien er gewerkt wordt met een DHCP server die at random een IP-adres toekent aan de server, dan is de kans groot dat bij het herstarten van dit toestel, er een ander IP-adres toegekend wordt. Voor dit probleem zijn twee mogelijke oplossingen: ofwel stelt men in op de DHCP server dat het toestel met het MAC-adres van de applicatieserver gekoppeld wordt aan een (vast) IP-adres, ofwel stelt men lokaal in op de server dat deze zichzelf een vast IP-adres toekent. Beide oplossingen worden hieronder beschreven.

##### 13.1.1.4.1 Statisch (lokaal)

Linux kent net zoals dat in Windows het geval is een naam toe aan iedere netwerkadapter. Om wijzigingen te kunnen aanbrengen moet deze naam gekend zijn. Standaard wordt aan fysieke toestellen de naam eth0 toegekend, maar dit verschilt van distributie tot distributie. Om de naam te achterhalen moet het eerste commando uitgevoerd worden. Daarna kan het configuratiebestand aangepast worden van die netwerkadapter. Daarbij is het belangrijk om de naam in te vullen die teruggegeven wordt bij het eerste commando.

```
ifconfig
nano /etc/sysconfig/network-scripts/ifcfg-eth0
```

Hieronder staat een voorbeeld van beide commando's:

```
[root@new-host-4 ~]# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.6 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::a00:27ff:fe11:9a70 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:11:9a:70 txqueuelen 1000 (Ethernet)
    RX packets 471 bytes 52079 (50.8 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 328 bytes 54864 (53.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 0 (Local Loopback)
    RX packets 268 bytes 21716 (21.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 268 bytes 21716 (21.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@new-host-4 ~]# nano /etc/sysconfig/network-scripts/ifcfg-enp0s3
[root@new-host-4 ~]#
```

Nu het configuratiebestand geopend is, moeten onderstaande regels aangepast of toegevoegd worden:

```
BOOTPROTO=none
ONBOOT=yes
NETWORK=192.168.1.1
NETMASK=255.255.255.0
IPADDR=192.168.1.10
USERCTL=no
```

#### 13.1.1.4.2 Dynamisch

Het instellen van een DHCP-server zal hier niet in detail beschreven worden aangezien er erg veel types, versies en varianten bestaan. Zowel Linux als Windows kunnen een DHCP-server draaien, routers kunnen instaan als DHCP-server, ... Omdat het uitschrijven van de configuratie voor iedere variant onmogelijk is, worden enkel algemene instructies gegeven die overal van toepassing zijn.

Om een toestel te kunnen linken aan een IP-adres in de DHCP-server, heeft deze server het MAC-adres nodig van het toestel. Het MAC-adres kan gevonden worden via onderstaand commando:

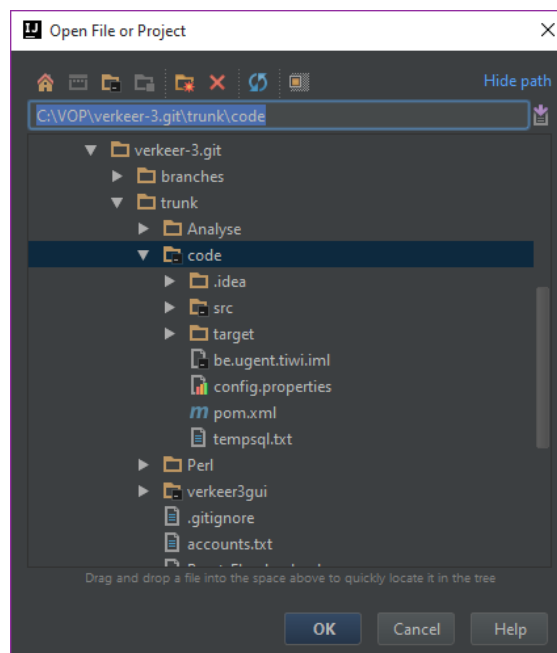
```
ifconfig -a
```

Bij de netwerkadapter (enkele vaak voorkomende namen zijn eth0, eth1, enp0s1, ...) staat een regel met als titel HWaddr, hiernaast staat het MAC-adres vermeld. Bijvoorbeeld: 00:08:C7:1B:8C:02. Dit moet vervolgens in de DHCP-server toegevoegd worden, samen met het gekoppelde IP-adres.

## Desktop

Op de desktop dient eender welke versie van Windows aanwezig te zijn. De desktop is nodig om de broncode van het project om te zetten in een uitvoerbaar programma, wat vervolgens op de server geplaatst kan worden. GlassFish zal dit uitvoerbaar bestand vervolgens voor geïnteresseerde gebruikers uitvoeren. De ontwikkeling van de software gebeurt grotendeels in de IntelliJ IDEA. Dit is een programmeeromgeving, vergelijkbaar met Netbeans en Eclipse. Het is dan ook perfect mogelijk om vertrekkend van de broncode, het volledig project te compileren in de vooraf genoemde programmeeromgevingen. Het is zelfs mogelijk om het project in de command prompt te compileren, maar dit zou de toegankelijkheid van deze handleiding drastisch verlagen. IntelliJ kan gedownload worden via de officiële site van de ontwikkelaars: <http://www.jetbrains.com/idea/>.

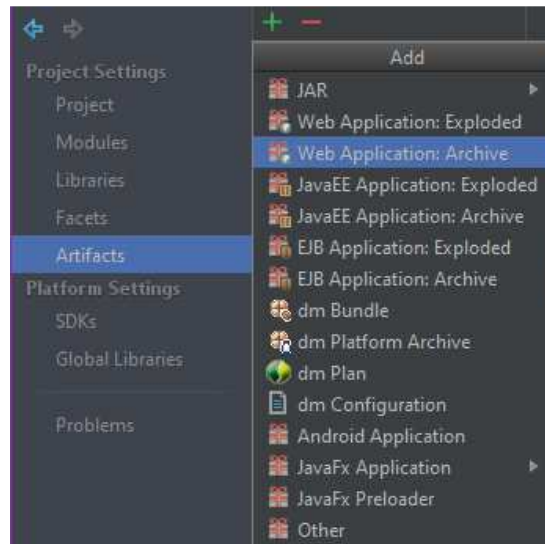
Na de installatie dient het project geïmporteerd te worden. Dit kan via de knoppen **file** → **open**. Vervolgens moet genavigeerd worden naar de projectmap en wordt de map code geselecteerd. Vervolgens klikt men op ok. IntelliJ zal nu de broncode en instellingen laden, dit kan even duren.



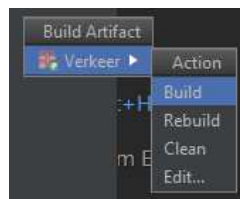
## Deployment

### 13.1.1.5 De JAVA libraries

Om de applicatie te runnen op de server vereist GlassFish dat er een WAR-file (Java Web Application) aangemaakt wordt. Dit kan bereikt worden via **file** → **project structure** → **artifacts**. Vervolgens klikt met op het groene + icoontje en wordt “Web Application: Archive” geselecteerd. Klik vervolgens op ok.

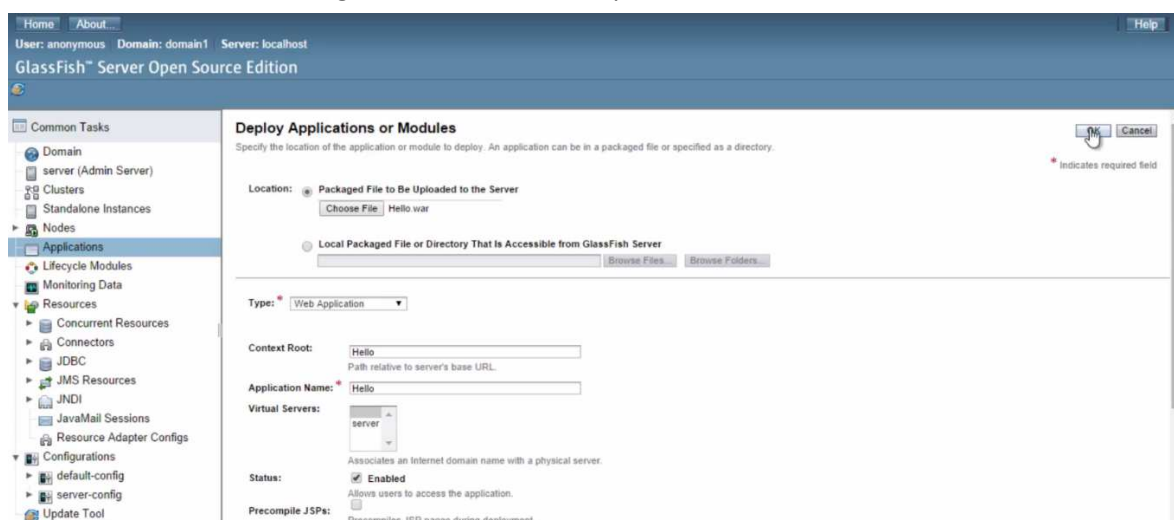


Nu IntelliJ weet wat er in de WAR file geplaatst moet worden, kan de opdracht gegeven worden om dit bestand ook effectief aan te maken. Dit wordt gedaan **via build** → **build artifacts** → **action** → **build**



Het WAR bestand is nu aangemaakt onder \code\classes\artifacts\Verkeer

De volgende stap is om dit bestand te deployen op de server. Dit kan bekomen worden door te surfen naar het administratorpaneel van Glassfish. Deze webpagina is actief op poort 4848 van de server bv: 192.168.1.10:4848. Wanneer men inlogt op de webpagina is er aan de linker kant een overzicht met allerlei opties. Om te deployen, druk op “applications” en vervolgens op “choose file”. Selecteer nu het WAR bestand dat eerder aangemaakt werd en klik op ok.



#### 13.1.1.6 Configuratie Glassfish

Na het uploaden van de war-file, zal het nodig zijn om de correcte configuratie in te stellen voor de server. De service zal bijvoorbeeld moeten weten welke API-keys het moet gebruiken en welke authenticatie nodig is voor de databank. Hiervoor gebruiken we het “config.properties”-bestand in de map “code”.

```
//<domain> is meestal "domain1"
cd /opt/glassfish4/glassfish/domains/<domain>/config
nano config.properties
//Plak hier de data van de config.properties file in, sluit met ctrl+X
```

Ook zal het nodig zijn om de MariaDB JDBC Connector in de classpath van Glassfish te plaatsen.

Dit gebeurt met de volgende commando's

```
cd /opt/glassfish4/glassfish/domains/<domain>/lib/ext
wget http://central.maven.org/maven2/org/mariadb/jdbc/mariadb-java-
client/1.4.2/mariadb-java-client-1.4.2.jar
//Indien er een nieuwe versie is, kan deze uiteraard ook gebruikt worden
```



## 14 Analyse providers en reistijden

### 14.1 Inleiding

De bedoeling van dit document is om, in opdracht van het mobiliteitsbedrijf van Gent, informatie te verzamelen over de gebruikte providers en trajecten in het Vakoverschrijdend project van Projectgroep 3. Dit document functioneert als een extra hulpmiddel om uiteindelijk een finale keuze te maken uit één van de providers.

Eerst komt het hoofdstuk “Het project” aan bod. Hierin wordt het doel en de werking van het project uitgelegd. Ook komt een trajectenlijst aan bod die aangeeft welke trajecten gebruikt worden door het project.

Daarna worden de belangrijkste gegevens per provider neergeschreven. Deze gegevens worden gebruikt om onze back-end op te maken.

Tot slot is er een samenvattingstabel die heel kort herhaalt welke kenmerken iedere provider heeft.

## 14.2 Het project

Het mobiliteitsbedrijf van de stad Gent is sinds 2014 bezig met het opzetten van een regionaal verkeerscentrum. Men wil op termijn een manier ontwikkelen om (semi-)automatisch naar iedere weggebruiker in de regio Gent informatie door te spelen via een online platform. Om dit idee te verwezenlijken moet men eerst enkele voorbereidende projecten realiseren. Dit project is hier een grote stap in.

### Doel

Dit project heeft als bedoeling om periodiek van verschillende aanbieders (providers) reistijden op te vragen voor een aantal drukke trajecten in gent. Deze reistijden worden opgeslaan in een database. Dit gebeurt op een manier waarop het mobiliteitsbedrijf gemakkelijk deze gegevens kan raadplegen en vergelijken met elkaar om zo de meest relevante providers voor de regio Gent te achterhalen.

### Werking

Het project bestaat uit 3 grote fases:

1. analyse van de providers,
2. implementatie,
3. testfase.

#### 14.2.1.1 De analyse

De analyse van de providers is van cruciaal belang. Dankzij deze analyse zal het mobiliteitsbedrijf van Gent al een goede eerste indruk krijgen van de mogelijke providers.

De rest van dit document is volledig toegewijd tot dit onderwerp.

#### 14.2.1.2 De implementatie

Na de analyse zal een implementatie gemaakt worden van al de vergaarde kennis van de providers. Deze implementatie bevat het maken van een REST-applicatie die de data van de verschillende providers samenbrengt, deze verwerkt en zo een mogelijkheid geeft om gemakkelijk de providers te vergelijken.

#### 14.2.1.3 De testfase

Na het implementeren van het project zal de overige tijd besteed worden aan het testen van de verschillende functies.

## Trajecten

Om dit te verwezenlijken worden reistijden verzameld van de volgende trajecten.

ID	Name	Afstand (in m)	Letter
1	E40 (Oostende) > centrum	12176	A
2	E40 (Oostende) > centrum via Drongen	6677	B
3	E40 (Brussel) > centrum	8027	C
4	E17 (Kortrijk) > centrum	10479	D
5	E17 (Antwerpen) > centrum	8689	E
6	R4 Noordwest tegenwijzerzin	14966	F
7	R4 Noordwest wijzerzin	15310	F'
8	R4 West tegenwijzerzin	10508	G
9	R4 West wijzerzin	10131	G'
10	R4 Noordoost wijzerzin	12597	H
11	R4 Noordoost tegenwijzerzin	12682	H'
12	R4 Oost wijzerzin	17530	I
13	R4 Oost tegenwijzerzin	19149	I'
14	R40 Zuidwest wijzerzin	3719	J
15	R40 Zuidwest tegenwijzerzin	3824	J'
16	R40 Noordwest wijzerzin	1080	K
17	R40 Noordwest tegenwijzerzin	1104	K'
18	R40 Noordoost wijzerzin	1424	L
19	R40 Noordoost tegenwijzerzin	1447	L'
20	R40 Zuidoost wijzerzin	2112	M
21	R40 Zuidoost tegenwijzerzin	2246	M'
22	E40 (Oostende) > B402 via E40	6997	N
23	E40 (Oostende) > B402 via Drongen	12264	N'
24	E40 (Brussel) > B402	11493	O
25	E17 (Antwerpen) > B402 via E40	12994	P
26	E17 (Antwerpen) > B402 via E17/R4	13675	Q
27	E17 (Antwerpen) > B402 via R4	16948	R
28	E17 (Kortrijk) > B402	11294	S
29	E17 (Antwerpen) > P+R Gentbrugge	6096	T
30	E17 (Kortrijk) > P+R Gentbrugge	10473	U

## 14.3 Providers

### Google Maps

Google Maps is de meest gekende en meest gebruikte online kaartweergave. Over de jaren heen zijn er heel wat extra toepassingen toegevoegd aan Google Maps. Zo werd navigatie geïntegreerd voor zowel voetgangers, fietsers, auto, bus en trein. Deze werden aangevuld met data zoals fileproblemen, wegenwerken of ongevallen. Google Maps staat erom bekend om nauwkeurig te zijn. Het opnemen van deze provider zou zeker een meerwaarde bieden.



#### 14.3.1.1 Beschikbare data

- + Reistijden m.b.t. verkeersproblemen
- Oorzaken van verkeersproblemen / verkeersproblemen zelf

#### 14.3.1.2 Verbonden quota's

Er zijn verschillende quota's voor standaard en premium gebruikers:

##### 14.3.1.2.1 Standaard

- 2,500 elementen per dag (288 query's nodig, aantal elementen valt nog te bekijken)

*We rekenen voor iedere provider op 1 query iedere 5 minuten. Dit komt op 288 per dag, max. 8928 per maand en max. 105408 per jaar).*

- 100 elementen per query
- 100 elementen iedere 10 seconden
- \$0.50 USD / 1000 extra elementen, met een maximum van 100,000 per dag

##### 14.3.1.2.2 Premium

- 100,000 elementen per dag
- 625 elementen per query
- 1000 elementen iedere seconde
- 24/7 support

#### 14.3.1.3 API

De API van Google kan via een query benaderd worden. Deze kan echter geen verkeersproblemen weergeven. Wat deze wel kan is de tijd die je over het traject zal doen (afhankelijk van de verkeerssituatie) berekenen. Deze is ter beschikking gesteld onder de tag: "duration\_in\_traffic".

<https://developers.google.com/maps/documentation/distance-matrix/intro#Introduction>

##### 14.3.1.3.1 Voorbeeld van een request/response

Request (URL):

```
https://maps.googleapis.com/maps/api/distancematrix/json?  
origins=Vancouver+BC|Seattle&destinations=San+Francisco|Victoria+BC  
&mode=bicycling&language=fr-FR&key=YOUR\_API\_KEY
```

Response (JSON):

```
{  
  "status": "OK",  
  "origin_addresses": [ "Vancouver, BC, Canada", "Seattle, État de Washington, États-Unis" ],  
  "destination_addresses": [ "San Francisco, Californie, États-Unis", "Victoria, BC, Canada" ],  
  "rows": [ {  
    "elements": [ {  
      "status": "OK",  
      "duration": {  
        "value": 340110,  
        "text": "3 jours 22 heures"  
      },  
      "distance": {  
        "value": 1734542,  
        "text": "1 735 km"  
      }  
    }, {  
      "status": "OK",  
      "duration": {  
        "value": 24487,  
        "text": "6 heures 48 minutes"  
      },  
      "distance": {  
        "value": 129324,  
        "text": "129 km"  
      }  
    }  
  ]  
}]  
}
```

## Bing Maps

Bing Maps, ontwikkeld door Microsoft, levert voor België informatie in de vorm van Traffic Flow en Traffic Incidents. We kunnen dus alle huidige verkeersproblemen over een bepaalde route opvragen en per verkeersprobleem extra informatie te weten komen.



Met de Routes API worden reistijden aangeboden (*veranderen ze als er verkeersproblemen zijn?*).

### 14.3.1.4 Beschikbare data

- + Reistijden m.b.t. verkeersproblemen
- + Oorzaak van het verkeersprobleem

### 14.3.1.5 Verbonden quota's

Er zijn verschillende quota's voor basic- en enterprise keys:

#### 14.3.1.5.1 Basic keys

- 125,000 queries per jaar (1 query iedere 5 minuten: 114,336 queries per jaar nodig)

#### 14.3.1.5.2 Enterprise key

- Geen directe quotelimiet per jaar
- Betalend

### 14.3.1.6 API

- Traffic API: <https://msdn.microsoft.com/en-us/library/hh441725.aspx>
- Routes API: <https://msdn.microsoft.com/en-us/library/ff701717.aspx>

De gegevens kunnen zowel in XML als JSON formaat teruggegeven worden.

#### 14.3.1.6.1 Voorbeeld van een request/response

##### 14.3.1.6.1.1 Traffic API

Request:

```
http://dev.virtualearth.net/REST/v1/Traffic/Incidents/mapArea/  
includeLocationCodes?severity=severity1,severity2,severityn  
&type=type1,type2,typen&key=BingMapsKey
```

Response:

```
{  
  "authenticationResultCode": "ValidCredentials",  
  "brandLogoUri": "http://dev.virtualearth.net/Branding/logo_powered_by.png",  
  "copyright": "Copyright 2011 Microsoft and its suppliers. All rights reserved...",  
  "resourceSets": [  
    {  
      "estimatedTotal": 131,  
      "resources": [  
        {  
          "__type": "TrafficIncident:http://schemas.microsoft.com...",  
          "point": {  
            "type": "Point",  
            "coordinates": [  
              38.85135,  
              -94.34033  
            ]  
          },  
          "congestion": "",  
          "description": "M0-150 is closed between 5th Ave S and...",  
          "detour": "",  
          "end": "\\Date(1310396400000)\\",  
          "incidentId": 210546697,  
        }  
      ]  
    }  
  ]  
}
```

```

        "lane": "",
        "lastModified": "\/Date(1309391096593)\/",
        "roadClosed": true,
        "severity": 3,
        "start": "\/Date(1307365200000)\/",
        "type": 9,
        "verified": true
    },
    {
        "__type": "TrafficIncident:http://schemas.microsoft.com...",
        "point": {
            "type": "Point",
            "coordinates": [
                38.85872,
                -94.54638
            ]
        },
        "congestion": "",
        "description": "Botts Rd is closed between Andrews Rd and ...",
        "detour": "To go north take US-71 NB to 140th St and go west on ...",
        "end": "\/Date(1315244760000)\/",
        "incidentId": 191097424,
        "lane": "",
        "lastModified": "\/Date(1309391096593)\/",
        "roadClosed": true,
        "severity": 1,
        "start": "\/Date(1295704800000)\/",
        "type": 9,
        "verified": true
    }
]
},
"statusCode": 200,
"statusDescription": "OK",
"traceId": "38491198bf6a42f5b7e60c18aa08ec02"
}

```

#### 14.3.1.6.1.2 Routes API

Request:

```

http://dev.virtualearth.net/REST/V1/Routes/Driving?wp.0=redmond%2Cwa&wp.1=Issaquah%2Cwa&avoid=minimizeTolls&key=BingMapsKey

```

Response:

```
//Te veel data, zie 14.3.1.6 API.
```

## Coyote Systems

Coyote Systems specialiseert zich in radar-waarschuwingssystemen. Deze kleine toestellen vertellen de bestuurder als die een mobiele- of vaste snelheidsradar benaderen. Gebruikers kunnen gebruik maken van hun toestellen om andere bestuurders te waarschuwen indien men een nieuwe mobiele radar tegenkomt.



De website <https://maps.coyotesystems.com/traffic> toont een kaartoverzicht van alle huidige incidenten binnen Gent. Er is geen API beschikbaar, maar na wat reverse-engineering is op te merken dat de data volledig wordt verstuurd met behulp van een GET-request een JSON-response.

### 14.3.1.7 Beschikbare data

- + Real-time en normale reistijden
- + Huidige incidenten
- + Gesloten overeenkomst → mogelijkheid tot uitbreiding van de verkrijgbare gegevens

### 14.3.1.8 Verbonden quota's

Het mobiliteitsbedrijf van Gent heeft een gesloten overeenkomst met Coyote. Concreet zijn er momenteel dus geen quota's.

### 14.3.1.9 Ophalen van de gegevens

Om de gegevens op sub domein <http://maps.coyotesystems.com> af te halen, wordt geauthentiseerd met een username en password van Stad Gent.

<https://maps.coyotesystems.com/traffic/login.php>

Na authenticatie kan de data met de volgende URL opgevraagd worden:

[https://maps.coyotesystems.com/traffic/ajax/get\\_perturbation\\_list.ajax.php](https://maps.coyotesystems.com/traffic/ajax/get_perturbation_list.ajax.php)



Response:

```
{
  "Gand":{
    "Rooigemlaan (R40) Northbound - Drongensesteenweg - Palinghuizen":{
      "normal_time":47.821090909091,
      "real_time":51.911590909091,
      "diff_time":4.0905,
      "length":1340,
      "geometries":[
        [
          {
            "lng":3.69508,
            "lat":51.05617
          }, ... ,{
            "lng":3.69971,
            "lat":51.06629
          }
        ]
      ],
      "sections":[
        {
          "length":101,
          "nb_pos":1,
          "avg_speed":32,
          "speed_limit":50,
          "geometries":[
            {
              "lng":3.69508,
              "lat":51.05617
            }, ... ,{
              "lng":3.69413,
              "lat":51.05684
            }
          ]
        },
        {
          "ratio":1
        }
      ],
      "alerts":[]
    },
    "Rooigemlaan (R40) Southbound - Palinghuizen - Drongensesteenweg":{
      "normal_time":68.04,
      "real_time":88.538823529412,
      "diff_time":20.498823529412,
      "length":1353,
      "geometries":[
        [
          {
            "lng":3.69966,
            "lat":51.06634
          },
          {
            "lng":3.69492,
            "lat":51.0561
          }
        ]
      ],
      "sections":[
        {
          "ratio":1
        }
      ],
      "alerts":[]
    }, ...
  }, ...
},
```

```

"alerts":{
  "rm_208152683":{
    "id":"208152683",
    "type_id":"26",
    "lat":51.02987,
    "lng":3.65493,
    "heading":"294",
    "speed_limit":"50",
    "road_name":"",
    "first_declaration":"2016-02-23 09:11:49",
    "last_confirmation":"2016-02-23 09:33:38",
    "confirmations":"30",
    "length":"696",
    "zones":false,
    "type_lbl":"Incident"
  },
  "rm_208157503":{
    "id":"208157503",
    "type_id":"122",
    "lat":51.01702,
    "lng":3.66521,
    "heading":"252",
    "speed_limit":"70",
    "road_name":"N43",
    "first_declaration":"2016-02-23 09:16:01",
    "last_confirmation":"2016-02-23 09:22:21",
    "confirmations":"2",
    "length":"1999",
    "zones":false,
    "type_lbl":"Retrecissement"
  }, ...
}
}

```

## Waze

Waze is een community-based navigatie- en verkeersapp waarmee gebruikers real-time verkeersinformatie kunnen delen en ontvangen. Waze is verkrijgbaar in de App Store, Google Play en de Windows Phone Store.



### 14.3.1.10 Beschikbare data

- + Vertragingen
- + Huidige incidenten
- + Gesloten overeenkomst → mogelijkheid tot uitbreiding van de verkrijgbare gegevens

### 14.3.1.11 Verbonden quota's

Het mobiliteitsbedrijf van Gent heeft een gesloten overeenkomst met Waze. Concreet zijn er momenteel dus geen quota's.

### 14.3.1.12 Ophalen van de gegevens

Vooraleer informatie opgehaald kan worden, moet de sessie ingelogd zijn:

<https://www.waze.com/nl/signin?redirect=/nl/trafficview>

Request:

[https://www.waze.com/row-rtserver/web/TGeoRSS?ma=200&mj=100&mu=100&types=alerts%2Ctraffic&left=3.359584808349609&right=4.109401702880859&bottom=50.97371070722659&top=51.13082505687575&\\_id=1456225127992](https://www.waze.com/row-rtserver/web/TGeoRSS?ma=200&mj=100&mu=100&types=alerts%2Ctraffic&left=3.359584808349609&right=4.109401702880859&bottom=50.97371070722659&top=51.13082505687575&_id=1456225127992)

Response:

```
{
  "alerts": [
    {
      "country": "BE",
      "nThumbsUp": 0,
      "city": "Gent",
      "reportRating": 0,
      "reliability": 6,
      "type": "ROAD_CLOSED",
      "uuid": "52a54985-1c14-38ff-9d9e-0c2969530ee3",
      "speed": 0,
      "reportMood": 0,
      "subtype": "ROAD_CLOSED_EVENT",
      "street": "Sint-Denijslaan",
      "additionalInfo": "{\n  \"updatedAt\": 1449930148122,\n  \"createdBy\": 161264227,\n  \"location\": \"Sint-Denijslaan, Gent\",\n  \"eventId\": null\n}",
      "id": "alert-120286399/52a54985-1c14-38ff-9d9e-0c2969530ee3",
      "nComments": 0,
      "reportBy": "webke",
      "inscale": false,
      "comments": [
      ],
      "isJamUnifiedAlert": false,
      "confidence": 0,
      "nImages": 0,
      "magvar": 0,
      "showFacebookPic": false,
      "wazeData": "world,3.69009039,51.03474191,52a54985-1c14-38ff-9d9e-0c2969530ee3,120286399",
      "reportDescription": "Wegenwerken",
      "location": {

```

```

        "x":3.69009039,
        "y":51.03474191
    },
    "pubMillis":1440267642457
}, ... , {...}
],
"endTimeMillis":1456224840000,
"startTimeMillis":1456224780000,
"startTime":"2016-02-23 10:53:00:000",
"endTime":"2016-02-23 10:54:00:000",
"jams":[
    {
        "severity":4,
        "country":"BE",
        "city":"Gent",
        "level":5,
        "line":[
            {
                "x":3.740949,
                "y":51.041508
            },
            {
                "x":3.74035,
                "y":51.04175
            }
        ],
        "blockType":"ROAD_CLOSED_EVENT",
        "length":50,
        "turnType":"NONE",
        "type":"Medium",
        "blockingAlertID":122551833,
        "blockExpiration":1462114800000,
        "uuid":"7e5d9b24-4d64-35c5-9994-b99864c36519",
        "speed":0,
        "segments":[
            {
                "fromNode":95347580,
                "ID":88480562,
                "toNode":95349088,
                "isForward":false
            }
        ],
        "blockingAlertUuid":"7ddd565e-37b4-3051-be94-219d35254a72",
        "roadType":6,
        "delay":-1,
        "updateMillis":1456224857791,
        "street":"N9 Brusselsepoortstraat",
        "id":1770257980,
        "blockStartTime":1446418800000,
        "blockUpdate":1446493672219,
        "pubMillis":1455116459976
    }, ... ,{...}
]
}

```

Here

Here is een bedrijf dat deels in het bezit is van de autofabrikanten Audi en BMW. Doorheen de jaren heeft het bedrijf heel wat naamsveranderingen doorgemaakt. Origineel werd het bedrijf in Amerika opgericht onder de naam Navteq. Later werd het opgekocht door Nokia en kreeg het de nieuwe naam Nokia Maps. Toen Nokia in slechte papieren raakte heeft het bedrijf besloten om het te verkopen aan de groep van Duitse autodistributeurs Audi en BMW.



#### 14.3.1.13 Beschikbare data

- + Reistijden
- + Oorzaak van het verkeersprobleem

#### 14.3.1.14 Verbonden licenties

Er zijn verschillende pakketten beschikbaar:

##### 14.3.1.14.1 Evaluation

- Enkel voor development
- Toegang tot alle API's

##### 14.3.1.14.2 Basic

- Gratis
- 100,000 queries per maand (8,928 queries nodig)
- Routing API

##### 14.3.1.14.3 Advanced

- €2,990 per jaar
- 700,000 queries per maand
- Routing API
- Traffic Incidents

Meer info:

<https://developer.here.com/plans/api/consumer-mapping>

#### 14.3.1.15 API

In de onderstaande PDF op pagina tien "Requesting a Simple Route" staat een voorbeeld. De reistijd kan gevonden worden in de JSON response onder summary, trafficTime en travelTime.

[https://developer.here.com/documentation/download/routing\\_nlp/7.2.30/Routing%20API%20v7.2.30%20Developer's%20Guide.pdf](https://developer.here.com/documentation/download/routing_nlp/7.2.30/Routing%20API%20v7.2.30%20Developer's%20Guide.pdf)

De oorzaak van verkeersproblemen eveneens opgehaald worden via de traffic API. Het resultaat van deze request kan in XML als in JSON formaat teruggegeven worden. Here werkt met quadkeys. Deze worden gegenereerd aan de hand van een locatie om de kaart en een zoomlevel. Het is niet mogelijk om verkeersproblemen tussen twee punten aan te halen, enkel 'in de omgeving van'.

<https://developer.here.com/rest-apis/documentation/traffic/topics/quick-start.html>

#### 14.3.1.15.1 Ophalen van de gegevens

##### 14.3.1.15.1.1 Routes (Routing API)

Request:

[http://route.cit.api.here.com/routing/7.2/calculateroute.json?app\\_id=DemoAppId01082013GAL&app\\_code=AJKnXv84fjrb0KIHawS0Tg&waypoint0=geo!52.5,13.4&waypoint1=geo!52.5,13.45&mode=fastest;car;traffic:enabled](http://route.cit.api.here.com/routing/7.2/calculateroute.json?app_id=DemoAppId01082013GAL&app_code=AJKnXv84fjrb0KIHawS0Tg&waypoint0=geo!52.5,13.4&waypoint1=geo!52.5,13.45&mode=fastest;car;traffic:enabled)

Response:

```
{
  "response": {
    "metaInfo": {
      "timestamp": "2016-03-01T07:55:49Z",
      "mapVersion": "8.30.61.156",
      "moduleVersion": "7.2.66.0-1329",
      "interfaceVersion": "2.6.21"
    },
    "route": [
      {
        "waypoint": [
          {
            "linkId": "-53623477",
            "mappedPosition": {
              "latitude": 52.4999825,
              "longitude": 13.3999652
            },
            "originalPosition": {
              "latitude": 52.5,
              "longitude": 13.4
            },
            "type": "stopOver",
            "spot": 0.351145,
            "sideOfStreet": "left",
            "mappedRoadName": "Neuenburger Straße",
            "label": "Neuenburger Straße",
            "shapeIndex": 0
          },
          {
            "linkId": "+53473308",
            "mappedPosition": {
              "latitude": 52.4987912,
              "longitude": 13.4510744
            },
            "originalPosition": {
              "latitude": 52.5,
              "longitude": 13.45
            },
            "type": "stopOver",
            "spot": 1.0,
            "sideOfStreet": "neither",
            "mappedRoadName": "Schleusenufer",
            "label": "Schleusenufer",
            "shapeIndex": 55
          }
        ]
      }
    ]
  }
}
```

```

    "mode":{
      "type":"fastest",
      "transportModes":[
        "car"
      ],
      "trafficMode":"enabled",
      "feature":[
      ]
    },
    "leg":[
      {
        "start":{
          ...
        },
        "end":{
          ...
        },
        "length":4015,
        "travelTime":760,
        "maneuver":[
          {
            "position":{
              "latitude":52.4999825,
              "longitude":13.3999652
            },
            "instruction":"Head <span class=\"heading\">southeast</span>
on <span class=\"street\">Neuenburger Straße</span>. <span class=\"distance-
description\">Go for <span class=\"length\">46 m</span>.</span>",
            "travelTime":27,
            "length":46,
            "id":"M1",
            "_type":"PrivateTransportManeuverType"
          },
          {
            "position":{
              "latitude":52.4987912,
              "longitude":13.4510744
            },
            "instruction":"Arrive at <span
class=\"street\">Schleusenufer</span>.",
            "travelTime":0,
            "length":0,
            "id":"M10",
            "_type":"PrivateTransportManeuverType"
          }
        ]
      }
    ],
    "summary":{
      "distance":4015,
      "trafficTime":760,
      "baseTime":579,
      "flags":[
        "noThroughRoad",
        "builtUpArea",
        "park",
        "privateRoad"
      ],
      "text":"The trip takes <span class=\"length\">4.0 km</span> and <span
class=\"time\">13 mins</span>.",
      "travelTime":760,
      "_type":"RouteSummaryType"
    }
  }
}

```

```
    ],  
    "language": "en-us"  
  }  
}
```

#### 14.3.1.15.1.2 Verkeersproblemen (Traffic Incidents)

Er kan ook informatie over de huidige verkeersproblemen opgehaald worden. Er zijn geen concrete voorbeelden beschikbaar, maar de volledige documentatie is te vinden op:

<https://developer.here.com/rest-apis/documentation/traffic/topics/resource-parameters-incidents.html>



## TomTom

TomTom is een gekende fabrikant van GPS toestellen. Dit Nederlands bedrijf werd opgericht in 1991 en haalde een enorme omzet in de eerste tientallen jaren. De laatste jaren blijft hun omzet dalen en volgen ze hiermee dezelfde trend van de andere GPS fabrikanten. Deze dalende trend heeft als oorzaak de sterke concurrentie van gratis mobiele apps.



### 14.3.1.16 Beschikbare data

- + Reistijden
- + Verkeersproblemen & traffic flow

### 14.3.1.17 Verbonden licenties

Licenties worden verstrekt via een systeem van dagelijkse betaling. Er is een mogelijkheid om hun systeem te testen aan de hand van een evaluatie versie. Hoelang deze evaluatie versie gebruikt kan worden, het aantal query's die per dag verstuurd kunnen worden is niet duidelijk. Wanneer er besloten zou worden om na de evaluatie een betalende key aan te schaffen, is het onduidelijk hoeveel deze zal kosten. Hieromtrent moet overleg gepleegd worden met een verantwoordelijke salesman van TomTom.

### 14.3.1.18 API

Onderstaande link is een verwijzing naar het ontwikkelaarsportaal van TomTom. Hier kan sample data gegenereerd worden die ter beschikking wordt gesteld via hun API.

<http://developer.tomtom.com/io-docs>

Routing:

<http://developer.tomtom.com/products/onlinenavigation/onlinerouting>

Traffic:

<http://developer.tomtom.com/products/onlinenavigation/onetraffic>

#### 14.3.1.18.1 Ophalen van de gegevens

##### 14.3.1.18.1.1 Routing

Request:

```
https://api.tomtom.com/routing/1/calculateRoute/51.0845337,3.6645412:51.1923622,3.776722/json?routeType=fastest&traffic=true&language=en-GB&key=YOUR\_API\_KEY
```

Response:

```
{
  "formatVersion": "0.0.12",
  "copyright": "Copyright 2016 TomTom International BV. All rights...",
  "privacy": "TomTom keeps information that ...",
  "routes": [{
    "summary": {
      "lengthInMeters": 29661,
      "travelTimeInSeconds": 1530,
      "trafficDelayInSeconds": 0,
      "departureTime": "2016-03-01T10:41:40+01:00",
      "arrivalTime": "2016-03-01T11:07:09+01:00"
    },
    "legs": [{
      "summary": {
        "lengthInMeters": 29661,
        "travelTimeInSeconds": 1530,
        "trafficDelayInSeconds": 0,
        "departureTime": "2016-03-01T10:41:40+01:00",
        "arrivalTime": "2016-03-01T11:07:09+01:00"
      },
      "points": [{
        "latitude": 51.08456,
        "longitude": 3.66452
      }, ..., {
        "latitude": 51.19237,
        "longitude": 3.77670
      }
    ]
  }],
  "sections": [{
    "startPointIndex": 0,
    "endPointIndex": 295,
    "travelMode": "car"
  }
  ]
}
```

#### 14.3.1.18.1.2 Verkeersproblemen

Request:

```
https://api.tomtom.com/traffic/services/4/incidentDetails/s3/6841263.950712,511972.674418,6886056.049288,582676.925582/11/1335294634919/json?key=YOUR\_API\_KEY
```

Response:

```
{
  "tm":{
    "@id":"1456826213729",
    "poi":[
      {
        "id":"europe_CLUSTER_11_-1748358079",
        "p":{
          "x":573464.9,
          "y":6853453.4
        },
        "ic":8,
        "ty":0,
        "cbl":{
          "x":572278.9,
          "y":6852481.1
        },
        "ctr":{
          "x":574650.8,
          "y":6854425.7
        },
        "cs":4,
        "l":1100
      },
      {
        "id":"europe_HD_NL_TTR116257196019392",
        "p":{
          "x":554857.0,
          "y":6856767.4
        },
        "ic":8,
        "ty":4,
        "cs":0,
        "d":"closed",
        "f":"Kromwijkdreef (s113)",
        "t":"Gaasperdammerweg (A9)",
        "l":450
      }, ...
    ]
  }
}
```

## 14.4 Samenvatting

Om alle belangrijkste gegevens netjes te kunnen vergelijken met elkaar, wordt gebruik gemaakt van een kruistabel

	Google Maps	Bing Maps	Coyote Systems	Waze	Here API	Tom Tom
Opvragen van reistijden?	Ja	Ja	Ja	Ja	Ja	Ja
Opvragen van verkeersproblemen?	Nee	Ja	Ja	Ja	Ja	Ja
Gesloten overeenkomst?	Nee	Nee	Ja	Ja	Nee	Nee
Betalend plan nodig?	~	Nee	Ja	Ja	Ja	~
Prijs plan per jaar	~	€ 0	~	~	€ 2,990	~

*Gesloten overeenkomst: mogelijkheid tot uitbreidbaarheid, maar betalend*

*Is er een betalend plan nodig om de quota's te kunnen halen/de benodigde API's te kunnen gebruiken?*

~ = Onduidelijk