

Faculteit Ingenieurswetenschappen en Architectuur  
  
Schakelprogramma tot Master of Science in de   
Industriële Wetenschappen: Informatica

Vakoverschrijdend project (E767014)

Projectgroep 3

VOP Verkeerscentrum:

Software-Analyse

De Bock Jelle

Floré Brent

Stofferis Jeroen

Vandemoortele Simon

Vervenne Jan



Academiejaar 2015-2016



Faculteit Ingenieurswetenschappen en Architectuur  
  
Schakelprogramma tot Master of Science in de   
Industriële Wetenschappen: Informatica

Vakoverschrijdend project (E767014)

Projectgroep 3

VOP Verkeerscentrum:

Softwareanalyse

De Bock Jelle

Floré Brent

Stofferis Jeroen

Vandemoortele Simon

Vervenne Jan

Academiejaar 2015-2016

Inhoud

[Inleiding 3](#_Toc447719659)

[Analyse 4](#_Toc447719660)

[1 User Stories 5](#_Toc447719661)

[2 Use cases diagram 6](#_Toc447719662)

[3 Use cases 7](#_Toc447719663)

[3.1 Use Case 1: Meld abnormaliteiten 7](#_Toc447719664)

[3.2 Use Case 2: Bekijk reistijden 8](#_Toc447719665)

[3.3 Use Case 3: Controleer data 9](#_Toc447719666)

[3.4 Use Case 4: Bekijk actuele verkeerssituatie 10](#_Toc447719667)

[3.5 Use Case 5: Vergelijk verkeersinformatie 11](#_Toc447719668)

[3.6 Use Case 6: Data ophalen en hergebruiken 12](#_Toc447719669)

[Ontwerp 13](#_Toc447719670)

[4 Inleiding 14](#_Toc447719671)

[5 Providerapplicatie 14](#_Toc447719672)

[5.1 DCD 14](#_Toc447719673)

[5.1.1 Detailoverzicht scrapers 14](#_Toc447719674)

[5.1.2 Detailoverzicht domein 14](#_Toc447719675)

[5.1.3 Detailoverzicht repository/controller 14](#_Toc447719676)

[5.1.4 Legende 15](#_Toc447719677)

[5.2 Algoritmes 16](#_Toc447719678)

[6 Webapplicatie 16](#_Toc447719679)

[7 Databank 16](#_Toc447719680)

[7.1 ERD 16](#_Toc447719681)

[Testen 17](#_Toc447719682)

[Installatie en onderhoud 19](#_Toc447719683)

[Retrospective 21](#_Toc447719684)

[8 Aandachtspunten sprints 22](#_Toc447719685)

[8.1 Sprint 2 22](#_Toc447719686)

# Inleiding

Deze softwareanalyse is er gekomen naar aanleiding van het vakoverschrijdend eindproject. Dit is een vak in 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 is deelproblemen en deze zo beknopt mogelijk te formuleren. Het herwerken van deze bundel van vereisten is essentieel om een goeie oplossing te bekomen.

Hoofdstuk een behandelt de user stories. 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

# 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 verkeersituatie 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 |  | 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. |

# Use cases diagram



# Use cases

## Use Case 1: Meld abnormaliteiten

|  |  |
| --- | --- |
| **Use Case naam** | Meld abnormaliteiten |
| **Use case nummer** | 1 |
| **Toepassing** | Back-end |
| **Beschrijving** | Een abnormaliteit wordt automatisch gemeld via de beschikbare kanalen. |
| **Primaire Actoren** | het systeem |
| **Stakeholders** | De gebruikers van de communicatiekanalen en operator. |
| **Precondition** | nvt |
| **Trigger** | Een bepaalde hoeveelheid vertraging werd overschreden op een welbepaald traject. |
| **Basisverloop** | 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. |
| **Alternatief verloop** | nvt |
| **Domeinregels** | 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 |

## Use Case 2: Bekijk reistijden

|  |  |
| --- | --- |
| **Use Case naam** | Bekijk reistijden |
| **Use case nummer** | 2 |
| **Toepassing** | front end |
| **Beschrijving** | De actor bekijkt de reistijden die voor hem relevant zijn. |
| **Primaire Actoren** | Gebruiker, operator |
| **Precondition** | nvt |
| **Postcondition** | De actor heeft een overzicht van de reistijden. |
| **Trigger** | nvt |
| **Basisverloop** | 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. |
| **Alternatief verloop** | 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. |
| **Domeinregels** | 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] |

## Use Case 3: Controleer data

|  |  |
| --- | --- |
| **Use Case naam** | Controleer data |
| **Use case nummer** | 3 |
| **Toepassing** | back-end |
| **Beschrijving** | Periodiek de data in de database controleren en corrigerende updates doorvoeren. |
| **Primaire Actoren** | systeem |
| **Preconditie** | nvt |
| **Postconditie** | De optimale reistijden zijn represenatief voor de huidige verkeerssituatie op macroniveau. |
| **Trigger** | Elke week op middernacht. |
| **Basisverloop** | 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. |
| **Alternatief verloop** | nvt |
| **Domeinregels** | 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 |

## Use Case 4: Bekijk actuele verkeerssituatie

|  |  |
| --- | --- |
| **Use Case naam** | Bekijk actuele verkeerssituatie |
| **Use case nummer** | 4 |
| **Toepassing** | front end |
| **Beschrijving** | De actor bekijkt de actuele verkeerssituatie |
| **Primaire Actoren** | Gebruiker, Operator |
| **Stakeholders** | / |
| **Precondition** | / |
| **Postcondition** | De actor beschikt over de actuele verkeerssituatie. |
| **Basisverloop** | 1. De actor wenst de actuele verkeersituatie 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. |
| **Alternatief verloop** | 2A. Het systeem kan geen routes met bijhorende data ophalen. |
|  | 2A1. Het systeem geeft een foutboodschap weer. |
|  | 2A2. Einde use case zonder bereiken postconditie. |
| **Domeinregels** | 1. Verkeersituatie wordt weergegeven op een kaart. |

## Use Case 5: Vergelijk verkeersinformatie

|  |  |
| --- | --- |
| **Use Case naam** | Vergelijk verkeersinformatie |
| **Use case nummer** | 5 |
| **Toepassing** | front end |
| **Beschrijving** | De actor vergelijkt actuele of historische verkeerssituaties |
| **Primaire Actoren** | Operator |
| **Stakeholders** | / |
| **Precondition** | / |
| **Postcondition** | De actor kan verschillende verkeerssituaties met elkaar vergelijken. |
| **Basisverloop** | 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 verkeersituaties. |
|  | 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. |
| **Alternatief verloop** | 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 |
| **Domeinregels** | **Mogelijke types vergelijking:** |
|  | 1. Gegevens van verschillende momenten(uren, dagen, speciale perioden) |
|  | 2. Gegevens van verschillende trajecten |
|  | 3. Verkeerssituatie met bezettingsgraad bepaalde parking |

## Use Case 6: Data ophalen en hergebruiken

|  |  |
| --- | --- |
| **Use Case naam** | Data ophalen en hergebruiken |
| **Use case nummer** | 6 |
| **Toepassing** | front end |
| **Beschrijving** | De actor maakt een API call om de verzamelde data te krijgen en te hergebruiken |
| **Primaire Actoren** | Gebruiker |
| **Stakeholders** | / |
| **Precondition** | / |
| **Postcondition** | De actor beschikt over de verzamelde verkeersdata. |
| **Basisverloop** | 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. |
| **Alternatief verloop** | 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. |
| **Domeinregels** | 1. Verkeersituatie wordt weergegeven op een kaart. |

Ontwerp

# 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 verkeersituatie. Hiernaast zijn er voor mensen van het verkeercentrum nog meer mogelijkheden zoals het wijzigen van de op te volgen trajecten, geschiedenis van de verkeerssituatie vergelijken en meer.

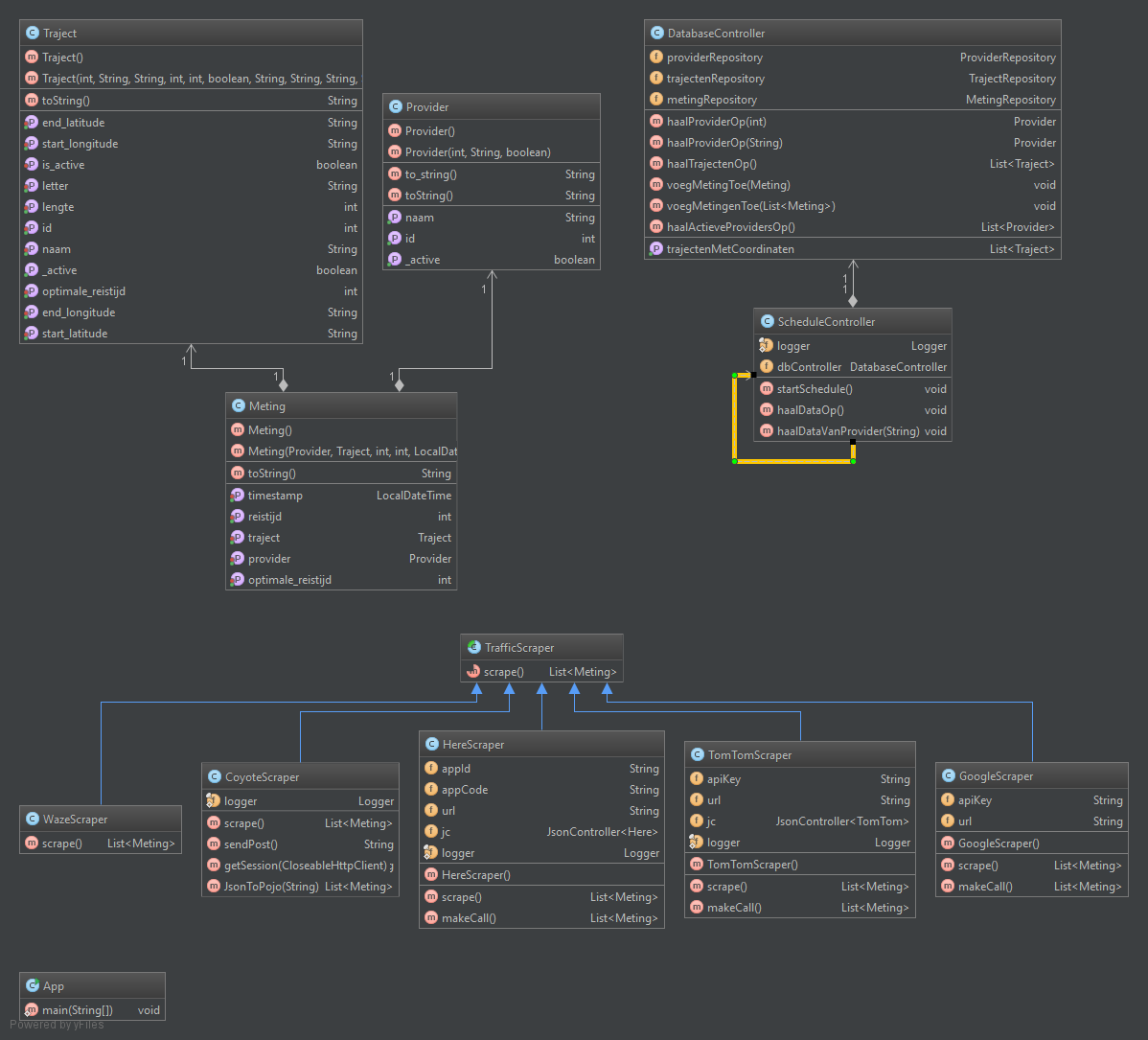
Natuurlijk is er ook een databank aanwezig achter de schermen waarin alle informatie wordt bijgehouden.

# Providerapplicatie

## State diagram



## DCD



### Detailoverzicht scrapers

### Detailoverzicht domein

### Detailoverzicht repository/controller

### Legende

|  |  |
| --- | --- |
| Icon | Description |
| classTypeJavaClass | Class |
| classTypeAbstract | Abstract Java class |
| classTypeAnnot | Annotation |
| classTypeEnum | Enumeration |
| classTypeException | Exception |
| img | Final Java class |
| classTypeInterface | Interface |
| classTypeMain | Java class that contains declaration of the main() method. |
| classTypeTestCase | Test case |
| classTypeJavaOutOfSourceRoot | Java class located out of the source root. Refer to the section [Configuring Content Roots](https://www.jetbrains.com/help/idea/2016.1/configuring-content-roots.html) for details. |
| excludeFromCompilation.png | Java class [excluded from compilation](https://www.jetbrains.com/help/idea/2016.1/configuring-compiler-settings.html). |
| method | Method |
| method_abstract | Abstract method |
| field | Field |
| variable | Variable |
| property | Property |
| property | Property with getter |
| property | Property with setter |
| property_yellow | Parameter |
| xml_element | Element |
| folder | Directory |
| moduleFolder | Module |
| groupOfModules | Group of modules |
| iconPackage | Package |
| rootSource | Source root |
| rootTest | Test root |
| rootExcluded | Excluded root |
| rootResourceIJ | Resources |
| rootTestResourceIJ | Test resources |
| rootGeneratedSourceIJ | Generated source roots |
| rootGeneratedTestSourceIJ | Generated test source roots |
| Visibility modifiers | |
| locked | Read-only class, e.g. from a jar of an external library. |
| private.png | private |
| protected.png | protected |
| package_protected.png | package protected |
| staticMark | static |
| public.png | public |

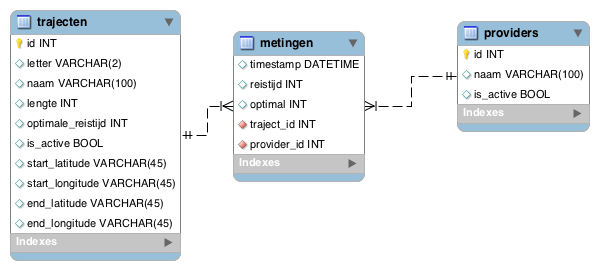
## Algoritmes

# Webapplicatie

## Algoritmes

# Databank

## ERD



Testen

Installatie en onderhoud

## Inleiding

Dit onderdeel van de software-analyse zal gaan over de installatie. Hieronder wordt in detail besproken hoe vanaf de broncode een werkend programma gemaakt kan worden.

## Benodigdheden

Om het project te laten runnen zijn er wat praktische componenten 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 Linux draaien. Een tweede component is een Windows toestel. Deze 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 keuze gemaakt worden welk besturingssysteem gebruikt zal worden. De keuze die gemaakt wordt hangt af van persoonlijk voorkeur. Het is aan te raden om een distributie te kiezen die eerder gericht is naar servers en niet naar desktops. In deze installatiehandleiding zullen twee distributies in detail besproken worden: enerzijds Fedora en anderzijds CentOS. De reden hiervoor is dat een Fedoraserver 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 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 onderhouden worden tot juni 2024.

Stabiliteit is eveneens een belangrijk aspect van dit project. Het is de bedoeling dat deze server (en het project) 24 uur per dag, 7 dagen per week draait. Hoe stabieler het besturingssysteem, hoe groter de kans dat dit zich ook daadwerkelijk in de praktijk voordoet. Zowel CentOS als Fedora staan bekend voor hun stabiliteit, dit is voor heel wat Linux distributies het geval. Door de vele releases van Fedora lijdt de stabiliteit eronder. Als een CentOS een label van 100% stabiel krijgt, dan komt dit overeen met 99.5% stabiliteit bij Fedora. Uiteindelijk zijn beide server geschikt als server en ligt de keuze bij de eindgebruiker.

#### 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 getfedora.com. Hier is het belangrijk om de serverversie te selecteren. De keuze voor 32-bit of een 64-bit systeem is afhankelijk van de fysieke server. De algemene regel is om een 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 een 64-bit installatie.



#### CentOS

#### GlashFisher

#### MariaDB

### Desktop

Retrospective

# Aandachtspunten sprints

## Sprint 2

* Structuur van domein(en eventueel databank). Elke meting heeft provider => één provider met lijst van metingen is conceptueel logischer.
* Optimalisatie netwerkverkeer: elke meting wordt individueel opgehaald, verwerkt en weggeschreven naar de databank => onnodige overhead, indien mogelijk data in één keer binnenhalen en in groep inserten in databank.
* De gegevens worden momenteel direct gebruikt in de querries. Dit vormt een beveiligingsrisico voor SQL-injection. =>Code herwerken zodat deze gebruik maakt van Prepared Statements.