



**Projectrapport**

## **NS Travel Agent – Travel to Fun**



**Jelmer Guiaux**



## Projectrapport

# NS Travel Agent – Travel to Fun



Auteur	: Jelmer Guiaux
Datum	: 22-03-2010
Versie	: 1.0
Afstudeerbedrijf	: Logica Nederland BV
Opdrachtgever	: Logica Nederland BV
Divisie	: Professional Skills
Competence	: BI & SOA
Bedrijfsbegeleider	: Bram Vranken
Onderwijsinstelling	: De Haagse Hogeschool
Opleiding	: Bedrijfskundige Informatica
Afstudeerbegeleider	: Arianne Luik
Afstudeerexpert	: Ed Meijer

---

## Referaat

- Projectrapport
- Semantiek
- Ontologie
- Normalized Google Distance
- Ambigüiteit
- Reisassistent
- Functioneel ontwerp
- Technisch ontwerp
- Proof of Concept

---

## Voorwoord

Geachte Lezer,

Voor u ligt het projectrapport "NS Travel Agent – Travel to Fun". Dit document beschrijft de bevindingen die gedaan zijn tijdens het project voor Logica. Dit document geeft inzicht op de vraag hoe een applicatie semantisch moet functioneren om een reiziger met de trein te adviseren over zijn wachttijd.

Voor het opstellen van dit document wil ik graag een aantal mensen bedanken. Als eerste Bram Vranken en Edwin Essenius voor hun toegewijde begeleiding en feedback, Maarten Schoonwater voor zijn input over de opdracht en mijn collega's van de afdeling Working Tomorrow voor hun hulp bij het opzetten van het proof of Concept. Als laatste wil ik dan nog mijn vrouw en dochter bedanken voor hun steun en mooie momenten buiten werktijd.

Ik wens u veel lees plezier.

Met vriendelijke groet,

Jelmer Guiaux

## Inhoud

<b>Samenvatting .....</b>	<b>9</b>
<b>1      Inleiding .....</b>	<b>11</b>
<b>2      Aanleiding .....</b>	<b>12</b>
2.1    Probleemstelling.....	12
2.2    Vraagstelling.....	12
2.3    Opdracht .....	12
2.4    Doelstellingen .....	13
2.5    Hoofdvraag .....	14
2.6    Deelvragen .....	14
<b>3      Onderzoek naar gebruiker &amp; techniek.....</b>	<b>16</b>
3.1    Gebruikers onderzoek .....	16
3.1.1 Doelgroep.....	16
3.1.2 Enquête.....	16
3.1.3 Resultaten enquête.....	16
3.1.4 Conclusie enquête.....	16
3.1.5 Interview.....	17
3.1.6 Resultaten interview .....	17
3.1.7 Conclusie interview .....	17
3.1.8 Persona .....	17
3.2    Technologie onderzoek.....	18
3.3    Theorie van Semantiek.....	18
3.3.1 Semantiek .....	18
3.3.2 Ontologie .....	18
3.3.2 Ambigüiteit .....	19
3.3.4 Context-awareness.....	20
3.3.5 Normalized Google Distance .....	20
3.4    Praktijk van Semantiek .....	21
3.4.1 Semantisch web .....	21
3.4.2 Resource Description Framework .....	22
3.4.3 Resource Description Framework Schema .....	23
3.4.4 Web Ontologie Language.....	23
3.4.5 WordNet.....	24
3.4.6 Protégé.....	24
3.4.7 Jena.....	25
3.4.8 Andere semantische applicaties.....	25
3.5    Conclusie .....	26
<b>4      Case beschrijving van reisassistent .....</b>	<b>32</b>
4.1    Een dag in het leven .....	32
4.2    Onderdelen van reisassistent.....	33
4.3    De adviezen .....	33
4.3.1 Variaties .....	33
4.3.2 Detaillering.....	34
4.4    Basis van advies .....	36
4.5    Toegevoegde waarden.....	36
4.5.1 Gebruiker .....	36
4.5.2 Bedrijven .....	37
4.6    Medium reisassistent.....	37
4.7    Conclusie .....	38
<b>5      Het Functioneel ontwerp.....</b>	<b>40</b>
5.1    Use Cases.....	40
5.2    Functionaliteiten naar Use Cases.....	40
5.3    Use Case Diagram.....	41
5.4    Invullen profiel .....	42
5.5    Plannen wachttijd.....	43

5.6	Basis Comfort Luxe Kwalificatie .....	44
5.7	Conclusie .....	45
<b>6</b>	<b>Het Technisch ontwerp .....</b>	<b>48</b>
6.1	Onderdelen reisassistent .....	48
6.2	Architectuur reisassistent .....	50
6.2.1	Beschrijving .....	50
6.2.2	Architectuur model .....	51
6.3	Database structuur .....	52
6.4	Uitgewerkte architectuur .....	54
6.4.1	Use Case profiel invullen .....	54
6.4.2	Use Case plannen wachttijd .....	57
6.5	Conclusie .....	60
<b>7</b>	<b>Het Proof of Concept .....</b>	<b>62</b>
7.1	Onderdelen Proof of Concept .....	62
7.2	Google vs Yahoo .....	62
7.3	Uitwerking POC .....	62
7.3.1	Database .....	63
7.3.2	NGD formule naar code .....	63
7.3.3	Database connectie .....	64
7.3.4	Tekstveld input gebruiker .....	64
7.3.5	Hoeveelheid zoekresultaten extraheren .....	66
7.3.6	NGD waarde krijgen .....	66
7.3.7	Uiteindelijke werking POC .....	67
7.4	Bevindingen tijdens uitwerking .....	68
7.4.1	Opvallende uitkomsten .....	68
7.4.2	Verwachte uitkomsten .....	69
7.4.3	Spelling woorden .....	69
7.4.4	Wanneer semantisch verband .....	69
7.4.5	Afronden getallen .....	69
7.4.6	Combinatie chocolade melk .....	69
7.4.7	Cash resultaten .....	69
<b>8</b>	<b>Conclusie onderzoeksrapport .....</b>	<b>70</b>
8.1	Antwoord deelvragen .....	70
8.2	Antwoord hoofdvraag .....	71
8.3	Volledige reisassistent .....	71
<b>9</b>	<b>Discussiepunten &amp; Actiepunten .....</b>	<b>72</b>
9.1	Discussie punten .....	72
9.2	Actiepunten .....	72
<b>10</b>	<b>Bronnenlijst .....</b>	<b>75</b>
<b>11</b>	<b>Afbeeldingenlijst .....</b>	<b>77</b>
<b>12</b>	<b>Tabellenlijst .....</b>	<b>77</b>
<b>13</b>	<b>Afkortingenlijst .....</b>	<b>78</b>
<b>14</b>	<b>Register .....</b>	<b>79</b>
<b>Bijlage .....</b>		<b>81</b>
<b>A.</b>	<b>Personas .....</b>	<b>82</b>
<b>B.</b>	<b>Resultaten enquête .....</b>	<b>84</b>
<b>C.</b>	<b>Lijst semantisch applicaties .....</b>	<b>87</b>







## Samenvatting

In deze samenvatting wordt beschreven wat de uitkomsten waren van het project de NS Travel Agent Travel to Fun. De aanleiding voor dit project was een probleemstelling bij de NS waarbij zijn innovatieve manieren zoeken om servicegraad van hun diensten zo hoog mogelijk te houden. Hoe hoger deze servicegraad hoe meer mensen de trein zien als waardig vervoersmiddel.

Om dit een innovatieve manier te vinden om de service graad te verbeteren is er een vraagstelling ontworpen waarin wordt gevraagd op welke manier een smartphone er voor kan zorgen dat de reiziger op de hoogte blijft van zijn reis, vertragingen krijgt doorgegeven en hoe hij zijn wachttijd kan opvullen op station. Deze vraagstelling is uiteindelijk gevormd in samenspraak met de opdrachtgever naar de volgende opdracht:

Ontwerp een reisassistent die de zakelijke reiziger kan adviseren over verschillende aspecten van zijn reis. Deze adviezen zal de service baseren op een profiel wat op semantische wijze wordt gekoppeld aan de mogelijkheden op en rond het station.

Hieruit is een hoofdvraag opgesteld die luid als volgt:

Hoe moet een reisassistent functioneren om op semantische wijze rekening te kunnen houden met de verschillende aspecten van een treinreis.

Daarbij komen de volgende deelvragen

1. Welke aspecten van de reis wil de gebruiker terug zien in de reisassistent?
2. Wat is de huidige status van semantische technologie?
3. Hoe vertalen de reisaspecten naar functionaliteiten in de reisassistent?
4. Hoe moeten de functionele eisen op semantische wijzen gaan samenwerken?

Hieronder volgt kort de antwoorden op de deelvragen en de hoofdvraag

### *1. Welke aspecten van de reis wil de gebruiker terug zien in de reisassistent?*

De doelgroep als eerste graag terug willen zien dat zij te weten krijgen wanneer er een vertraging is en welke alternatieve reisopties er dan is. Als tweede wil de doelgroep weten wat zij kunnen doen op het station in de vorm van een advies. Hierna komt het aangeven van een looproute door het station met als laatste het bijhouden van hun reis. Aan de hand van deze uitkomsten zijn er functionaliteiten opgesteld die in de Case terug komen.

### *2. Wat is de huidige status van semantische technologie?*

Uit dit stuk onderzoek is gebleken dat de beste manier voor de reisassistent om semantiek te bereiken is via de Normalized Google Distance(NGD) formule. Deze formule brekend op basis van twee woorden die een aantal zoekresultaten geven in Google of er een semantisch verband is. De voordelen van deze techniek is dat alle woorden altijd bekend zijn en er geen uitgebreide database gemaakt hoeft te worden waarin alle woorden staan beschreven met hun semantiek.

### *3. Hoe vertalen de reisaspecten naar functionaliteiten in de reisassistent?*

De reisaspecten zijn vertaald tot de volgende functionaliteiten:

1. Profiel in kunnen vullen
2. Je reis plannen
3. Het melden van vertragingen en een alternatieve route aangeboden krijgen
4. Adviezen krijgen om je wachttijd te vullen
5. Looproute kunnen ontvangen naar activiteit of aansluiting

Het profiel geeft één van de twee woorden die nodig zijn voor de NGD formule. Het plannen van de wachttijd is de functionaliteit die voor de adviezen zorgt en maakt gebruik van een database waarin alle artikelen staan die beschikbaar zijn voor het betreffende station, dit is het tweede woord voor de NGD. Verder zijn er nog het verkrijgen van reisinformatie, het melden van vertragingen met alternatieve reisopties en het krijgen van een looproute. Deze functionaliteiten zijn niet verder uitgewerkt is dit rapport en dus buiten beschouwing gelaten.

#### 4. Hoe moeten de functionele eisen op semantische wijzen gaan samenwerken?

Het antwoord op deze deelvraag is dat door middel van het profiel dat de gebruiker invoert met zijn hobby's en favoriete voedingswaar en de database waarin alle mogelijke artikelen staan van een station er wordt bepaald wat het beste advies is voor de gebruiker. Met deze twee gegevens wordt er een NGD waarde gegenereerd die er voor zorgt dat de twee onderdelen met het grootste semantisch verband boven aan de adviezen komen te staan. De gebruiker kan dan kiezen uit het advies welke hem het meeste aanspreekt en deze wordt dan uitgebreide getoond in de reisassistent.

#### Hoe moet een reisassistent functioneren om op semantische wijze rekening te kunnen houden met de verschillende aspecten van een treinreis.

Met het POC bewezen dat de NGD methode daadwerkelijk kan werken om automatisch een semantisch verband te bepalen tussen twee woorden waarvan één door een gebruiker is ingevoerd en één uit een database komt.

De aspecten van de treinreis zijn bepaald als aan de hand van het onderzoek en het verkrijgen van informatie over wat te doen in wachttijd is op nummer twee geëindigd als meest belangrijke reiaspecten. Hieruit is de functionaliteit wachttijd plannen geformuleerd die er voorzorgt dat er een persoonlijk advies komt waarmee de gebruiker zijn wachttijd kan vullen. Dit advies wordt op semantische wijze samengesteld aan de hand van de NGD formule. Met behulp van een profiel en een database met artikelen van betreffende stations wordt er een semantisch verband berekend. Aan de hand van dit verband komt er een advies met het grootste semantisch verband tussen de twee woorden.

# 1 Inleiding

Het project is gedaan bij Logica te Rotterdam met als betrokken partij de Nederlandse Spoorwegen. Dit document is geschreven voor de betrokken partijen om inzicht te geven in bevindingen die zijn gedaan voor het project NS Travel Agent – Travel to Fun. Hierin is er onderzocht hoe een mobiele applicatie de treinreiziger moet adviezen hoe hij zijn wachttijd kan vullen met de mogelijkheden op het station.

Voor dit project zijn een aantal vragen opgesteld die als basis zijn gebruikt voor de betreffende activiteiten. Hieronder staat de hoofdvraag beschreven met hierna de deelvragen

Hoe moet een reisassistent functioneren om op semantische wijze rekening te kunnen houden met de verschillende aspecten van een treinreis.

1. Welke aspecten van de reis wil de gebruiker terug zien in de reisassistent?
2. Wat is de huidige status van semantische technologie?
3. Hoe vertalen de reisaspecten naar functionaliteiten in de reisassistent?
4. Hoe moeten de functionele eisen op semantische wijzen gaan samenwerken?

In de onderstaande tabel staat beschreven in welke hoofdstukken de betreffende deelvragen zijn beantwoord en kort wat er in het hoofdstuk staat beschreven.

Vraag	Hoofdstuk	Beschrijving
De opdracht	2	In dit hoofdstuk wordt beschreven welke probleemstelling vooraf ging aan het project, wat de uiteindelijk opdracht was en welke doelstellingen het project kent. Tevens worden hier de hoofdvraag en deelvragen behandeld.
Deelvraag 1 & 2	3	In dit hoofdstuk wordt het onderzoek behandeld welke is uitgevoerd om deze deelvraag te beantwoorden. Het onderzoek is opgedeeld in een gebruikersonderzoek met een enquête en interviews en een technologie onderzoek wat bestaat uit een deskresearch en literatuurstudie
Deelvraag 3	4 & 5	Deze hoofdstukken zijn de Case en het functioneel ontwerp. Hierin wordt beschreven welke functionaliteiten de reisassistent moet hebben en hoe deze vertalen naar Use Cases.
Deelvraag 4	6	Dit hoofdstuk behandelt het technisch ontwerp waarin de Use Cases en andere onderdelen van de reisassistent zijn uitgewerkt in de vorm van onder andere database ontwerp en systeem architectuur
Hoofdvraag	7 & 8	In deze hoofdstukken wordt de hoofdvraag beantwoord. Dit wordt gedaan door een proof of concept te beschrijven die gemaakt is aan de hand van de voorgaande stukken. Hierna volgt een totale conclusie van het project
Aandachtspunten	9	In dit hoofdstuk wordt beschreven welke aandachtspunten er tijdens het project zijn gevonden maar niet verder zijn uitgewerkt.

## 2 Aanleiding

Waarom het project is gestart en wat het project uiteindelijk moet opleveren, wordt behandeld in het volgende hoofdstuk. Tevens wordt beschreven wat de opdracht is van het project, deze opdracht is geëxtraheerd uit de probleemstelling en vraagstelling. Er wordt begonnen met de probleemstelling gevolgd door de vraagstelling, de opdracht en doelstelling. Als laatste worden de hoofdvraag en deelvragen beschreven.

### 2.1 Probleemstelling

De Nederlandse Spoorwegen(NS) is in het bezit van een groot aantal treinen en andere soorten materieel verspreid over heel Nederland. Deze middelen kosten geld en dus moeten de treinen vol blijven met passagiers om deze kosten te kunnen dekken. Voor de NS is een voldoende bezettingsgraad van de treinen van primair belang en daarom continu een aandachtspunt. De NS blijft op zoek naar manieren om een hoge servicegraad te kunnen garanderen zodat passagiers tevreden met de trein reizen.

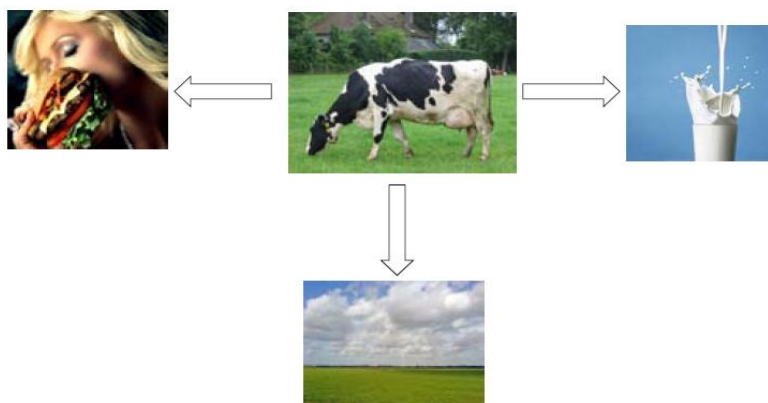
### 2.2 Vraagstelling

Om deze hoge service te kunnen leveren is de NS op zoek naar innovatieve manieren om de passagier te laten beschikken over reisinformatie. Vrijwel iedere passagier heeft een mobiele telefoon of laptop. Een steeds groter aantal van de telefoons zijn smartphones. Deze apparaten bieden tal van mogelijkheden om het de passagier makkelijker te maken. Hierbij ontstaat de volgende vraag: Hoe moet een mobiele informatie applicatie eruit zien die het reizen met de trein aantrekkelijker maakt? Welke aspecten zijn belangrijk bij een reis en welke behoeften heeft de reiziger?

De vraag vanuit de NS is om een mobiele reisassistent te ontwikkelen die de gebruiker kan assisteren bij het verloop van zijn reis. De reisassistent moet de reis van de gebruiker kunnen bewaken en alternatieve reisopties kunnen bieden indien er een vertraging is. Tevens moet de reisassistent het mogelijk maken om in geval van een vertraging, geplande tussenstop of bij een eindbestemming een advies te kunnen geven over de mogelijkheden op de desbetreffende locatie. Dit kan variëren van een plaats om te roken tot het vinden van een restaurant. De aanbeveling doet de reisassistent aan de hand van de interesses van de gebruiker, deze zijn ingevuld in een profiel. Om de gebruiker een goed advies te geven moet de reisassistent op semantische wijze de aanbevelingen verwerken in verhouding met de interesse van de gebruiker en de mogelijkheden op het station.

Semantiek heeft de letterlijke betekenis: 'betreffende betekenis van symbolen, woorden en zinnen'. Door de betekenis van een woord te gebruiken kan een applicatie beter bepalen wat een gebruiker bedoeld. Mede als hier de ontologie wordt gebruikt wat letterlijk: 'Met betrekking tot de leer der onderwerpen van het menselijk denkvermogen' betekend. Kan een applicatie, en begrijpen welke definitie van een woord gebruiker bedoeld, en hoe deze in relatie staat tot andere woorden.

Afbeelding vier is een goed voorbeeld van de betekenis van woorden en hoe deze woorden een relatie hebben tot andere woorden.



Afbeelding 1 semantiek voorbeeld 1

### 2.3 Opdracht

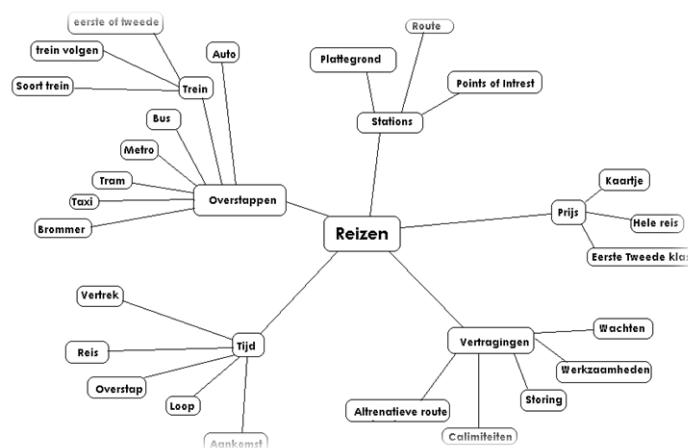
Aan de hand van de vraagstelling wordt de opdracht als volgt: Er moet een reisassistent ontworpen worden die de reis kan bijhouden en de een gebruiker kan adviseren over vertragingen en wat voor activiteiten hij kan doen op en

rond het station. Dit advies moet afgestemd zijn op de persoonlijke voorkeuren van de gebruiker in combinatie met de mogelijkheden en tijd op en rond het station.

Indien je in de trein zit wordt het mogelijk voor de reisassistent om bijvoorbeeld bij te houden waar je bent en je tijdig een signaal te geven dat je je laptop moet gaan inpakken omdat je zo moet uitstappen. Tevens kan de reisassistent bijvoorbeeld een plattegrond van je te lopen pad door het station geven naar een volgende aansluiting of je wijzen naar de dichtstbijzijnde verkooppunt van een kopje koffie. Een reiziger die met de trein reist en een kopje koffie wil tussen twee treinen in kan moeilijk inschatten hoeveel tijd hij nodig heeft om een kopje koffie te halen op een vreemd station. Daarom moet de reisassistent ook aangeven hoelang een bepaalde actie gaat duren. Om er voor te zorgen dat de reisassistent de relatie kan leggen tussen mogelijkheden op het station en het profiel van de gebruiker is het essentieel dat de reisassistent op semantische wijze gaat werken.

Als de geplande trein vertraging heeft is het de bedoeling dat de reisassistent een alternatieve route plant met daarbij een advies om de wachttijd te vullen. Is het warm buiten dan kan de reisassistent verwijzen naar een stand met koude dranken. Indien het 3 uur is wil de reiziger misschien koffie maar om 6 uur wil hij waarschijnlijk wat eten. Als de data op semantische wijze is gekoppeld kan de reisassistent deze conclusies maken en een passend advies geven

In afbeelding vijf is schematisch weergegeven hoe mogelijk bij een reis de verschillende aspecten in relatie staan met elkaar. De verbanden worden gelegd door de reisassistent waardoor de verschillende soorten informatie kan worden gecombineerd. Doordat alles met elkaar in verbinding staat is het ook mogelijk voor de reisassistent om te kunnen concluderen welke adviezen wel en welke niet relevant zijn.



Afbeelding 2 semantiek voorbeeld 2

Als de gebruiker moet wachten op een station en hij wil bijvoorbeeld weten waar hij een sigaretje kan roken om de tijd te doden kan de reisassistent hier assistentie in bieden. De gebruiker vraagt aan de reisassistent waar hij kan roken, de reisassistent weet waar dit is en kan dan ook een plattegrond van het station geven met daarop een route hoe te lopen naar een rookpaal.

Uiteindelijk komt het er op neer dat alle data met elkaar in verbinding staat op verschillende manieren. Zo staan elementen op meerdere manieren met elkaar in verbinding en wordt het mogelijk om een advies te genereren die rekening houdt met alle aspecten, van het weer tot de tijd van de dag.

De opdracht is om een reisassistent te ontwerpen die op semantische wijze rekening houdt met de verschillende aspecten van een treinreis. Hiervoor moet hij rekening houden met de wensen van de gebruiker en de mogelijkheden op het station.

## 2.4 Doelstellingen

Het ontwerpen van een reisassistent die op semantische wijze rekening kan houden met reisinformatie van verschillende bronnen. Om een dergelijke reisassistent te realiseren moet de vereiste informatiearchitectuur in kaart worden gebracht. Verder moet worden beschreven hoe de reisassistent op semantische wijze rekening gaat houden met verschillende soorten reisinformatie. Er dient antwoord te worden gegeven op de vraag hoe het metadatamodel eruit moet gaan zien en hoe deze gebruik gaat maken van de bestaande reisinformatie. Hierbij dient ook te worden onderzocht hoe gebruik kan worden gemaakt van een ontologietaal.

## 2.5 Hoofdvraag

Tijdens dit project zal er antwoord gegeven worden op de volgende hoofdvraag. De hoofdvraag dient tevens als leidraad voor de deelvragen.

Hoe moet een reisassistent functioneren om op semantische wijze rekening te kunnen houden met de verschillende aspecten van een trein reis.

Aan de hand van deze hoofdvraag kunnen er een aantal onderdelen opgesteld worden die ondernomen moeten worden. Als eerste moet vastgesteld worden welke aspecten van een treinreis een gebruiker wil weten waardoor de gebruiker baat heeft bij de reisassistent. Als tweede zal vastgesteld moeten worden wat semantiek is en hoe dit vertaald naar bestaande technieken. Dan moet er bepaald worden hoe de reisaspecten vertalen naar functionaliteiten in de reisassistent. En als laatste moet er bepaald worden hoe de semantische technieken de functionaliteiten gaan realiseren. Deze vragen vertalen naar de volgende deelvragen

## 2.6 Deelvragen

De volgende deelvragen zijn opgesteld aan de hand van de hoofdvraag. Wanneer de deelvragen zijn beantwoord is ook de hoofdvraag beantwoord.

1. Welke aspecten van de reis wil de gebruiker terug zien in de reisassistent?
2. Wat is de huidige status van semantische technologie?
3. Hoe vertalen de reisaspecten naar functionaliteiten in de reisassistent?
4. Hoe moet de gekozen semantische techniek de reisinformatie verwerken?



### 3 Onderzoek naar gebruiker & techniek

Om een reisassistent te ontwerpen moeten er eerst een aantal onderdelen uitgezocht worden. Deze onderdelen worden uitgezocht aan de hand van een gebruikersonderzoek en een technisch onderzoek. In het gebruikersonderzoek wordt er naar een antwoord gezocht op de eerste deelvraag: Welke aspecten van de reis wil de gebruiker terug zien in de reisassistent? Het gebruikersonderzoek bevat een verslag van een enquête die is gehouden onder de doelgroep en een verslag van een aantal interviews die zijn afgenomen onder de doelgroep. Met het technisch onderzoek wordt er naar een antwoord gezocht op de tweede deelvraag: wat is de huidige status van semantische technologie? Dit is gedaan door middel van een literatuur studie en een deskresearch.

#### 3.1 Gebruikers onderzoek

Tijdens dit project is er een gebruikersonderzoek uitgevoerd onder de samen met de opdrachtgever uitgewerkte doelgroep. Dit gebruikersonderzoek is gedaan om de functionaliteiten voor de reisassistent te specificeren en om bij de doelgroep te testen of zij interesse zouden hebben in een te ontwerpen reisassistent.

##### 3.1.1 Doelgroep

De doelgroep voor dit onderzoek is de zakelijke reiziger die momenteel weinig of geen gebruikmaakt van het OV. Mensen die overdag reizen voor zaken nemen momenteel meestal de auto. De NS wil graag dat deze mensen de trein nemen om hun treinen overdag te vullen.

Zakelijke reizigers zijn meestal in het bezit van een laptop of smartphone en genoeg werk om tijdens een treinreis te doen. De eerste reisassistent die ontworpen is draait daarom op een laptop met daarna de mogelijkheid om een smartphone versie te maken. Tevens reist een zakelijke reiziger meestal lange afstanden en komen voor de minder bekend trajecten en stations.

##### 3.1.2 Enquête

Om te achterhalen welke wensen de doelgroep heeft zijn er verschillende methoden, één daar van is een enquête en via deze methode is het mogelijk op een snelle manier een grote groep mensen te bereiken. Voor dit onderzoek is er ook gekozen voor een enquête waarmee de wensen en eisen van de doelgroep in kaart gebracht kunnen worden (1).

In de enquête worden er eerst een aantal basis vragen gesteld zoals hoe oud de respondent is en wat voor werk hij doet. Dan wordt er vastgesteld waarom hij/zij momenteel niet voor het OV kiest. Hiermee wordt er een beeld gevormd van de doelgroep. Hierop volgen een aantal vragen waarmee de functionaliteiten van de reisassistent gedefinieerd kunnen worden. Een al bekende functionaliteit van de reisassistent is dat hij moet gaan adviseren in geval van vertraging. Hierover zijn een aantal vragen gesteld waarmee kan worden vastgesteld welke informatie de doelgroep wil ontvangen. De laatste vraag betreft of de gebruiker bereid zou zijn om te betalen voor de reisassistent, deze vraag is gesteld uit persoonlijke interesse van de enquête maker

##### 3.1.3 Resultaten enquête

De enquête is verspreid onder de doelgroep binnen Logica en een aantal mensen buiten Logica. De enquête is drie weken beschikbaar geweest waarin het een totaal aantal respondenten heeft gekregen van 24. De enquête bestond uit 12 vragen en de gehele resultaten staan in de bijlage. In de volgende tekst zullen de resultaten geïnterpreteerd worden voor het functioneel ontwerp van de reisassistent.

Uit de enquête blijkt dat de meeste respondenten de auto pakken omdat het makkelijker is vanwege vertrektijd en snelheid. Natuurlijk vertrekt een auto wanneer iemand wil en meestal is de reistijd van deur tot deur minder met een auto. Als de reisassistent beschikbaar was zou 50% alsnog niet kiezen voor het OV maar 29% zou het wel overwegen met een reisassistent.

Als alle respondenten gebruik zouden maken van de reisassistent zouden 24 van de respondenten advies willen over alternatieve reisopties en wat de snelste alternatieve reisoptie is in geval van vertraging. 14 personen willen graag weten wat ze kunnen doen in hun wachttijd en daarbij moeten activiteiten niet verder dan loopafstand zijn van het station. 13 respondenten willen weten hoe ze ergens moeten komen met behulp van een plattegrond met looproute. Als laatste wil 58% van de respondenten niet betalen voor de reisassistent.

##### 3.1.4 Conclusie enquête

Uit deze resultaten kan geconcludeerd worden dat 91% van de doelgroep graag wil weten hoe hij kan reizen indien er een vertraging is en hoe lang de vertraging duurt. Er is minder animo voor een advies voor de wachttijd en een plattegrond van de looproute maar volgens wel als twee en drie van de meest gewilde opties voor de reisassistent. Als men een advies krijgt voor de wachttijd wil de doelgroep vooral weten waar hij wat kan eten of waar hij eten kan halen en waar hij een boek/krant of tijdschrift kan halen. Er moet ook geconcludeerd worden dat de helft van de doelgroep waarschijnlijk niet zal kiezen voor het OV als er een dergelijke reisassistent is en een selecte groep zal overwegen om gebruik te gaan maken van het OV.



### 3.1.5 Interview

Met behulp van een enquête is het mogelijk veel mensen te bereiken. Helaas is diepgang in een enquête niet mogelijk tenzij er een uitgebreide enquête gemaakt wordt die vrij lang duurt voor de respondent. Dit is meestal niet gewenst waardoor er meer respondenten afhaken. Om toch deze diepgang te krijgen is er voor gekozen om interviews af te nemen met de doelgroep.

### 3.1.6 Resultaten interview

Het interview met de doelgroep is afgenomen met twee personen die aan het reizen waren met de trein. Deze mensen vallen onder de doelgroep van zakelijke reiziger. Er is voor gekozen om een aantal interviews af te nemen met reizigers in de trein om erachter te komen welke aspecten van het treinreizen zij onduidelijk vinden. Met andere woorden waar de reisassistent informatie kan bieden aan de zakelijke reiziger. De twee geïnterviewde waren van middelbare leeftijd, beide waren man en waren in het bezit van een laptop. De vragen die aan hun gesteld werden waren dezelfde als die van de enquête maar dit keer met meer doorvragen over hun mening achter de vragen.

Uit de interviews is gebleken dat momenteel één persoon met het OV reist omdat de aansluiting adequaat is tussen zijn werk en huis. De ander reist met het OV vanwege werk wat verzet kan worden in de trein en op lange afstanden is het OV prettiger dan de auto. Wel is er een onvrede is over de hoeveelheid informatie die de NS verschaft tijdens een vertraging. Er wordt alleen vermeld hoeveel vertraging een trein heeft opgelopen maar niet wat te doen door de vertraging. Bij grote verstoringen wordt er wel vermeld welke alternatieve route te nemen maar deze is niet toegespitst op de persoon maar op het gros van de mensen. Graag zouden ze hier meer informatie ontvangen over de verschillende mogelijkheden of minder vertragingen. De adviezen tijdens de wachttijd was ook een interessante optie maar deze moeten niet dwingend over komen. Hiermee bedoelen ze dat de keuze altijd bij de gebruiker blijft en de reisassistent niet gaat voorschrijven wat de gebruiker leuk moet vinden. De looproute werd ook beschouwd als een mooie optie maar deze is ook onhandig omdat je niet de map kan bekijken terwijl je aan het lopen bent. De eerste reisassistent draait op een laptop en om deze open mee te nemen op het station is niet gewenst. Dit waren de belangrijkste bevindingen uit de interviews betreffende de functionaliteiten van de reisassistent.

### 3.1.7 Conclusie interview

Uit de interviews kan geconcludeerd worden dat er wel interesse is in de reisassistent maar dat hij wel een paar belangrijke functies goed moet uitvoeren. Voor de geïnterviewde is één van de belangrijkste functies de mogelijkheid om de reis realtime bij te houden en gewaarschuwd worden wanneer er een vertraging is. Hierbij is het belangrijk dat er informatie beschikbaar is over alternatieve opties om op de plaats van bestemming te komen. Hoewel de opties voor een wachttijdadvies en looproute niet de belangrijkste mogelijkheden waren volgens de geïnterviewde zouden ze wel van toegevoegde waarde zijn. Uit de interviews blijkt ook dat de laptop als medium misschien niet de beste keus is omdat deze niet gebruikt zal worden op het station.

### 3.1.8 Persona

Personas zijn een precieze beschrijving van een gebruiker en waarom hij of zij gebruik zou maken van het systeem. Personas zijn geen echte personen maar bedachte personen die echte personen zouden kunnen zijn. Om een duidelijk beeld te scheppen van de persona moeten ze zo specifiek mogelijk gemaakt worden. Zoals een naam zodat er tijdens het ontwerp proces naar de naam gerefereerd kan worden in plaats van "de gebruiker". (2)

Deze personas beschrijven twee personen van de doelgroep die gebruik zouden kunnen maken van de reisassistent. De personas zijn tot stand gekomen aan de hand van gesprekken met de opdrachtgever over de doelgroep en aan de hand van het gebruikersonderzoek. De eerste persona beschrijft een man die vrijwel dagelijks gebruik maakt van het OV en naar verschillende werkplekken reist. De tweede persona beschrijft een vrouw die beschikt over een auto en dus overgehaald moet worden om het OV te gebruiken. De personas staan beschreven in de bijlage.

### 3.2 Technologie onderzoek

In het bovenstaande stuk valt te lezen welke aspecten van een treinreis de reisassistent moet gaan verhelderen en wat de doelgroep van de reisassistent zullen vinden. Hoe moeten de aspecten gerealiseerd worden die zijn aangegeven door de gebruikers. In het volgende stuk staat dit beschreven. Er komt aan bod welke technologieën worden gebruikt en wat deze technologieën inhouden.

Het advies van voor de wachttijd wordt gebaseerd op de persoonlijke voorkeuren van de gebruiker door middel van een profiel. Om deze voorkeuren te koppelen aan mogelijkheden op het station en om deze mogelijkheden relevant te laten zijn aan de voorkeuren is er gekozen om deze semantisch te laten werken. Semantische technologie maakt het mogelijk om een gericht advies aan de gebruiker te geven. Dit is het onderwerp waar het technisch onderzoek zich vooral op heeft gericht.

### 3.3 Theorie van Semantiek

Als eerste volgt een beschrijving van de theorie achter semantiek. Hierbij wordt beschreven wat semantiek moeilijk maakt en welke complicaties er zijn voor het realiseren van een semantisch systeem.

#### 3.3.1 Semantiek

Het magische woord van deze reisassistent is semantiek, het lijkt de oplossing voor veel problemen die zich voordoen zowel op het internet als voor de reisassistent maar was is semantiek eigenlijk en hoe gaat het helpen om deze reisassistent te maken?

Wij gebruiken onze taal om te communiceren met anderen. Dit doen wij via spraak of schrift maar altijd heeft hetgeen wat wij zeggen een betekenis want als het niks betekend heeft het ook geen nut om te zeggen. Volgens het Computable ICT woordenboek (3) betekent semantiek:

- Semantic – *semantisch* - betreffende betekenis van symbolen, woorden en zinnen; als zelfstandig naamwoord: semantics

Wanneer wij iets communiceren weten wij vrijwel altijd wat de betekenis is achter het woord of de zin omdat wij de relatie tussen de woorden kunnen leggen. Als iemand zegt dat de ander een mooie das om heeft dan weten wij dat hij geen dood marterachtig roofdier om zijn nek heeft geknoopt. Zelfs wanneer deze zin is opgeschreven en het wordt gelezen weten wij nog waar men op doelt. De betekenis van een woord is dus sterk afhankelijk van de context waarin een woord wordt geplaatst (4). Dit is ambiguïteit, verder in het hoofdstuk wordt dit beschreven.

Voor een computer is het echter lastig om de betekenis van een woord te extraheren. Vooral als dit woord een homoniem is waardoor de zin bepaald in welke context het woord geplaatst moet worden. De informatie die op een computer of het internet staat is gemaakt door mensen, voor mensen. Hierdoor is alles voor een machine wel leesbaar maar het is niet voor een machine begrijpbaar (5). Dit zorgt er dus voor dat een machine met de huidige software nooit zal weten of het gaat om een dier of een kledingstuk.

Het maken van een machine die de mogelijkheid heeft om informatie te begrijpen wordt gezien als een stap dichterbij kunstmatige intelligentie (6). Wanneer een machine kan begrijpen wat er met een zin of woord bedoeld wordt aan de hand van de context kan een machine een van de belangrijkste kenmerken van de menselijke taal verwerken. Hoewel we hier nog ver vandaan zijn zou het een belangrijke stap zijn, hier wordt verder niet op in gegaan in dit rapport omdat een dergelijke studie een ruim rapport op zichzelf waard is.

#### 3.3.2 Ontologie

Ontologie is net als semantiek een begrip wat geregeld voorkomt wanneer men aan het zoeken is over het semantische web. Dit is ook niet gek want het heeft alles te maken met de functionaliteiten die een semantisch web moet gaan realiseren. Volgens het Computable ICT woordenboek (3) betekent ontologie

- Ontological – *ontologisch* – Met betrekking tot de leer der onderwerpen van het menselijk denkvermogen (ontology: ontologie)

Een ontologie is het woord voor hetgeen wat de relatie maakt tussen twee onderwerpen. Wel zijn er verschillende soorten van een ontologie, zo zijn er ook taxonomieën en folksonomieën (6). Deze zullen in de onderstaande tekst uitgelegd worden.

### Taxonomieën

Met een ontologie wordt een relatie gelegd tussen de twee objecten. In een taxonomie wordt het object ingedeeld in een klasse. Hiermee kunnen woorden worden ingedeeld in groepen. Deze relatie wordt genoemd als "is een". Een aantal voorbeelden zijn:

Das	is een	Dier
Das	is een	Marterachtige
Das	is een	Kleding accessoire

### Folksonomie

Een folksonomie is eigenlijk een soort taxonomie maar wordt in dit geval gemaakt door "het volk". Met een folksonomie wordt bedoeld dat mensen zelf een 'tag' kunnen geven aan objecten. Een tag is een beschrijving van het object in één woord. Dit object kan voorbeeld een afbeelding, film of woord zijn. Hiermee wordt het makkelijker om te zoeken op het internet. Een voorbeeld van een folksonomie is:

Das:

- Dier
- Roofdier
- Zwart/wit

### Ontologie

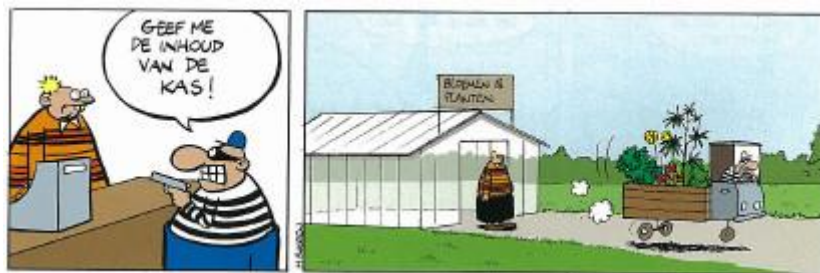
De vorige twee termen beschrijven maar een deel van de relatie tussen twee objecten. Bij een ontologie wordt de echte relatie beschreven tussen twee object. Een aantal voorbeelden van een ontologie zijn:

- Das woont in: Bos
- Das hangt in: Kast
- Das eet: Regenwormen

Met behulp van een ontologie is het mogelijk om de relaties te beschrijven die een mens van zichzelf al kan maken. Een machine zou in theorie dus kunnen begrijpen wat een mens bedoelt met een bepaalde zin. Alleen het verschil tussen theorie en praktijk is dat deze in theorie hetzelfde zijn, maar in de praktijk meestal niet. In de praktijk is het probleem dat de hoeveelheid van ontologie zo groot is dat dit lastig is om tot een goed functioneren systeem te laten komen. Tevens is taal zo interpreteerbaar dat het moeilijk is om de juiste ontologie aan elk object te hangen.

#### 3.3.2 Ambigüiteit

Een ander probleem voor het opzetten van een goede semantische reisassistent is ambigüiteit. Ambigüiteit betekent dat de betekenis van een woord afhankelijk is van de context waarin het geplaatst wordt. Voor mensen is dit over het algemeen geen probleem omdat de klemtoon meestal aangeeft welke betekenis er bedoeld wordt. Machines kunnen de klemtoon niet horen en zijn volledig afhankelijk van de vooraf gegeven betekenis en de context waarin het woord zich bevindt. Wanneer een taalconstructie meerdere betekennissen heeft wordt dit een equivociteit genoemd en één betekenis heet univociteit. (7)



Afbeelding 3 Lexicale ambigüiteit

Er zijn verschillende soorten ambigüiteit namelijk: Lexicale ambigüiteiten en syntactische ambigüiteiten (4).

#### *Lexicale*

Bij een lexicale ambigüiteit gaat fout wanneer de zin niet duidelijk maakt welke betekenis van het woord bedoeld wordt zie afbeelding 3. Bijvoorbeeld met de zin: 'Er ligt een das op de stoep'. Hier is niet uit te halen of het gaat om een dier of een stuk kleding.

#### *Syntactische*

Hier is syntactische ambigüiteit het probleem dat de analyse van de zin kan leiden tot verschillende interpretaties. Een voorbeeld hiervan is: 'De das loopt naar het bos'. Hierbij kan de das naar het bos willen maar hij kan ook richting het bos lopen en ergens anders naartoe willen wat op de weg naar het bos ligt.

Tot op heden is het niet mogelijk om deze ambigüiteiten goed te verhelpen tenzij meer informatie over de zin of het woord beschikbaar is en dat deze informatie kan worden verwerkt door de machine.

### 3.3.4 Context-awareness

Zoals hierboven beschreven is de context waarin een zin wordt geplaatst belangrijk om de zin te begrijpen. Maar wat ook belangrijk is om bepaalde zinnen te begrijpen is de situatie waarin de personen zich bevinden. Om er voor te zorgen dat een applicatie rekening houdt met de context waarin de gebruiker zich bevindt zijn de volgende elementen ontworpen.

Elementen die belangrijk zijn voor een context-aware applicatie zijn wie, waar, wanneer en wat (wat is in dit geval wat de gebruiker aan het doen is). Een voorbeeld hiervan is de plaats waar de gebruiker zich bevindt. Indien de gebruiker zich bevindt in Nederland zal de temperatuur in graden Celsius weergegeven worden. Bevindt de gebruiker zich in de Verenigde Staten dan zal de temperatuur in Fahrenheit weergegeven moeten worden. (8) (9)

Er zijn verschillende soorten context, zo is er primaire context types zoals locatie, identiteit, tijd en activiteit. Met deze context types kunnen de elementen waar, wie, wanneer en wat gedefinieerd worden. Tevens zijn er secundaire context types die geïdentificeerd kunnen worden met behulp van primaire context types. Als voorbeeld kunnen we de tijd nemen waardoor we kunnen bepalen of het dag of nacht is of dat er iets gedronken kan worden of het tijd is voor een avondmaal.

Veel onderzoekers definiëren een applicatie die context-aware is als een applicatie die dynamisch verandert aan de hand van de context van de applicatie en de gebruiker. De schrijvers van bron (9) hanteren voor een context-aware applicatie de volgende definitie:

*A system is context-aware if it uses context information to provide relevant information and/or services to the user, where relevancy depends on the user's task.*

Er zijn twee pogingen gedaan om een taxonomie te maken voor context-aware applicaties. Voor dit verslag is de taxonomie aangehouden van Schilit (10). Zo is een applicatie die handmatig informatie achterhaalt op basis van beschikbare context, een *proximate selection* applicatie genoemd. Dan zijn er applicaties die volledig automatisch informatie achterhalen en deze worden geclassificeerd als *automatic contextuel reconfiguration* applicaties. De applicaties die op commando van de gebruiker en op basis van context informatie achterhalen worden *contextual command* applicaties genoemd. Dan zijn er als laatste nog applicaties die op pas in werking treden als de goede combinatie van context is bereikt en een commando gaan uitvoeren, deze applicaties heten *context-triggered actions*.

### 3.3.5 Normalized Google Distance

In de voorgaande stukken van dit hoofdstuk valt te lezen hoe er met behulp van een uitgebreide opzet semantiek aan woorden kan worden toegevoegd die door machines te begrijpen is. Er is ook een andere manier om deze semantiek te verkrijgen. Deze techniek heet de Normalized Google Distance (NGD) en is ontworpen door twee onderzoekers van het Centrum Wiskunde & Informatica in Amsterdam (11).

Het internet wordt gebruikt door miljoenen mensen die allemaal karakters toevoegen en samen 50 miljard (12) webpagina's genereren die gemiddeld van lage kwaliteit zijn. Door deze enorme hoeveelheid informatie is het waarschijnlijk dat elk bedenkbare onderwerp, met extremen daar gelaten, beschikbaar is. Weliswaar van gemiddelde lage kwaliteit maar wel zinnig. Deze enorme hoeveelheid van lage kwaliteit informatie is gemaakt door personen over hun persoonlijke interesse waardoor er in principe een wereldwijde semantische database is gecreëerd. Deze database is beschikbaar voor iedereen en elke zoekmachine kan er zoek queries op loslaten. Deze queries geven een aantal webpagina's die geteld worden door pagecounters waardoor er geaggregeerde aantallen uit komen.

De logica achter de NGD is eigenlijk heel simpel alleen is de uitwerking wat ingewikkelder. De NGD werkt aan de hand van een vergelijking in het aantal pagina's die worden teruggeven door Google na een zoekopdracht. Een voorbeeld hiervan zijn de woorden *das*, *bos* en *papier*. Wanneer het woord *das* wordt gegeven als zoekopdracht geeft het 4.080.000 resultaten, het woord *bos* geeft 6.220.000 resultaten en het woord *papier* 6.060.000. Door de woorden te combineren kan er worden geconstateerd hoeveel de woorden met elkaar te maken hebben door kijken naar de hoeveelheid resultaten die dan terug komen. Een combinatie van de woorden *das* en *bos* geeft een aantal van 292.000 echter geeft *das* en *papier* een resultaat van 73.900. Hierdoor kan er geconcludeerd worden dat *das* en *bos* meer met elkaar te maken hebben dan *das* en *papier* (13) (14) (15).

Deze opzet is uitgewerkt in een formule waarbij wordt uitgerekend wat de NGD waarde is van bepaalde zoektermen.

$$NGD(x, y) = \frac{\max\{\log f(x), \log f(y)\} - \log f(x, y)}{\log M - \min\{\log f(x), \log f(y)\}}$$

In de formule is M het totaal aantal pagina's die Google heeft doorzocht. De x en y waarden zijn de aantallen van de zoekopdrachten die zijn uitgevoerd en de x, y is het aantal waarbij beide woorden als zoekopdracht zijn gebruikt.

De uitkomsten van de NGD formelen vallen tussen 0 en  $\infty$ . Hoe dichterbij de 0 hoe kleiner de afstand tussen de woorden en hoe groter het semantisch verband. Dit is uit te leggen aan de hand van het volgende voorbeeld. Als 'das' 100 pagina's oplevert, 'bos' 100 pagina's oplevert en 'das,bos' ook 100 pagina's oplevert geeft het dus aan dat op elke pagina waarop 'das' staat ook 'bos' staat. Hierdoor komt er een 0 uit de formule en hebben de twee woorden volgens Google een semantisch verband.

Hoewel het bereik tussen de 0 en  $\infty$  is zijn de meeste waarden tussen de 0 en 1. Uitkomsten boven de 1 zijn naar alle waarschijnlijkheid toe te wijzen aan het idee van negatieve correlatie. Een voorbeeld hiervan is 'door' en 'met'. Beide woorden worden veel gebruikt maar zijn synoniemen van elkaar waardoor een grote hoeveelheid websites of 'door' bevatten of 'met' maar waarschijnlijk niet beide.

Het aantal webpagina's geïndexeerd door Google komt in het bereik van  $10^{10}$  waardoor elke gebruikelijke zoekterm in miljoenen pagina's terug komt. Het aantal geïndexeerde webpagina's is zo groot dat het betekent dat er ook een groot aantal schrijvers van deze webpagina's moeten zijn. Hierdoor kan er vanuit worden gegaan dat het aantal schrijvers een representatieve steekproef is van de wereld bevolking. Hierdoor is het mogelijk om het aantal webpagina's geretourneerd van een zoekterm door Google te delen door het aantal geïndexeerde pagina's van Google en er een semantiek wordt gegenereerd die daadwerkelijk door de samenleving gebruikt wordt. Gebaseerd op deze theorie wordt er beweerd dat de NGD de ware semantiek achter de zoekterm kan representeren.

### 3.4 Praktijk van Semantiek

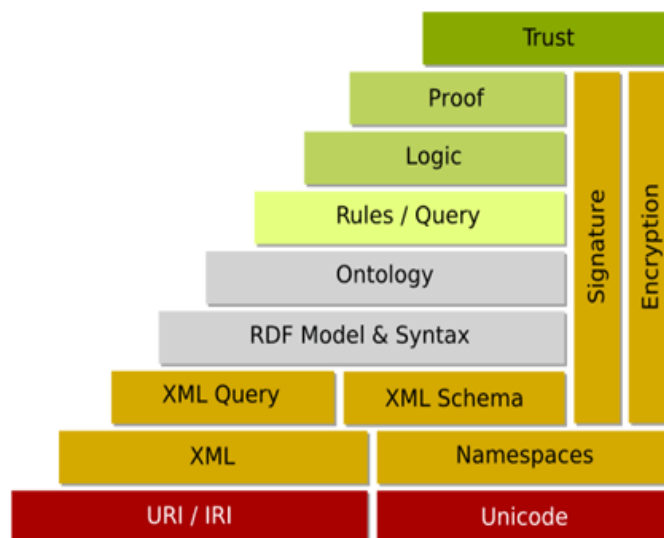
Hieronder volgt de praktijk van semantiek. Hoe wordt de theorie die hierboven beschreven is gerealiseerd in systemen en het internet. Welke mensen houden zich hiermee bezig en welke technieken hebben zijn ontworpen. Tevens wordt er gekeken naar al bestaande applicaties die semantiek gebruiken.

#### 3.4.1 Semantisch web

Het internet is een van de grootste bronnen van informatie en is voor een mens onmogelijk om alle informatie zelf te doorzoeken. Daarom zou het grote voordelen hebben als een computer begrijpt wat een gebruiker bedoelt. Denk hierbij aan zoekopdrachten die veel gericht antwoord geven of applicaties die voor jou informatie zoeken die bij een bepaald onderwerp past (16) (17) (18). Daarom zijn er inmiddels veel mensen bezig met het realiseren van het semantische web. Het semantische web is een uitspraak die veelvuldig voorkomt onder mensen die zich dagelijks bezig houden met het internet.

Mensen die zich bezig houden met het semantische web zijn onder andere Sir Tim Berners-Lee (19) en Frank van Harmelen (20). Sir Tim Berners-Lee is de oprichter van het internet zoals wij het nu kennen en streeft naar een semantisch web waarin ieder effectief informatie kan vinden. Sir Tim Berners-Lee is directeur bij het World Wide Web Consortium (W3C). Het W3C houdt zich bezig met het ontwikkelen van internet standaarden. De standaard is één van de belangrijkste punten van het semantisch web. Om het semantische web te laten werken is er informatie nodig maar deze moet wel in hetzelfde formaat aangeboden worden. Hyper Text Markup Language (HTML) en Extensible Markup Language (XML) zijn voorbeelden van een dergelijke standaard die nu al in gebruik zijn. Voor het semantische web zijn de standaarden Resource Description Framework (RDF) en Web Ontology Language (OWL) ontworpen maar hier later in het hoofdstuk meer over.

Tim Berners-lee heeft ook een schema ontworpen welke schematisch weergeeft hoe de ontwikkeling van het semantisch web zal gaan. In afbeelding 4 is te zien dat onderaan de technologie laag staat die nodig is om het semantische web te ontwikkelen (21). Er wordt begonnen met XML zoals het nu bestaat en breid zich verder uit naar RDF en uiteindelijk het gebruik van ontologie. Zo gaat de technologie steeds verder tot het punt van vertrouwen. Vertrouwen betekend dat de mensen die de systemen gaan gebruiken vertrouwen hebben in de systemen. Zonder dit vertrouwen zijn systemen gedoemd om te falen. Momenteel is de ontwikkeling van semantische technologie bij het maken van ontologie bereikt (22). Dit betekend dat het nog enige tijd gaat duren voordat een volledig semantisch web voltooid zal worden.



Afbeelding 4 Semantisch web opbouw

Frank van Harmelen is een professor aan de vrije universiteit van Amsterdam en heeft al artikelen en boeken geschreven over het semantische web. Tevens heeft hij op het European Semantic Web Conference in 2006 een aantal lezingen en interviews geven over het semantische web. Het semantische web is een moeilijk onderwerp maar zolang mensen zich ermee bezig houden zal er gestaag progressie zijn in de ontwikkeling ervan.

### 3.4.2 Resource Description Framework

Zoals eerder beschreven is het W3C (23) bezig met het ontwikkelen van standaarden voor het semantische web. Één van deze standaarden is RDF. RDF is eigenlijk een uitbreiding op het bestaande XML. Met XML is het mogelijk om gestructureerde data op te slaan in platte tekst. (24) XML maakt het mogelijk om folksonomie te definiëren van een bestand. Bijvoorbeeld bij een das kan er aangegeven worden dat het een das is en een marterachtig is maar je kunt er geen semantiek aan toevoegen.

XML

```
<soort dier>das</soort dier>
  <tags>roofdier, marterachtig, bos</tags>
```

(24)

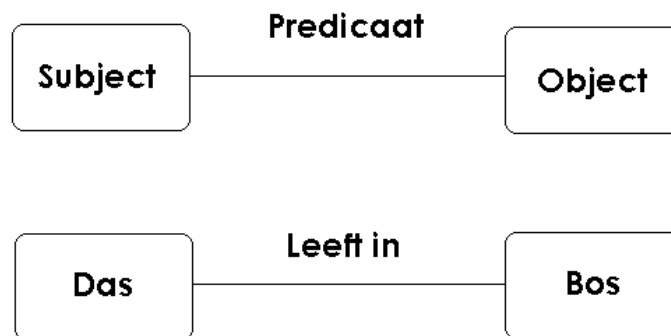
RDF maakt het mogelijk om taxonomieën toe te voegen aan de data maar nog wel in het formaat dat lijkt op een XML bestand. Hierdoor is het eenvoudig te begrijpen voor mens en machine. Een voorbeeld is het woord das waar je aan kan toevoegen welke relatie het heeft zoals: is familie van de marterachtigen en zijn leefomgeving. Om dit te realiseren werkt RDF aan de hand van een triple. Hierbij wordt er gebruik gemaakt van 3 elementen namelijk: Subject Predicaat en Object. De drie elementen zijn de basis van elke zin zoals wij die op school hebben geleerd. Ze werken als volgt:

1. Subject is het onderwerp
2. Predicaat (van Latijn praedicere) betekent letterlijk 'dat, wat ergens van gezegd' (25)
3. Object is hetgeen waar het subject bij hoort

(26)

Door deze structuur te verwerken in een RDF schema kan er een relatie worden gelegd tussen het subject en het object. Het predicaat beschrijft dan om wat voor een relatie het gaat. Afbeelding 5 Laat zien hoe dit in zijn werk gaat.

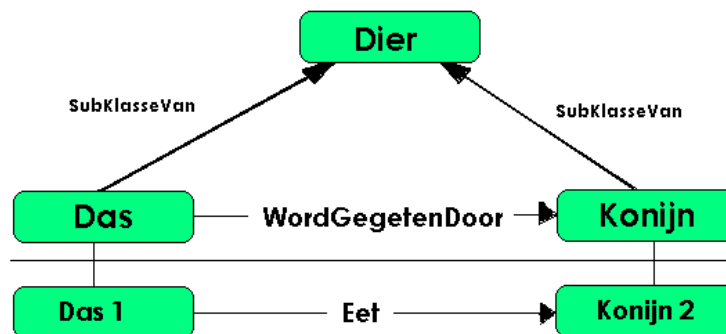




Afbeelding 5 Voorbeeld triple

### 3.4.3 Resource Description Framework Schema

RDFS is een uitbreiding van het bestaande RDF waarin de S staat voor Schema. RDFS is ook ontworpen door het W3C (27) en heeft als doel om het mogelijk te maken klasse te definiëren in RDF. Objecten kunnen met behulp van RDFS worden ingedeeld in een klasse. Je kunt als het ware een soort woordenboek maken over een bepaald onderwerp en in dit woordenboek kan je een hiërarchie aanbrengen. Je kunt het bijvoorbeeld hebben over dieren met verschillende soorten dieren zoals dassen en konijnen en hoe deze een relatie met elkaar hebben.



Afbeelding 6 Voorbeeld RDFS

Op afbeelding 6 is te zien hoe de klasse *dier* verschillende objecten heeft en zo is er bijvoorbeeld ook een klasse *kleren* waarin een *das* staat gedefinieerd met relatie als 'WordGedragenOver' tot een hemd.

Door deze opslag van data kunnen er ook betekenissen geëxtraheerd worden via het woordenboek. Als bijvoorbeeld onbekend is dat een *das* een *dier* is maar er wel bekend is dat *das1*, *konijn2* opeet kan er herleid worden dat een *das* ook een *dier* is. Mensen leggen deze relatie in taal zonder dat ze het zelf doorhebben. Als gezegd wordt dat Jelmer en Maggee getrouwd zijn dan weten anderen dat de relatie 'GetrouwdZijn' tussen twee mensen is. Zelfs al is het getrouwde stel nog nooit gezien is bekend dat het om twee mensen gaat en niet om twee huisdieren. Hiermee is het mogelijk om logica toe te voegen aan een RDFS bestand, er wordt een feit gegeven en aan de hand van dat feit kunnen er andere feiten achterhaald worden (22).

### 3.4.4 Web Ontologie Language

OWL is het Web Ontologie Language en zoals de naam al zegt is deze in staat om ontologie te verwerken. In deze taal kunnen alle soorten ontologie gedefinieerd worden zoals beschreven in de paragraaf over ontologie. OWL is ontworpen door het W3C (28), Web Ontology Working Group waar Frank van Harmelen een lid van is. Het semantische web zal gemaakt worden op de tagging van XML en de mogelijkheden die RDF biedt. Maar om een machine bruikbare redeneringen te laten maken op documenten moet er een taal zijn die de betekenissen kan geven van de gebruikte woorden in web documenten. Deze taal moet verder gaan dan de basis semantiek die RDFS biedt. OWL maakt het mogelijk om eigenschappen en klasse te definiëren en gaat verder met het werk van RDF(S).

OWL kan deze diepgang bieden maar doet dit wel in verschillende lagen. OWL is opgedeeld in drie lagen namelijk: OWL lite, OWL DL en OWL Full. OWL lite is zoals de naam al suggereert de lichtste versie van OWL en maakt het mogelijk om simpele hiërarchieën te maken met een kardinaliteit van 0 of 1. OWL DL is een stapje verder en maakt het mogelijk om wel maximale expressiviteit te halen maar beperkt in hoe deze gebruikt kan worden in samenwerking met RDF. Bij OWL Full is alles mogelijk betreffende de expressiviteit en de samenwerking met RDF. Hierdoor schiet de complexiteit wel door het dak en wordt het ook moeilijk voor machines om te verwerken (5).

### 3.4.5 WordNet

WordNet is een semantisch lexicon ontworpen door de universiteit van Princeton. Het beschrijft de semantische verbanden van de Engelse taal met onder andere zelfstandige naamwoorden, werkwoorden en bijvoeglijke naamwoorden. Het is gratis te downloaden voor verschillende platformen waardoor ieder het kan gebruiken om een semantische applicatie te bouwen. De meest recente versie is 3.0 gemaakt is 2005 waarin ongeveer 150.000 woorden, 115.000 synsets en 207.000 woord paren staan. Tevens is er een RDF versie beschikbaar via het W3C maar dit is versie 2.0 van WordNet.

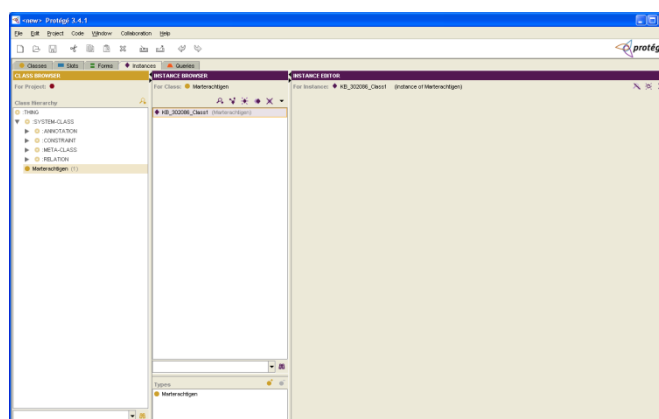
WordNet maakt gebruik van cognitieve synoniemen om een semantiek te realiseren, dit wordt door WordNet gedefinieerd als een synset. Een voorbeeld hiervan is: (wagen, auto, automobiel, motorvoertuig, machine). Het is een lexicale verbinding die op uiterlijk is gebaseerd waardoor bijvoorbeeld 'automobiel' is gerelateerd aan 'automobilist'. Tevens wordt er gebruik gemaakt van semantische relaties zoals; hyponiem, hyperoniem, meronym en antoniem (6) (25) (29). Een aantal voorbeelden hiervan zijn:

- Das is een hyponiem van Dier
- Dier is een hyperoniem van Das
- Poot is een meronym van Das
- Rijk is een antoniem van Arm

Wel kent WordNet een aantal problemen dan wel beperkingen. In WordNet staan geen etymologieën (geschiedenis van een woord en hoe deze veranderd door tijd), uitspraken en onregelmatige werkwoorden beschreven. Hierdoor is het gebruik van de informatie dus beperkt tot hetgeen wat staat gedefinieerd in de lexicale verbindingen.

### 3.4.6 Protégé

Protégé is een programma om zelf ontologie en knowledge-base frameworks te maken. Het is een gratis, open source programma dat verkrijgbaar is in Protégé-Frames en Protégé-OWL. Protégé maakt het mogelijk om ontologie te exporteren naar RDF(S), OWL en XML Schema. Protégé is gebaseerd op Java met een grafische interface die het mogelijk maakt om snel een prototype of applicatie te ontwikkelen. Momenteel kent Protégé 7,500 geregistreerde gebruikers.



Afbeelding 7 Voorbeeld Protégé

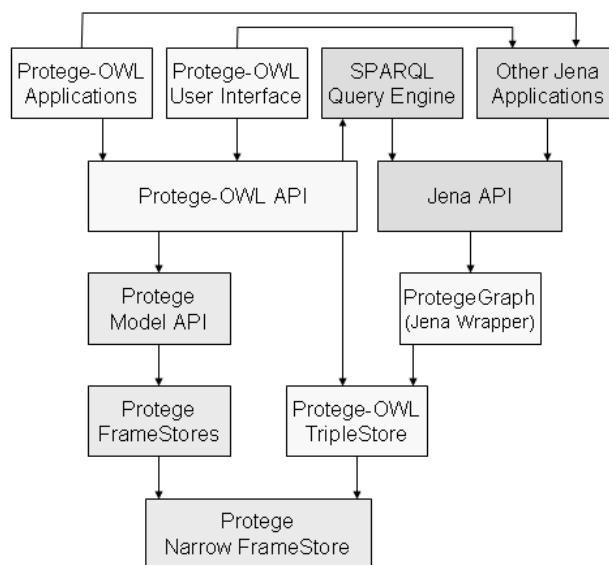
Protégé heeft een aantal belangrijke kenmerken waardoor het een eenvoudiger(er) programma is voor gebruikers. Waaronder valt dat ze een grafische interface bieden waarmee gebruikers model klasse kunnen maken met de daar bijhoren attributen en relaties. Het is ook mogelijk om in Protégé de verschillende OWL variaties te gebruiken (30).



### 3.4.7 Jena

Jena is een framework voor Java waarmee het mogelijk wordt om semantische applicaties te maken. Met behulp van Jena wordt het mogelijk om in een programmeer omgeving RDF, RDFS en OWL te gebruiken. Tevens biedt het een omgeving waar de Protocol and RDF Query Language (SPARQL) gebruikt kan worden en een rule-based inference engine. SPARQL is de query taal om query los te laten op een RDF database. Een inference engine is een programma dat antwoorden probeert te halen uit een knowledge database zoals een RDF database.

Jena maakt het mogelijk om semantiek te verwerken in een Java applicatie. Hierdoor is Jena ook een integraal onderdeel van Protégé. In Protégé zorgt Jena ervoor dat de modellen die gemaakt worden in Protégé omgezet kunnen worden naar leesbare RDF en OWL bestanden voor Java. In afbeelding 8 is te zien hoe Jena past in het hele plaatje van Protégé, OWL en SPARQL.



Afbeelding 8 Plaatsing Jena

### 3.4.8 Andere semantische applicaties

Wat er in dit rapport beschreven staat is maar het topje van de ijsberg. Betreffende het semantische web en ontwikkelingen op dat gebied zijn boekwerken over te schrijven. Tevens zijn er al verschillende onderzoeken gedaan op de HHS en op de WT naar het semantische web en haar onderdelen. In deze onderzoeken komt ruim aan bod welke ontwikkelingen er op het internet zijn betreffende semantisch zoeken, semantische applicaties en hoe semantiek toegepast kan worden in het bedrijfsleven.

Indien u meer wilt weten over het onderwerp en al zijn facetten verwijs ik u door naar de bijlage en bronnenlijst. In de bronnenlijst staan alle bronnen die gebruikt zijn voor dit onderzoek en in de bijlage staat een lijst met daarin alle gevonden boeken over semantiek en een lijst met applicaties die semantiek gebruiken.

### 3.5 Conclusie

In dit hoofdstuk wordt beschreven wat het resultaat was van het onderzoek naar de doelgroep en welke ontwikkelingen er zijn op het gebied van semantiek. Hiermee wordt er ook antwoord gegeven op de eerste en tweede deelvraag. Deze vragen zijn:

1. Welke aspecten van de reis wil de gebruiker terug zien in de reisassistent?
2. Wat is de huidige status van semantische technologie?

De eerste deelvraag is beantwoord met behulp van het gebruikersonderzoek. Dit onderzoek was in de vorm van een enquête en een interview met de doelgroep. De tweede deelvraag is beantwoord met behulp van een literatuuronderzoek en deskresearch.

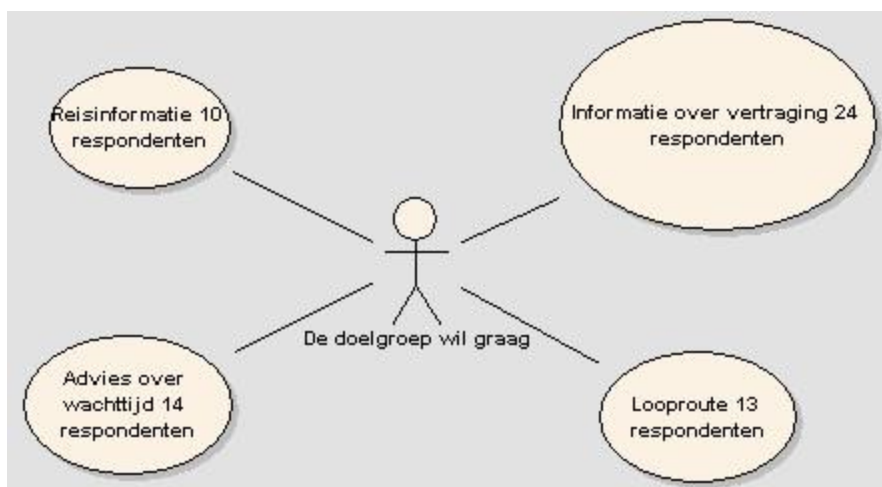
#### 3.5.1 Eerste deelvraag

Aan de hand van de resultaten van het gebruikersonderzoek moet de eerste deelvraag beantwoordt worden. Deze deelvraag is: Welke aspecten van de reis wil de gebruiker terug zien in de reisassistent? Het antwoord hierop is het volgende:

Met behulp van de enquête is naar voren gekomen dat de doelgroep een behoefte heeft aan informatie over vertragingen en welke alternatieve reisopties er zijn in geval van een vertraging. Hierop volgt de dat de doelgroep wil weten wat hij allemaal kan doen in de tijd die ertussen twee aansluitingen zit. Hoewel deze informatie zich niet alleen beperkt tot de tijd tussen twee treinen. De doelgroep wil ook weten waar hij zou kunnen eten of eten zou kunnen halen. Dit zou resulteren in een restaurant of bar op loopafstand van het station. Tevens zou de doelgroep willen weten hoe hij moet lopen naar een overstap of activiteit. Hierbij gaat het niet direct van spoor naar spoor want dit is vrij eenvoudig maar van spoor naar winkel of bus. Dit is over het algemeen moeilijker te vinden op een vreemd station en kan dus handige informatie zijn.

Uit de interviews kan hetzelfde geconcludeerd worden maar wat tijdens de interviews aan het licht kwam is dat de reisassistent wel minstens deze 3 opties moet geven wil de doelgroep gebruik gaan maken van de reisassistent.

Op de vraag over welke reisaspecten de reisassistent informatie zou moeten bieden was de verdeling van de top vier antwoorden het volgende:



Afbeelding 9 Verdeling enquête resultaten

Dit vertaalt naar een aantal functionaliteiten die de reisassistent moet bezitten:

- Je reis plannen
- Het melden van vertragingen en een alternatieve route aangeboden krijgen
- Profiel in kunnen vullen
- Adviezen krijgen om je wachttijd te vullen
- Looproute kunnen ontvangen naar activiteit of aansluiting

Aan de hand van deze functionaliteiten zal de reisassistent verder uitgewerkt worden.

### 3.5.2 Tweede deelvraag

Het antwoord op de vraag: Wat is de huidige status van semantische technologie? Is het volgende:

In de afgelopen jaren is er veel gebeurd op het gebied van semantiek en het semantische web. Er zijn verschillende standaarden ontwikkeld waaronder RDF en OWL. Deze standaarden verwerken op verschillende niveaus ontologie. Het is namelijk onmogelijk om in één keer een semantische applicatie of het web te maken. Hiervoor zijn een aantal stappen nodig die beginnen bij taxonomie en folksonomie maar zich uiteindelijk zal uitbreiden naar ontologie. Een groot probleem hierbij is ambiguïteit waardoor er niet valt af te lezen aan de hand van de context waarin het woord is geplaatst welke betekenis het woord moet hebben. Het uiteindelijke doel is om de data in onze machine niet alleen machine leesbaar te maken maar ook machine begrijpbaar.

Tevens is er al geruime hoeveelheid literatuur verschenen over het semantische web en hoe deze zou moeten werken met de standaarden. Mensen zoals Sier Tim Berners-Lee en Frank van Harmelen zetten zich dagelijks in om het semantische web een realiteit te laten worden. Met als gevolg dat er steeds meer gebeurt op het gebied van semantiek en er ook interesse wordt gewekt bij bedrijven.

Het aantal programma's om semantische applicatie te maken groeit ook en deze worden steeds uitgebreider waardoor het makkelijker wordt voor de 'gewone' gebruiker om semantiek te verwerken. Het betreft hier programma's als Wordnet, Protégé, en Jena.

Het is ook mogelijk om semantiek te krijgen op een 'brute force' manier. Hierbij wordt er geen gebruik gemaakt van uitgebreide databases waarin ontologie staat beschreven per woord maar word het huidige internet gebruikt als database. Hierbij wordt de NGD uitgerekend. Bij de NGD worden de pagecounts van zoekopdrachten vergeleken met de hoeveelheid geïndexeerde pagina's en aan de hand daarvan word bepaald of woorden wel of niet met elkaar te maken hebben.

### 3.5.3 Advies voor de reisassistent

In de conclusie is te lezen wat de uitkomsten zijn van het onderzoek en hoe deze de eerste twee deelvragen beantwoorden. Het geeft nog geen antwoord op hoe de reisassistent ontworpen zou moeten worden. Met andere woorden, in welke programmeertaal moet de reisassistent gemaakt worden, welke techniek gebruikt moet er worden en hoe gaan de onderdelen samen werken. Om een reisassistent te maken die de gebruiker kan assisteren bij zijn reis en hem advies kan geven die voor hem van toepassing zijn en dit advies op semantische wijze maakt is het volgende het advies.

Door de reisassistent semantisch te maken wordt er een functie toegevoegd aan de reisassistent waardoor het mogelijk is om een advies te geven die is toegespitst op de gebruiker. Doordat de reisassistent semantisch is wordt het mogelijk om te bepalen dat een gebruiker met de hobby motorrijden geïnteresseerd is in een motorblad. Door deze functionaliteit toe te voegen wordt het gebruiken van de reisassistent persoonlijker en kan de gebruiker adviezen ontvangen die direct op hem zijn toegespitst. Tevens geeft de semantiek in combinatie met een context-awareness de reisassistent een unieke functionaliteit om de gebruiker beter en sneller te bedienen tijdens zijn reis.

Er is uiteindelijk voor gekozen om de NGD formule te gebruiken in de reisassistent om semantiek te bereiken. Hier zijn de volgende overwegingen aan vooraf gegaan:

Techniek → Overwegingen ↓	Conventionele technieken(RDF(S), OWL)	NGD
Mate van complexiteit	Deze technieken zijn erg lastig te realiseren omdat voor elk woord er een betekenis(sen) gedefinieerd moet worden evenals welke relatie het heeft tot een ander woord.	Deze techniek is relatief eenvoudig omdat het alleen een formule betreft die uitgerekend moet worden om de semantiek tussen twee woorden te bepalen.
Hoeveelheid bekende woorden	Bij deze techniek moet dus elk woord gedefinieerd worden door iemand. Dit kost veel tijd en moeite. Tevens is er geen garantie dat alle aangewezen betekenissen correct zijn	Google kent alle woorden omdat deze door de wereld bevolking worden toegewezen. Hierdoor kan er gebruik gemaakt worden van een uitgebreide database die altijd up to data is.
Reeds toegepast	Deze technieken zijn al werkend op het web te vinden in de vorm van bijvoorbeeld WordNet, Protégé en Jena.	Deze techniek is voor zover bekend nog niet toegepast in een applicatie.
Geschikt voor doeleinde reisassistent	Deze techniek is op zich geschikt voor de doeleinde maar is wel erg uitgebreid en bied meer functies dan nodig.	Deze techniek is geschikter voor de doeleinden omdat er alleen maar bekend hoeft te zijn of er een relatie is tussen twee woorden en niet wat die relatie is.
Te realiseren in beschikbare tijd	Omdat deze techniek erg ingewikkeld is, is deze ook moeilijk te realiseren in de beschikbare tijd.	Deze techniek is relatief eenvoudig te realiseren. Het moeilijkste is om de hoeveelheid zoekresultaten te extraheren van Google. Verder betreft het redelijk standaard elementen voor een programma.

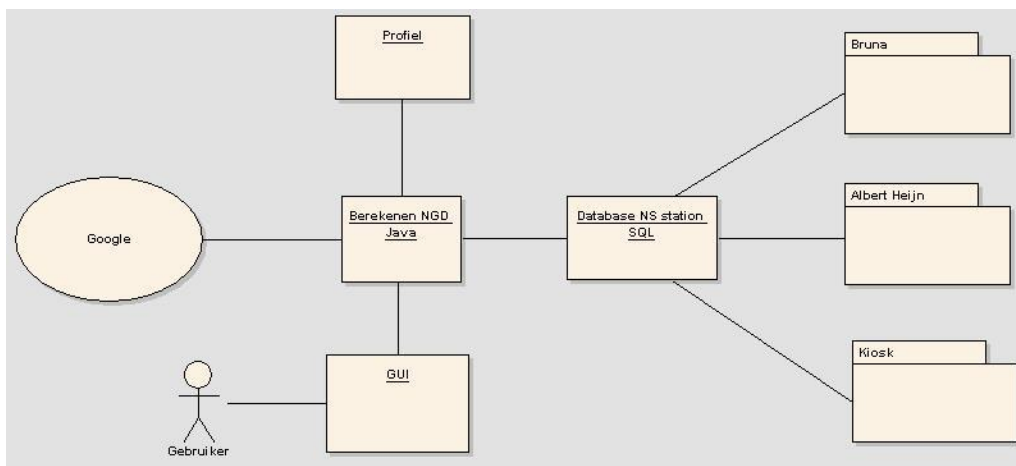
Tabel 1 Overwegingen conventionele technieken/NGD

Door middel van de NGD wordt er bepaald dat er een relatie is tussen twee woorden, er wordt alleen niet bepaald wat deze relatie is. Hierdoor valt de technologie voor ontologie weg. De reisassistent zal werken met behulp van de NGD die op basis van het profiel, tijd en de mogelijkheden op en rond het station een advies zal maken. In het profiel van de gebruiker staan een aantal gegevens over hem, zoals zijn hobby's. Over het station zijn gegevens beschikbaar als de aanwezige winkels en welke artikelen daar verkrijgbaar zijn. Deze twee onderdelen worden met elkaar vergeleken door de NGD. Op welk station de gebruiker zich bevindt wordt bepaald met behulp van GPS. De tijd is een constante factor en eenvoudig te bepalen. De NGD heeft een aantal voordelen tegenover de 'traditionele' manier om semantiek te bereiken. Google kent vrijwel alle woorden die bekend zijn bij de mensheid. Hierdoor kan in het profiel ieder zijn persoonlijke hobby invoeren. Indien er voor de traditionele manier was gekozen hadden deze hobby's eerst gedefinieerd moeten worden in een database. Een ander voordeel van de NGD is dat voor de ontworpen reisassistent de relatie tussen de twee woorden niet belangrijk is. Het is niet nodig om te weten welke relatie er is tussen de woorden als maar bekend is dat er een relatie is. In een traditionele oplossing zou het wel noodzakelijk zijn om aan te geven wat voor relatie er tussen de woorden is. Als laatste bied Google ook een database die altijd up to date is. Omdat Google in theorie user generated content is, is er geen noodzaak om een database up to date te houden.

Om de reisassistent te maken is er gekozen voor de programmeertaal Java. Java wordt ondersteund door meerdere programma's en kan op verschillende platformen draaien. In Java kan de functionaliteit voor het uitrekenen van de NGD verwerkt worden. Het is ook mogelijk een koppeling te maken naar een database waarin het profiel staat opgeslagen en de gegevens van de stations. Als laatste kan Java de interface ondersteunen waarmee de

gebruiker in aanraking komt wanneer hij met de reisassistent werkt. Tevens is het met Java mogelijk om eerst een versie te ontwerpen die op een laptop draait en deze hierna om te zetten naar een smartphone.

In eerste instantie zal de reisassistent draaien op een laptop om er voor te zorgen dat de zakelijke gebruiker tijdens zijn werk in de trein gebruik kan maken van de reisassistent. Indien dit hier een succes is zal de reisassistent worden uitgebreid met een mogelijkheid op de smartphones. Op de laptop zal de reisassistent worden uitgebracht in de vorm van een gadget of widget waarna hij eenvoudig te converteren is naar een smartphone applicatie.



Afbeelding 10 Eerste schets reisassistent

In afbeelding 10 is te zien hoe deze onderdelen samen zouden komen in een reisassistent. De NGD die in Java wordt uitgerekend staat centraal waaraan alle andere onderdelen zijn gekoppeld. Via Google wordt de NGD uitgerekend en met de GUI kan de gebruiker de reisassistent bedienen. Het profiel en de station database geven samen de informatie die de NGD nodig heeft om te functioneren.





## 4 Case beschrijving van reisassistent

Deze Case beschrijft hoe een normale dag met de reisassistent eruit zou kunnen zien. Er wordt beschreven hoe één van de personas gebruik zou maken van de functionaliteiten die de reisassistent biedt en hoe de dag van de persona wordt beïnvloed door de reisassistent.

In deze Case wordt uitgebreid beschreven hoe de reisassistent in zijn werk moet gaan. Gezegd moet worden dat deze Case niet beschrijft hoe het proof of concept eruit zal zien. Het proof of concept is een manier om te bewijzen dat de techniek werkt en zal niet alle functies omvatten die hieronder beschreven worden. Indien er voor wordt gekozen om de reisassistent wel helemaal uit te werken zullen deze functies gerealiseerd worden.

### 4.1 Een dag in het leven

In de personas wordt Lisa als zakenvrouw beschreven en reist voor haar werk door heel Nederland. Ze heeft twee afspraken op een dag en om daar te komen zal ze gebruik maken van de trein. Hierbij maakt zij gebruik van de reisassistent tijdens haar reis en de in wachttijd die ze tussen de aansluiting heeft.

#### 9 uur richting Nijmegen

Lisa heeft om half elf in Nijmegen een afspraak met een klant. Zelf woont zij in Rotterdam en moet ze een uur en drie kwartier reizen voordat ze in Nijmegen is. De reis heeft ze gepland met behulp van de reisassistent en komt er zo achter dat ze de trein van kwart voor negen moet hebben. De reis van Rotterdam naar Utrecht, waar ze moet overstappen duurt drie kwartier. In deze tijd kan ze eenvoudig werken met haar laptop in de trein. In de trein is internet beschikbaar (dit wordt in 2010 beschikbaar (31) (32)) waardoor ze gewoon kan mailen en de reisassistent kan gebruiken.

Voordat ze aankomt, ziet ze dat de wachttijd tot haar volgende aansluiting tien minuten bedraagt. Vijf minuten voordat Lisa aankomt op Utrecht centraal geeft de reisassistent haar een waarschuwing dat ze binnenkort aankomt. De reisassistent geeft tevens aan welke activiteiten Lisa kan doen in de tien minuten die ze moet wachten. Zo kan ze een kopje koffie gaan halen of een krant gaan halen bij de Bruna. Bij Lisa komt koffie halen bovenaan te staan in haar lijst voor adviezen. Het is namelijk nog relatief vroeg in de ochtend, ze heeft niet veel wachttijd en Lisa heeft al eerder gekozen voor het halen van een kopje koffie. Tevens staat in haar profiel aangegeven dat ze van koffie houdt, deze factoren worden meegenomen in het advies.

De reisassistent onthoudt welke keuzes een gebruiker eerder heeft gemaakt en wanneer er dan weer adviezen worden getoond weegt het eerder gekozen advies zwaarder dan andere adviezen. Lisa kiest er voor om een koffie te halen in de tien minuten die ze heeft op Utrecht centraal. Via het advies krijgt ze te zien welke looproute ze moet nemen om naar de dichtstbijzijnde plaats voor koffie te komen. Na het koffie halen loopt ze naar haar aansluiting en gaat ze op weg naar Nijmegen.

#### 12 uur richting Dordrecht

Lisa is inmiddels onderweg naar haar tweede afspraak van de dag. Haar tweede afspraak is in Dordrecht waarvoor ze via Tilburg moet reizen. In de trein kan ze op haar laptop haar aantekeningen van haar eerste afspraak verwerken. Tijdens haar treinreis richting Tilburg krijgt Lisa de melding via de reisassistent dat de trein naar Dordrecht een vertraging heeft van 10 minuten. Helaas zijn er geen alternatieve reisopties beschikbaar die haar sneller in Dordrecht brengen dus besluit Lisa om te wachten. Hierdoor komt haar totale wachttijd bij Tilburg op een half uur.

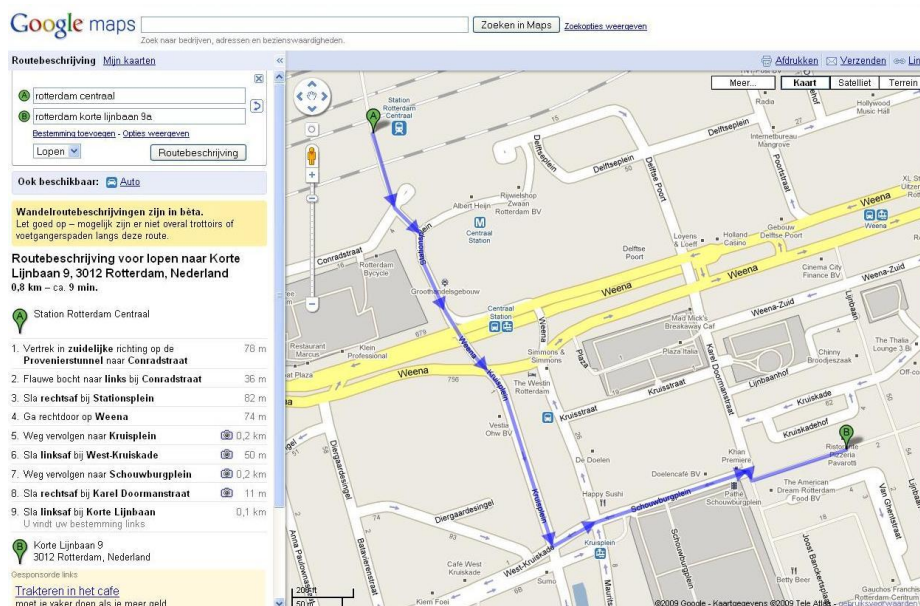
Ze komt om één uur aan in Tilburg en krijgt van de reisassistent een aantal adviezen om haar half uur te vullen. Het is tijd om te eten en hierop geeft de reisassistent ook advies. Tevens heeft Lisa aangegeven in haar profiel dat ze als hobby mode en squash heeft. Ze kan er voor kiezen om een broodje te gaan halen bij de C'est du pain of kan ze een blaadje over squash halen bij de Bruna. Lisa krijgt ook de optie om te kijken waar een Wi-Fi hotspot op het station is, daar kan ze als ze wil nog even verder werken.

#### 5 uur richting Rotterdam

Na haar tweede afspraak gaat Lisa terug naar huis en neemt ze de trein naar Rotterdam. Eenmaal in de buurt van Rotterdam geeft de reisassistent nog een aantal adviezen aan haar. Deze adviezen betreffen twee mogelijkheden om haar tijd te vullen rond op Rotterdam. Ze kan ook direct naar huis gaan waardoor de reisassistent haar route plant naar huis waarbij rekening wordt gehouden met haar favoriete natransport. Natransport is het vervoermiddel wat haar na de trein op haar plaats van bestemming krijgt. Dit kan bijvoorbeeld tram, metro of bus zijn. De adviezen die ze krijgt zijn om iets te gaan eten bij Pavarotti de dichtstbijzijnde Italiaan of nog een film te pakken bij Pathé Schouwburgplein.

Lisa kiest ervoor om nog even wat te gaan eten bij Pavarotti. De reisassistent laat haar via een looproute zien hoe ze er moet komen. Voor de looproute naar bestemming buiten het station kan gebruik gemaakt worden van Google maps zoals te zien in afbeelding 11. Ze eet nog wat lekkers in het restaurant en gaat daarna naar huis.





Afbeelding 11 Google maps

## 4.2 Onderdelen van reisassistent

In de bovenstaande tekst staat er beschreven hoe een normale werkdag eruit zou zien met behulp van de reisassistent. De reisassistent heeft een aantal functionaliteiten die in deze tekst naar voren komen en die samen de reisassistent vormen. De volgende functionaliteiten bevat de reisassistent:

1. Profiel in kunnen vullen
2. Je reis plannen
3. Het melden van vertragingen en een alternatieve route aangeboden krijgen
4. Adviezen krijgen om je wachttijd te vullen
5. Looproute kunnen ontvangen naar activiteit of aansluiting

De hierboven beschreven functionaliteiten zijn niet allemaal nieuw of innoverend. Het plannen van je reis bestaat al enige tijd in de vorm van 9292OV.nl of de reisplanner Xtra van de NS. Tevens zijn er twee andere opties waarvoor andere projecten draaien binnen Logica. Een onderzoek naar het melden van vertragingen en het weergeven van een looproute wordt al gemaakt door andere leden Logica.

Om deze redenen wordt in de rest van dit verslag zijnde het functioneel en technisch ontwerp alleen het invullen van het profiel en het krijgen van adviezen volledig uitgewerkt. Indien alle projecten goed verlopen en er vraag is naar een volledige reisassistent kunnen de verschillende onderdelen worden samengevoegd om een volledige reisassistent te maken zoals hierboven wordt beschreven.

## 4.3 De adviezen

De adviezen generen en het profiel vormen de belangrijkste elementen van de rest van dit verslag. De adviezen die de gebruiker ontvangt worden, zoals al eerder omschreven op een semantische wijze samengesteld. Maar er is nog geen antwoord gegeven op de vraag hoe de gebruiker de adviezen te zien krijgt. Hier zijn namelijk een aantal variaties in en gradaties van detailleringen.

### 4.3.1 Variaties

Hoe de adviezen aan de gebruiker getoond worden kan twee kanten opgaan. De adviezen kunnen getoond worden uit een gebruikersperspectief, uit een bedrijfsperspectief en een combinatie van beide. Hierin zit een verschil van belangen waardoor de uiting van informatie verschilt.

#### Gebruikersperspectief

Degene die de reisassistent gebruikt wil natuurlijk het beste advies krijgen, de beste koffie voor de laagste prijs of de koffie die het dichtbij is tot het punt waar hij nu staat. Als men uit dit perspectief gaat kijken moet een advies niet gebonden zijn tot bepaalde merken of winkels. Dan moet de reisassistent meningsloos zijn tot de gebruiker anders aan geeft. Dit perspectief dient alleen de belangen gebruiker en hoe hij informatie kan halen uit de adviezen.

#### Bedrijfsperspectief

De bedrijven zijn de winkels en merken op en rond het station en de bedrijven zullen in een reisassistent die een manier zien om hun producten aan de man te brengen. Als de reisassistent een voorkeur heeft voor de koffie van de Albert Heijn To Go (AHTG) boven die van de Kiosk kan dit leiden tot meer verkoop van AHTG koffie. De gebruiker kan tevens aanbiedingen ontvangen van een AHTG op basis van zijn voorkeuren en eerder gekozen adviezen. Het rekening houden met eerder gemaakte keuzes is een vorm van Contextual Marketing (33) waar bedrijven gebruik van zouden kunnen maken. Er is ook de mogelijkheid om gebruikers te confronteren met aanbiedingen die niet gebaseerd zijn op de voorkeuren van de gebruiker en dit is een vorm van direct marketing (34).

Er is ook nog mogelijk om gebruikers producten aan te bieden die een associatie hebben met producten die ze al gekozen hebben, dit is een vorm van affiliate marketing (35). Denk hierbij aan een gebruiker die een kopje koffie wil gaan halen bij de AHTG en ook een aanbieding krijgt voor koffie leutjes van Peijnenburg in reisverpakking. Hier kan de NGD een rol spelen in samenwerking met bijvoorbeeld beschikbare aanbiedingen.

Bij het bedrijfsperspectief is het nog steeds van belang dat de gegevens adviezen worden gedaan op basis van de voorkeuren van de gebruiker. Het is mogelijk om verschillende marketing technieken toe te passen binnen de reisassistent maar er zal op gelet moeten worden dat deze niet de overhand krijgen in de reisassistent waardoor de gebruiker geen vertrouwen meer heeft in het advies.

#### Combinatie

Het is ook mogelijk om een combinatie te maken van het gebruikersperspectief en het bedrijfsperspectief. Een goedvoorbeeld hiervan zijn de "sponsored link" (36). Een voorbeeld van sponsored links is te zien bij Google nadat een zoekopdracht heeft uitgevoerd. In de resultaten zitten links die gesponsord zijn door bedrijven duidelijk te onderscheiden van 'gewone' resultaten. Een dergelijke opstelling kan ook gebruikt worden in de reisassistent, waarbij gesponsorde adviezen op een andere manier worden weergegeven dan 'gewone' adviezen.

Voor de eerste uitwerking van de reisassistent is er gekozen voor het gebruikersperspectief. Hiermee wordt de reisassistent als eerste objectief tegenover de gebruiker en kan de technologie bewezen worden. Als deze technologie goed blijkt te werken kan er een stap gezet worden richting bedrijven die mogelijk interesse zouden hebben in een aandeel in de reisassistent waardoor hun producten eerst zouden komen.

#### 4.3.2 Detaillering

De adviezen die de gebruiker krijgt kunnen in verschillende gradaties van detaillering getoond worden. Een voorbeeld hiervan is de AHTG; Bij de AHTG zijn verschillende artikelen te verkrijgen. Van deze artikelen kan een taxonomie gemaakt worden waardoor ze niet meer per artikel getoond worden maar per categorie. Zo is koffie een categorie waarbij er allerlei soorten koffie beschikbaar zijn in variërende mate van detaillering zoals: Koffie met suiker en melk of koffie met alleen melk of alleen suiker. De reisassistent moet adviezen gaan geven aan de gebruiker over de artikelen die bij hem passen aan de hand van zijn profiel. Maar de vraag is of deze adviezen worden gegeven per artikel of per categorie van artikelen. Hier zitten een aantal overwegingen die belangrijk zijn voor het functioneren van de reisassistent, zowel voor de gebruiker als voor de prestatie van de reisassistent.

#### Gebruiker

Voor de gebruiker is de verschillende mate van detaillering van groot belang omdat bij een lagere mate van detaillering de gebruiker minder specifiek kan zoeken naar de artikelen die hij zou willen. Dit kan ook invloed hebben in de mate waarin de gebruiker tevreden is met het functioneren van de reisassistent. Wanneer een gebruiker naar een specifiek artikel op zoek is en de reisassistent biedt alleen maar categorieën kan dit tot ergernis leiden. Aan de andere kant is het ook aannemelijk dat een gebruiker heeft ingevuld dat zijn favorieten drank melk is. Hiervoor hoeft hij alleen maar te weten waar er zuivel te verkrijgen is, omdat melk één van de meest geconsumeerde zuivel vormen is. Indien de gebruiker een niet al te gebruikelijke vorm van zuivel consumeert, zoals umer, zal hij zelf wel weten waar dit te verkrijgen is.

#### Prestatie

De keuze voor het detaillering niveau van de adviezen heeft ook invloed op de hoeveelheid adviezen die de reisassistent kan generen. Er zijn duizenden verschillende artikelen maar categorieën zijn er een stuk minder. Als een advies per categorie wordt gegeven kan dit het aantal adviezen sterk terug dringen. Dit heeft nadelige gevolgen voor de mate van detaillering zoals hierboven aangegeven maar heeft het volgende voordeel, zoals hieronder aangegeven.

In de onderstaande tabel is af te lezen hoeveel tijd Google gemiddeld nodig heeft om een zoek query terug te geven. Dit getal is berekend uit een negental ingevoerde zoekopdracht. De tijd die hieruit terugkomt kan variëren per gebruiker en computer. Hierdoor is deze gemiddelde tijd alleen een schatting.

De tijd is van invloed op de tijd die de reisassistent nodig heeft om de adviezen uit te rekenen. Als de reisassistent per artikel gaat vergelijken of deze bij het profiel past moeten er duizenden berekeningen gemaakt worden. Aan de hand hiervan kan er een formule worden gemaakt over de benodigde tijd die er nodig zou zijn om de adviezen uit te rekenen over de aantal mogelijke combinaties. Deze berekening is alleen een schatting omdat elke keer dat een zoekquery wordt uitgevoerd in Google een andere tijd terug komt en wanneer een zoekquery twee keer wordt uitgevoerd de zoekterm in de cache staat.

Product/categorie	Tijd nodig van Google in seconden
Zuivelproducten	0.08
Melk	0.39
Kaas	0.25
Koffie	0.11
Espresso	0.26
Koffieverkeerd	0.26
Motor	0.17
Promotor	0.13
Moto73	0.17
Gemiddelde benodigde rekentijd	0.20

Tabel 2 Gemiddelde rekentijd Google

De volgende formule is opgesteld om de rekentijd uit te rekenen:

$$r = a(h + v)0.2$$

Bij deze formule is  $r$  de hoeveelheid rekentijd die nodig is om alle combinaties uit te rekenen.  $A$  staat voor het aantal artikelen,  $h$  voor het aantal hobby's,  $v$  voor de hoeveelheid favorieten voedingswaren en 0.20 voor de gemiddelde tijd die Google nodig heeft.

#### Gebruikte variabelen en aannamen

##### Hobby's en voedingswaren

De hobby's en voedingswaren die gebruikt worden om de NGD uit te rekenen komen uit het profiel die de gebruiker heeft ingevuld. De volgende aannamen worden gedaan voor het uitwerken van de onderstaande voorbeelden: De gemiddelde gebruiker heeft drie hobby's, drie verschillende soorten eten en drinken

##### Artikelen

Artikelen wordt in de rest van dit document gezien als alles wat de reisassistent kan aanbieden aan adviezen. Dit zijn niet alleen de artikelen die beschikbaar zijn in de winkels op het station maar ook de diensten die worden aangeboden op en rond het station zoals een massage, een film of restaurant. Voor de volgende voorbeelden wordt er geschat dat er 50.000 artikelen staan in de database. Laat duidelijk zijn dat het aantal artikelen van 50.000 een schatting is.

Als in het voorbeeld de bovenstaande waardes worden gehanteerd zou er bij elke advies wat gegeven wordt een rekentijd nodig zijn van 9000 minuten. Dit vertaald naar uren zou het betekenen dat voor elke keer dat de gebruiker adviezen wil en de reisassistent alle combinaties gaat uitrekenen er een tijd nodig is van 25 uur. Duidelijk is deze hoeveelheid tijd onacceptabel en zal er een manier uitgevonden moeten worden om de hoeveelheid van de variabelen te verminderen.

Het maken van categorieën zou hier dan het aantal van de variabele 'artikelen' aanzienlijk verminderen. Naar een schatting kunnen de 50.000 artikelen samengevat worden in 10.000 categorieën. Als dit aantal wordt aangehouden voor het aantal artikelen wordt de rekentijd al terug gedrongen naar 300 minuten zijnde 5 uur. Dit is nog steeds te veel maar al aanzienlijk minder dan 25 uur.

Om dit probleem geheel te overbruggen zal er een matrix opgesteld moeten worden waarin de hobby's, voedingswaren en categorieën worden voorberekend. Hiermee wordt er bedoeld dat bijvoorbeeld voetbal, mode en patat wordt berekend met de beschikbare categorieën. De NGD waarden van de combinaties worden opgeslagen in een matrix die door de reisassistent wordt uitgelezen. Dit zal een hoop tijd besparen en een realistische rekentijd geven voor het generen van adviezen. Deze matrix zal regelmatig geüpdate moeten worden

om de NGD waarden actueel te houden en met eventueel nieuwe hobby's of categorieën die gebruikt worden. Een voorbeeld van een matrix is hieronder te zien.

Hobby's & voedingswaar → ↓ Artikelen	Motorrijden	Gamen	Espresso	Broodje gezond
Zuivel	0.966	1.084	1.054	1.132
Koffie	0.644	0.735	0.909	1.120
Motorblad	0.451	0.822	1.238	∞
Bioscoop film	0.974	0.887	1.442	∞
Haar knippen	0.936	1.119	1.413	0.769

Echte NGD waarde

Tabel 3 Matrix voorbeeld 1

Door het maken van een taxonomie over de artikelen en een matrix op te stellen die de NGD waarden van combinaties op slaat kan er een realistische rekentijd worden gerealiseerd. Hoewel de taxonomie wel nadelige effecten heeft voor de detaillering van de adviezen is het een noodzaak voor de prestatie van de reisassistent.

#### 4.4 Basis van advies

De adviezen die de reisassistent zal uitbrengen zijn zoals hierboven gelezen kan worden, gebaseerd op in ieder geval het profiel van de gebruiker en de mogelijkheden op het station. drie andere factoren spelen ook mee in de te geven adviezen. In totaal spelen de volgende factoren mee voor het advies:

- Profiel
- Mogelijkheden op en rond het station
- Huidige tijd
- Locatie
- Eerder gekozen adviezen

Met het profiel en de mogelijkheden op en rond het station wordt er met behulp van de NGD een semantiek tussen de beide factoren vastgesteld waardoor adviezen zijn afgestemd op de gebruiker. Door de factor huidige tijd mee te nemen wordt de reisassistent context-aware en kan de reisassistent bepalen of het ergens wel of geen tijd voor is.

Om de locatie van de gebruiker te bepalen zal er gebruik gemaakt moeten worden van Global Positioning System (GPS). Als de reisassistent verbinding maakt met het internet in de trein kan er gebruik gemaakt worden van de GPS systemen in de trein en hetzelfde geldt voor een hotspot op het station. Indien er een versie gecreëerd wordt die functioneert op een smartphone zal deze telefoon over een GPS functie moeten beschikken.

Door de eerder gekozen adviezen van de gebruiker te registreren wordt het mogelijk om de voorkeuren van gebruikers te gebruiken in de adviezen. Op het mobiele apparaat komt een tabel waarin de eerder gemaakte keuzes opgeslagen worden. Hierdoor komt een eerder gekozen advies sneller terug bij de gebruiker. Tevens kan deze functie gebruikt worden voor het contextual marketing dat geïmplementeerd kan worden.

#### 4.5 Toegevoegde waarden

Om een bestaansrecht te hebben moet de reisassistent een toegevoegde waarden bieden voor de reizigers. In de onderstaande tekst staat beschreven welke functionaliteiten de reisassistent moet bieden die momenteel nog niet beschikbaar zijn. Eerst komen de voordelen voor de gebruiker en hierna voor bedrijven.

##### 4.5.1 Gebruiker

###### Advies geven

Dit is de basis van de reisassistent die in dit verslag wordt uitgewerkt maar is ook hetgeen dat nog niet wordt aangeboden op een persoonlijk niveau. Er zijn applicaties die je adviseren over mogelijkheden in de buurt maar geen die daarbij rekening houden met de persoonlijke voorkeur. Tevens heeft de reisassistent een registratie functionaliteit waarmee naar mate de reisassistent vaker gebruikt wordt steeds meer de persoonlijke voorkeuren van de gebruiker 'leert' kennen.

###### Alle mogelijke hobby en voedingswaren kennen

Doordat de reisassistent gebruik maakt van Google is er geen hobby of voedingswaar die hij niet kent. Indien er gebruik gemaakt zou worden van gebruikelijke semantische technologieën zouden deze eerst gedefinieerd moeten worden met de kans dat er hobby's of voedingswaar weggelaten worden. Tevens is Google altijd up to date waardoor er nooit gebruikers buiten de boot vallen.

#### Meenemen van omstandigheden

De reisassistent is context-aware waardoor hij rekening houdt met de huidige tijd en plaats waar de gebruiker zich bevindt. Door deze functionaliteit worden de adviezen toegespitst op de specifieke situatie van de gebruiker. Hierdoor zal de gebruiker bijvoorbeeld nooit om 7 uur 's ochtends te horen krijgen dat hij ergens een patatje kan halen.

#### Adviezen per station

Doordat de reisassistent weet welke artikelen er per station beschikbaar zijn zal de gebruiker nooit een advies krijgen wat niet uit te voeren is op het station waar hij zich bevindt. Hierdoor wordt het advies betrouwbaar voor de gebruiker en kan hij er vanuit gaan dat het advies altijd uitvoerbaar is.

De hierboven genoemde punten vormen samen de mogelijkheden die de reisassistent aan de gebruiker te bieden heeft en op welke manie dit gerealiseerd zal worden. De reisassistent kan er voor zorgen dat de gebruiker altijd een passend advies heeft voor zijn situatie en dit op basis van zijn persoonlijke voorkeuren. Tevens zal de reisassistent er altijd voor zorgen dat het advies voor de gebruiker uitvoerbaar is.

#### 4.5.2 Bedrijven

##### Nieuwe manier van marketing

Voor bedrijven schuilt er in de reisassistent een nieuwe manier van marketing. Door bijvoorbeeld contextueel marketing of affiliate marketing kunnen bedrijf meer specifiek op de klant gerichte producten aanbieden waardoor de klant minder het gevoel heeft van reclame.

##### Adviezen kunnen eenzijdig worden

Ethische gezien zouden de adviezen altijd objectief moeten zijn maar er is de mogelijkheid om de adviezen specifiek naar één of meerdere bedrijven toe te maken. Hierdoor gaan reizigers meer gebruik maken van de winkels die door de reisassistent worden aangegeven.

Voor bedrijven wordt het mogelijk met de reisassistent om de klant op een nieuwe manier te benaderen die de klant niet als storend kan ervaren. Tevens wordt het mogelijk om de klanten te leiden naar producten van een leverancier waardoor deze meer zal verkopen.

#### **4.6 Medium reisassistent**

De keuze voor het medium waarop de reisassistent zal draaien kan twee kanten op. Het is mogelijk om de reisassistent te draaien op een laptop maar ook op een smartphone. Met deze keuzes voor een laptop of smartphone komen een aantal overwegingen. Deze worden hieronder beschreven:

##### Laptop

In eerste instantie zal de reisassistent draaien op een laptop. Een laptop past bij het profiel van de zakelijke reiziger die in de trein een lange reis maakt en hier gewoon verder wil werken. Tevens biedt een laptop alle mogelijkheid om een applicatie te maken met verschillende functies. Zo heeft een laptop meer rekenkracht dan een smartphone en kan een laptop meer gegevens opslaan. Om een GPS (zoals beschreven in de Case) signaal te ontvangen zal de reisassistent op de laptop gebruikt maken van de GPS in de betreffende trein.

Het nadeel van een laptop is dat deze minder mobiel is dan een smartphone. Zo is het mogelijk om het advies en andere functies op de laptop te laten functioneren even als de looproute. Maar als de looproute in de vorm een kaart wordt getoond is het onwaarschijnlijk dat je met een open laptop door het station zal lopen. Hierdoor zal er op een laptop de looproute onthouden moeten worden.

##### Smartphone

Als de reisassistent aanslaat op de laptop en hij blijkt adequaat te functioneren kan er een vertaalslag naar de smartphone gemaakt worden. Omdat de reisassistent in Java wordt geprogrammeerd, moet deze vertaalslag relatief eenvoudig zijn. De voordelen van de smartphone zijn in dit geval dat hij meer mobiel is dan de laptop. Hierdoor kan een looproute op kaart dus wel mee naar buiten genomen worden op het station. Tevens bezit de smartphone meestal een GPS chip waardoor hij zijn eigen positie kan bepalen. Dit kan dus helpen om een looproute te bepalen en eventueel de positie bij te houden wanneer er gelopen wordt.

Een eigenschap van de smartphone is dat ze een stuk kleiner zijn dan een laptop. De zakelijke reiziger valt in een leeftijdscategorie waarin een grotere interface misschien gewenst is. Tevens bezit een smartphone minder rekenkracht dan een laptop en is de internetverbinding over het algemeen langzamer via het 3G netwerk.

##### iPad

Als gulden middenweg tussen de twee mogelijke mediums is er de iPad van Apple. Dit is noch een laptop noch een smartphone. Deze zou wel draagbaar zijn in de trein en op het station maar heeft geen 3G verbinding in Europa. Dit mogelijk medium is in de rest van dit rapport buiten beschouwing gelaten.

#### 4.7 Conclusie

In de Case wordt omschreven welke mogelijkheden de reisassistent biedt aan een persona. Hiermee worden de in het onderzoek gevonden reiaspecten vertaald naar functionaliteiten in de reisassistent. Tevens wordt in dit hoofdstuk behandeld welke praktische problemen er zijn met betrekking tot de reisassistent en welke oplossing hiervoor is gevonden. Dit betreft bijvoorbeeld de rekentijd waarvoor uiteindelijk is gekozen om de adviezen in categorieën te tonen met een als extra hulp een matrix. Een ander voorbeeld is de toegevoegde waarde van de reisassistent die onder andere zit in het adviseren van de gebruiker op persoonlijk niveau en dat bedrijven een nieuwe manier hebben voor marketing.



## 5 Het Functioneel ontwerp

In de vorige hoofdstukken is bepaald welke reisaspecten de reisassistent moet tonen aan de gebruiker en hoe dit vertaalt naar functionaliteiten. In dit hoofdstuk wordt beschreven hoe de functionaliteiten interacteren met de gebruiker. Als eerst wordt wat Use Cases zijn en hoe deze gebruikt zijn om de functionaliteiten uit te werken. Hierna wordt beschreven hoe de gebruiker zal interacteren met de reisassistent. Hierna volgt een Basis, Comfort, Luxe kwalificatie (wordt beschreven in betreffende paragraaf) die beschrijft welke component van de reisassistent erin moeten en welk optioneel zijn.

### 5.1 Use Cases

Om de functionaliteiten in kaart te brengen is er gebruik gemaakt van Use Cases. Deze techniek is onderdeel van de RUP methode om een software applicatie te ontwikkelen. De User Cases beschrijven de reisassistent uit een gebruikersperspectief waarbij de Use Case Diagram (UCD) de interactie schematisch weergeeft en de tekstuele beschrijving van de Use Cases verdere uitleg geeft over de interactie. Tevens wordt er per functionaliteit beschreven welk doel de functionaliteit zal volbrengen (37).

In het algemeen zijn Use Cases te gebruiken bij het beschrijven van een reisassistent die werkt met behulp van een Pull strategie. Hierbij is de gebruiker degene die voor de trigger zorgt waardoor het systeem in werking treedt. In het geval van deze reisassistent gaat het voornamelijk om een Push systeem. Hierbij is de reisassistent degene die ervoor zorgt dat er een actie in gang treedt. Als de gebruiker in de buurt van een bestemming komt zal de reisassistent adviezen gaan genereren zonder dat de gebruiker daartoe opdracht hoeft te geven.

De Use Cases zijn in drie delen beschreven, eerst worden ze schematisch weer gegeven in een diagram, hierna volgt een tekstuele beschrijving wat er met de Use Case bereikt kan worden en tenslotte volgt een stap voor stap beschrijving van actie en reactie tussen gebruiker en reisassistent. De beschrijving van de Use Case is gebaseerd op het boek: Writing Effective Use Cases geschreven door Alistair Cockburn (38). Er wordt onder andere beschreven wie de primaire actor is en welke stappen leiden tot een succesvolle uitvoering van de Use Case.

### 5.2 Functionaliteiten naar Use Cases

Tijdens een reis kan de gebruiker informatie willen over verschillende aspecten van zijn reis. Om deze informatie duidelijk in kaart te brengen zijn er een vijftal functionaliteiten ontworpen waarin de reisinformatie opgedeeld zal worden. Deze vijf functionaliteiten zijn de volgende:

#### Functionaliteiten

1. Profiel in kunnen vullen
2. Je reis plannen
3. Het melden van vertragingen en een alternatieve route aangeboden krijgen
4. Adviezen krijgen om je wachttijd te vullen
5. Looproute kunnen ontvangen naar activiteit of aansluiting

Om deze functionaliteiten verder uit te werken naar een werkend systeem zijn de volgende Use Cases opgesteld:

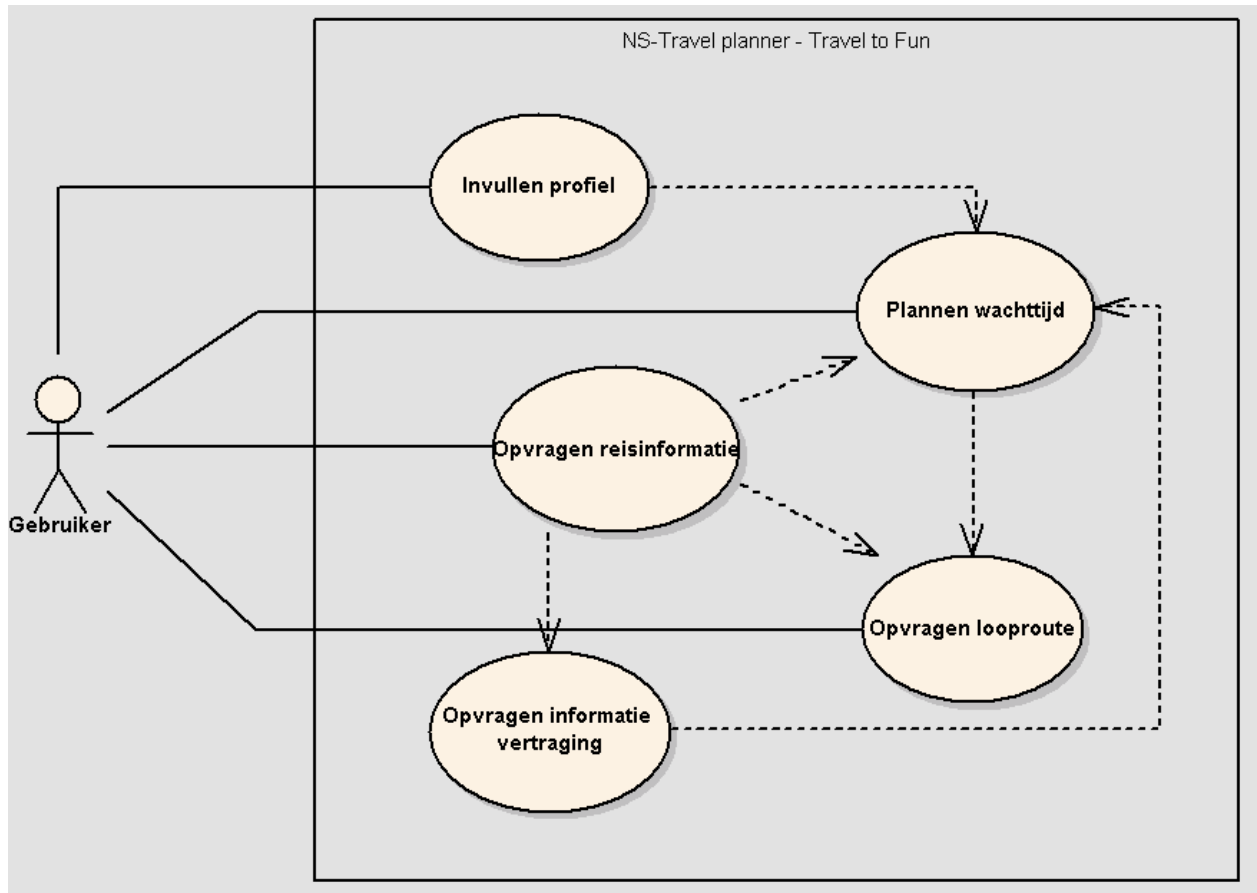
#### Use Cases

1. Invullen profiel
2. Opvragen reisinformatie
3. Opvragen informatie vertraging
4. Plannen wachttijd
5. Opvragen looproute



### 5.3 Use Case Diagram

In de USD, afbeelding 12, wordt schematisch weergegeven hoe de functionaliteiten van de reisassistent samen gaan werken met de gebruiker.



Afbeelding 12 Use Case diagram

In afbeelding 12 zijn alle Use Cases te zien die de reisassistent zal bevatten. Zoals al eerder vermeld in de Case zal in de rest van dit document alleen de Use Cases 'Invullen Profiel' en 'Plannen wachttijd' uitgewerkt worden.

## 5.4 Invullen profiel

Tijdens de wachttijd of op verzoek van de gebruiker zal de reisassistent advies aanbieden over de activiteiten die op en rond het station te doen zijn. Om dit advies adequaat te formuleren is er informatie nodig over de gebruiker. Deze informatie zal de gebruiker invullen in een persoonlijk profiel. De informatie in het profiel zal gekoppeld worden aan de informatie die bekend is over het station om op semantische wijze een passend advies te geven. In dit profiel staat de basis informatie over de gebruiker. Deze informatie zal zijn:

- Geslacht
- Leeftijd
- Hobby's
- Voedingswaren
- Favoriete natransport

Omdat in het huidige project er alleen een proof of concept gemaakt zal worden, zullen de keuze mogelijkheden beperkt zijn tot een drietal keuzemogelijkheden per opties. In het technisch ontwerp zal beschreven worden hoe de ingevulde informatie hier gebruikt zal worden om een advies te geven aan de gebruiker in combinatie met station gegevens.

Use Case: Invullen Profiel	
<b>Primary Actor</b>	Gebruiker
<b>Goal in Context</b>	Het invullen van het profiel dat later gebruikt kan worden door de reisassistent
<b>Scope</b>	Profiel van reisassistent
<b>Level</b>	High level voor het project
<b>Stakeholders and Interests</b>	Gebruiker: Wil een advies aan de hand van voorkeuren Station: Bedrijven op en rond station hebben voordeel in adviezen voor hun winkel
<b>Precondition</b>	Geen
<b>Minimal Guarantees</b>	De gebruiker heeft zijn hobby's, eten, drinken, geslacht en leeftijd ingevuld
<b>Success Guarantees</b>	De gebruiker heeft zijn hobby's, eten, drinken, geslacht, leeftijd en natransport ingevuld
<b>Trigger</b>	Gebruiker start voor het eerst de applicatie op of gebruiker wil opties wijzigen
<b>Main success Scenario</b>	1. Gebruiker: Start reisassistent op voor de eerste keer 2. Reisassistent: Vraagt voor het invullen van het profiel Gebruiker: klikt op OK 3. Reisassistent: Vraagt invullen geslacht 4. Gebruiker: Kiest geslacht 5. Reisassistent: Vraagt leeftijd van gebruiker 6. Gebruiker: Vult leeftijd in 7. Reisassistent: Vraagt invullen hobby's 8. Gebruiker: Kiest max. drie hobby's 9. Reisassistent: Vraagt invullen favorieten voedingswaren 10. Gebruiker: Kiest max. drie soorten favorieten voedingswaren 11. Reisassistent: Vraagt favorieten natransport 12. Gebruiker: Kiest natransport 13. Reisassistent: Geeft scherm met ingevulde opties en vraagt om bevestiging 14. Gebruiker: Bevestigt 15. Reisassistent: Geeft melding opgeslagen
<b>Extensions(wat kan er anders gaan)</b>	1a. Gebruiker: klikt op bevestig zonder dat alle vereiste opties zijn ingevuld 1b. Reisassistent: geeft melding met de nog in te vullen opties 1c. Gebruiker: vult opties alsnog in 1d. Reisassistent: Geeft scherm met ingevulde opties en vraagt om bevestiging 2a. Gebruiker: Vult te veel hobby's, voedingswaren in(max 3, indien meer melding) 2b. Reisassistent: Geeft melding 'te veel ingevuld bij hobby's of voedingswaren'

Tabel 4 Uitwerking Profiel invullen

## 5.5 Plannen wachttijd

De wachttijd die een gebruiker moet spenderen op een station kan gebruikt worden voor verschillende activiteiten. De reisassistent zal de gebruiker adviseren over de activiteiten die hij kan doen in de wachttijd die hij spendeert op het station. Tevens kan de reisassistent activiteiten adviseren voor langere tijd buiten het station.

Plannen wachttijd moet de volgende acties kunnen uitvoeren:

- Activiteit adviseren
- Activiteit kiezen

Daarbij horende informatie

- Hoeveel tijd kosten de activiteiten
- Locatie activiteit
- Looproute naar activiteit (doorverwijzing naar looproute)

Wanneer de gebruiker een activiteit heeft gekozen zal de reisassistent de route daarna toe aangeven hoe hij daar moet komen. Voor deze functionaliteit wordt gebruik gemaakt van de Use Case 'aanvragen looproute', dit wordt verder in dit rapport niet behandeld. De gekozen activiteiten worden onthouden waardoor bij een volgende keer de eerder gekozen activiteiten bovenaan komen te staan. In het Proof of Concept zal deze functie nog niet gerealiseerd worden.

Use Case: Plannen wachttijd	
<b>Primary Actor</b>	Gebruiker
<b>Goal in Context</b>	Het verkrijgen van advies waarmee de wachttijd gevuld kan worden.
<b>Scope</b>	Het geven van advies aan de gebruiker
<b>Level</b>	High level voor het project
<b>Stakeholders and Interests</b>	Gebruiker: Wil een advies aan de hand van voorkeuren Station: Bedrijven op en rond station hebben voordeel in adviezen voor hun winkel
<b>Precondition</b>	<ul style="list-style-type: none"> <li>• Gebruiker moet <u>het profiel</u> hebben ingevuld</li> <li>• Er moet verbinding zijn met reisassistent server</li> </ul>
<b>Minimal Guarantees</b>	Gebruiker heeft een aantal adviezen ontvangen
<b>Success Guarantees</b>	Gebruiker heeft een aantal adviezen ontvangen, kiest een advies wat hij prefereert, advies wordt uitgespecificeerd met beschikbare informatie.
<b>Trigger</b>	Gebruiker komt in de buurt van plaats waar adviezen gebruikt kunnen worden.
<b>Main success Scenario</b>	<ol style="list-style-type: none"> <li>1. Reisassistent: Detecteert dat gebruiker in de buurt van bestemming/tussenstop komt.</li> <li>2. Reisassistent: Genereert voor gebruiker top drie adviezen</li> <li>3. Gebruiker: Kiest het advies dat hij prefereert.</li> <li>4. Reisassistent: Toont beschikbare informatie over het gekozen advies</li> </ol>
<b>Extensions(wat kan er anders gaan)</b>	<ol style="list-style-type: none"> <li>1a. Reisassistent: geeft de top drie adviezen aan de gebruiker</li> <li>1b. Gebruiker: is niet tevreden met de adviezen en wil meer ontvangen. Klik op de knop 'advies'</li> <li>1c. Reisassistent: genereert nieuwe scherm met meer adviezen waar de gebruiker doorheen kan scrollen</li> <li>1d. Gebruiker: kiest het advies uit die hij prefereert.</li> <li>1e. Toont beschikbare informatie over het gekozen advies</li> <li>2a. Reisassistent: geeft de top drie adviezen aan de gebruiker</li> <li>2b. Gebruiker: wil advies over een specifieke categorie producten of activiteiten. Klik op 'advies'.</li> <li>2c. Reisassistent: genereert nieuwe scherm met meer adviezen waar de gebruiker doorheen kan scrollen</li> <li>2d. Gebruiker: kiest één van de mogelijkheden voor het specificeren van de adviezen.</li> <li>2e. Reisassistent: genereert een lijst met adviezen over de gekozen categorie.</li> <li>2f. Gebruiker: kiest het advies uit die hij prefereert.</li> <li>2g. Toont beschikbare informatie over het gekozen advies</li> <li>3a. Reisassistent: geeft de top drie adviezen aan de gebruiker</li> <li>3b. Gebruiker: wil adviezen ontvangen over de activiteiten buiten het station. Klik op 'advies'</li> <li>3c. Reisassistent: genereert nieuw scherm met meer adviezen waar de gebruiker doorheen kan scrollen</li> <li>3d. Gebruiker: klik op 'omliggend station'</li> <li>3e. Reisassistent: genereert adviezen voor rond het station.</li> <li>3f. Gebruiker: kiest het advies uit die hij prefereert.</li> <li>3g. Toont beschikbare informatie over het gekozen advies, klik op 'advies'</li> </ol>

	<p>3c. <i>Reisassistent</i>: genereert nieuw scherm met meer adviezen waar de gebruiker doorheen kan scrollen</p> <p>3d. <i>Gebruiker</i>: klikt op 'omliggend station'</p> <p>3e. <i>Reisassistent</i>: genereert adviezen voor rond het station.</p> <p>3f. <i>Gebruiker</i>: kiest het advies uit die hij prefereert.</p>
--	--

Tabel 5 Uitwerking Plannen wachttijd

### 5.6 Basis Comfort Luxe Kwalificatie

De hierboven beschreven Use Cases geven aan wat de reisassistent allemaal moet gaan doen. Om aan te geven welke onderdelen van de reisassistent prioriteit hebben is er gebruik gemaakt van de Basis, Comfort, Luxe kwalificatie (BCL). Bij deze kwalificatie zijn de onderdelen die het belangrijkste zijn de Basis, meestal 80% van het project. Dan is er Comfort, hetgeen wat wenselijk is voor het project, meestal de resterende 20% van het project. Als laatste is er dan Luxe en dit vormt de extra's die worden toegevoegd aan het project.

Tijdens dit project worden er twee BCL kwalificaties behandeld. De eerste BCL is voor de reisassistent zoals deze moet worden wil deze geheel functioneren met de Use Cases "Invullen profiel" en "opvragen advies". De andere BCL is voor het Proof of Concept waarin alleen de techniek van de NGD bewezen moet worden.

Op basis van gesprekken met de opdrachtgever is de BCL kwalificatie opgesteld. Uit deze gesprekken is naar voren gekomen dat de reisassistent als eerste een advies moet kunnen maken op basis van Hobby's en favoriete voedingswaren, hierna als mogelijk, ook nog de informatie geslacht en leeftijd meenemen.

BCL geheel functioneren		
Basis	Use Case Invullen profiel	<ul style="list-style-type: none"> <li>• Mogelijkheid tot het invullen van hobby's</li> <li>• Mogelijkheid tot het invullen van favoriete eten</li> <li>• Mogelijkheid tot het invullen van natransport</li> <li>• Deze informatie opslaan in database</li> <li>• Melding geven die de opgeslagen informatie bevestigt</li> </ul>
	Use Case Plannen wachttijd	<ul style="list-style-type: none"> <li>• Een interface waarin de gebruiker de adviezen kan bekijken</li> <li>• Mogelijkheid tot het bepalen van positie van de gebruiker aan de hand van GPS</li> <li>• De huidige tijd te bepalen</li> <li>• GPS positie en huidige tijd te verwerken als context informatie.</li> <li>• Aanspreken van het opgeslagen profiel</li> <li>• Aanspreken van de matrix</li> <li>• Geven van adviezen over hobby's en favoriete eten samen met beschikbare artikelen aan de hand van de NGD</li> <li>• Mogelijkheid voor de gebruiker om een advies te kiezen</li> <li>• Het gekozen advies in meer detail te laten zien</li> <li>• Mogelijkheid voor de gebruiker om verder te zoeken dan eerste 3 adviezen</li> <li>• Mogelijkheid om adviezen buiten het station apart te zoeken.</li> </ul>
Comfort	Use Case Invullen profiel	<ul style="list-style-type: none"> <li>• Een interface die in de huisstijl is van de NS</li> <li>• Mogelijkheid tot het invullen van geslacht</li> <li>• Mogelijkheid tot het invullen van leeftijd</li> </ul>
	Use Case Plannen wachttijd	<ul style="list-style-type: none"> <li>• Een interface in de huisstijl van de NS</li> <li>• Mogelijkheid om leeftijd en geslacht te verwerken in het advies</li> <li>• Mogelijkheid om op per categorie op adviezen te zoeken</li> </ul>
Luxe	Use Case Invullen profiel	<ul style="list-style-type: none"> <li>• Suggesties voorbeeld Google zoekopdrachtherkenning of T9 herkenning</li> </ul>
	Use Case Plannen wachttijd	<ul style="list-style-type: none"> <li>• Een voorbeeld kaart met daarin hoe de gebruiker moet lopen bij de top drie adviezen</li> <li>• Bij top drie adviezen tonen hoelang ze ongeveer duren om uit te voeren</li> </ul>

Tabel 6 BCL reisassistent

Voor het POC geldt dat het als eerste moet bewijzen dat de NGD techniek werkt. Hierna is het van belang dat aan de hand van de NGD waarde er een advies gegenereerd kan worden. Als luxe zou het POC eventueel nog een interface krijgen in de stijl van de NS om te laten zien dat het om een NS applicatie gaat.

BCL Proof of Concept	
Basis	<ul style="list-style-type: none"> <li>• Database met 100 artikelen</li> <li>• Tekstveld waar gebruiker een hobby of voedingswaarde kan invullen</li> <li>• Tekstveld kunnen uitlezen</li> <li>• Uitgelezen tekst en database kunnen verwerken met NGD</li> <li>• Een NGD waarde kunnen teruggeven</li> </ul>
Comfort	<ul style="list-style-type: none"> <li>• Met behulp van de NGD waarde een advies kunnen geven</li> </ul>
Luxe	<ul style="list-style-type: none"> <li>• Een interface in de huisstijl van de NS</li> </ul>

Tabel 7 BCL POC

## 5.7 Conclusie

In dit hoofdstuk is de informatie uit het onderzoek en de Case vertaald naar een functioneel ontwerp. Dit functioneel ontwerp wordt opgedeeld in Use Cases die de informatie bevatten die een gebruiker nodig heeft. Hiermee wordt niet alleen een functioneel ontwerp gemaakt maar wordt er ook antwoord gegeven op de derde deelvraag. De derde deelvraag is: Hoe vertalen de reisaspecten naar functionaliteiten in de reisassistent?

Uit het onderzoek is gebleken dat de reiziger een behoefte heeft aan informatie tijdens zijn reis. Informatie over de voortgang van de reis maar ook over vertragingen en wat er gedaan kan worden in de wachttijd. In de Use Cases worden deze elementen naar een functionaliteit vertaald waaronder verschillende informatie is verdeeld. De informatie die onder de Use Cases staat is alles wat de gebruiker nodig heeft om zijn reis goed te kunnen overzien. In de onderstaande tabel staan de belangrijkste informatie punten die de doelgroep heeft aangewezen door middel van de enquête en hoe deze informatiepunten zijn vertaald naar functionaliteiten.

Informatie uit enquête	Functionaliteit van de reisassistent	Use Cases
Informatie over vertraging	<ul style="list-style-type: none"> <li>• Je reis plannen</li> <li>• Het melden van vertragingen en een alternatieve route aangeboden krijgen</li> </ul>	<ul style="list-style-type: none"> <li>• Opvragen reisinformatie</li> <li>• Opvragen informatie vertraging</li> </ul>
Willen weten wat er te doen is in wachttijd	<ul style="list-style-type: none"> <li>• Profiel in kunnen vullen</li> <li>• Adviezen krijgen om je wachttijd te vullen</li> </ul>	<ul style="list-style-type: none"> <li>• Invullen profiel</li> <li>• Plannen wachttijd</li> </ul>
Willen weten hoe te lopen op het station	<ul style="list-style-type: none"> <li>• Looproute kunnen ontvangen naar activiteit of aansluiting</li> </ul>	<ul style="list-style-type: none"> <li>• Opvragen looproute</li> </ul>

Tabel 8 Functionaliteiten naar Use Cases





## 6 Het Technisch ontwerp

In het vorige hoofdstuk staat beschreven welke functionaliteiten de reisassistent moet hebben in de vorm van Use Cases. Er wordt beschreven hoe de gebruiker zal werken met de reisassistent en welke informatie daar bij wordt verwerkt. In dit hoofdstuk wordt er antwoord gegeven op de vierde deelvraag: Hoe moeten de functionele eisen op semantische wijzen gaan samenwerken? Na het onderzoek, Case en FO is geconstateerd dat niet de functionele eisen samen gaan werken maar dat één functionele eis en daarvan de onderdelen semantisch gaan samenwerken. Aan de hand van het FO is het TO opgesteld. De Use Cases worden uitgewerkt aan de hand van architectuur en tekstuele ondersteuning.

### 6.1 Onderdelen reisassistent

Als eerste wordt de techniek van de reisassistent beschreven per onderdeel. Er wordt beschreven welke techniek er toegepast moet worden om deze functionaliteiten te realiseren.

#### Opslaan profiel

Het profiel dat de gebruiker invult moet opgeslagen worden om later te kunnen gebruiken voor het advies. Dit profiel kan worden opgeslagen op een centrale server of op het platform waarop de reisassistent draait. In dit geval; is er voor gekozen om het profiel op te slaan op het platform. Door het profiel op te slaan op het platform blijft het altijd beschikbaar aan de gebruiker om te veranderen ook als de gebruiker geen verbinding heeft met internet. Tevens wordt op deze manier die informatie verdeeld over verschillende opslagmedia's waardoor de hoeveelheid informatie wordt verdeeld.

#### Programma taal

De programmeer taal die gebruikt zal worden voor de reisassistent is Java. In deze programmeertaal is het mogelijk om onderdelen te combineren zoals een link naar een database en de berekeningen die nodig zijn voor de NGD te berekenen. Tevens is Java platform onafhankelijk waardoor het programma op Windows, Linux en Mac kan draaien. Op een later stadium van het project is het dan ook mogelijk om de reisassistent makkelijk beschikbaar te maken voor een smartphone door de platform onafhankelijkheid van Java. Wel moet erop gelet worden dat wanneer de reisassistent naar een smartphone wordt omgezet dat sommige smartphones een uigeklede versie van Java gebruiken genaamd Java ME (38). Deze Java versie bevat minder functies dan de standaard versie van Java en heeft daardoor minder mogelijkheden.

#### Database

Een database is gemaakt voor het verwerken van grote hoeveelheden data. De artikelen, stations en winkels van Nederland zullen op een ruime hoeveelheid data uitronnen. Om deze data adequaat te verwerken zal er een relationele database opgezet worden. Hierbij is ook met het volgende rekening gehouden:

Als de gebruiker bijvoorbeeld van een salade houdt kan hij terecht bij de AH To Go(AHTG). Hier verkopen ze verschillende salades. Er zou dus een klasse gemaakt moeten worden AH waarin staat welke artikelen de AHTG allemaal heeft. Het probleem hier is dat niet alle station een AHTG hebben die even groot is. Hierdoor heeft de AHTG op Rotterdam centraal meer dan de AHTG in Breda. Als er een klasse gemaakt zou worden van AH waarin alle producten beschikbare producten staan van de AHTG en deze zou gelden voor alle stations zou de gebruiker in Breda bedrogen uit komen. Daar zullen ze een aantal producten niet hebben. Dit geldt voor vrijwel alle winkels op de stations zoals de Bruna of de Etos.

Door dit probleem moet er een keuze gemaakt worden voor de structuur van de database. Er zijn drie opties voor dit probleem namelijk:

1. Er wordt een basis tabel gemaakt waarin alleen de artikelen staan die in elk filiaal te verkrijgen zijn. Hierdoor kan de gebruiker nooit bedrogen uit komen maar ontstaat het probleem dat het advies minder specifiek is en dat de gebruiker het idee krijgt dat het advies niet volledig is omdat er in bepaalde winkels meer verkrijgbaar is dan het advies kan aangeven
2. Er wordt voor elk station een aparte tabel gemaakt en voor de desbetreffende winkels. Elke AHTG op een station krijgt een tabel gelinkt aan het station waarin het volledige assortiment staat van die winkel. Hierdoor heeft de gebruiker altijd een compleet advies die alle mogelijkheden van de winkel bevat.
3. Dan is er nog de optie om klasse van winkels te bepalen. Hierbij worden de winkels ingedeeld in klasse die beschrijven welke artikelen ze bezitten. Zo kan bijvoorbeeld de AHTG op Rotterdam centraal klasse C zijn met veel artikelen en de AHTG op Breda klasse A met minder artikelen.

Er is voor gekozen voor optie drie. Hierdoor is het mogelijk om de gebruiker optimale service te bieden op elk station in Nederland.

#### Graphical User Interface



De Graphical User Interface (GUI) moet uiteindelijk gemaakt worden in de huisstijl van de NS maar zal voor het Proof of Concept simpel worden. De GUI zal zich manifesteren als een kleine applicatie op het bureaublad van de gebruiker en verplaatst kan worden naar de voorkeuren van de gebruiker. Indien de gebruiker Windows Vista heeft is het mogelijk een Gadget te maken van de reisassistent. Tevens moet deze GUI zo ontworpen zijn dat deze eenvoudig te converteren is naar het gebruik op een smartphone.

#### NGD

De NGD waarden worden berekend met behulp van de gegevens in het profiel en de data die staat in de database over de stations. Deze waarden worden verwerkt in een matrix waarmee de reisassistent kan werken om met een acceptabele rekentijd adviezen te genereren.

Om er voor te zorgen dat deze matrix gevuld wordt met waarden zal er een NGD module gebouwd moeten worden die dit kan volbrengen. Deze module zal alle artikelen moeten ophalen uit de database en alle profielen van de gebruiker. Hierna kan hij de hobby's en voedingswaren van de gebruiker naast de artikelen van de database leggen en de NGD waarde uitrekenen. Deze module moet draaien op een centrale server zodat deze genoeg rekenkracht heeft om deze taken uit te voeren.

#### Matrix

In de matrix worden de waarden opgeslagen die door de NGD worden gegenereerd. Om er voor te zorgen dat deze waarden snel toegankelijk zijn voor de reisassistent. Deze matrix wordt opgeslagen op een centrale server waar alle hobby's, voedingswaren en artikelen in worden verwerkt.

Een dergelijke matrix zal groot worden vanwege het aantal mogelijke combinaties tussen artikelen, hobby's en voedingswaren. Dit kan een probleem veroorzaken voor de toegankelijkheid van de matrix. Er zijn naar schatting 20.000 categorieën (uit de Case) voor alle artikelen. Het aantal mogelijke hobby's en voedingswaren wordt geschat op 300. Wanneer een individuele cel 2 kilobyte kost betekent dat, dat een matrix van 300 kolommen en 20.000 rijen een totaal van 6000000 kilobyte heeft wat vertaald naar 11 Gigabyte.

De oplossing hiervoor is door alleen te zoeken in de hobby's en voedingswaren die van toepassing zijn voor de desbetreffende gebruiker. De gebruiker kan maar drie hobby's en drie voedingswaren invullen waardoor het aantal kolommen waarin de reisassistent moet zoeken wordt teruggedrongen naar zes. De hoeveelheid data waardoor de reisassistent nu moet zoeken is 234 megabyte en dit is een hoeveelheid die wel uitvoerbaar is. Dan is er ook nog de tijd van de dag die dient als filter over de beschikbare combinaties van de matrix. Hierover meer in volgende paragraaf.

#### Context aware verwerking

Context-aware wordt gedefinieerd als een applicatie die rekening houdt met context en aan de hand daarvan zijn inhoud aanpast. Bij de reisassistent bestaat deze context uit twee delen, de positie van de gebruiker en de huidige tijd. Om deze functies te verwerken in een toepasbare oplossing is de volgende structuur ontworpen.

Als eerste wordt de positie van de gebruiker bepaald met behulp van GPS. Voor de toepassing van de reisassistent moet bekend zijn waar de gebruiker zich bevindt. Aan de hand van deze context informatie wordt er bepaald welke tabellen aangesproken moeten worden om de juiste adviezen te genereren voor die plaats. Vervolgens zal er bepaald worden welke tijd van de dag het is om accuraat advies te geven. Dit gebeurt door middel van de interne klok die zich op vrijwel elk elektronisch communicatie apparaat bevindt. Aan de hand van de tijd worden er mogelijkheden voor adviezen in of uitgeschakeld. Zo zal om tien uur in de ochtend het advies voor patat worden uitgeschakeld en voor koffie worden ingeschakeld.

Door aan de hand van bepaalde context te filteren op de te gebruiken informatie wordt er voor gezorgd dat de reisassistent advies kan aanbieden die rekening houdt met de situatie waarin de gebruiker zich bevindt.

## 6.2 Architectuur reisassistent

Uiteindelijk moeten alle componenten die de reisassistent vormen gaan samenwerken om tot een volledige applicatie te komen.

### 6.2.1 Beschrijving

Zoals al aangegeven in de Case zal er wanneer een gebruiker een station nadert een aantal adviezen gegeven worden. Wanneer de gebruiker op vijf minuten afstand is van het station zal de reisassistent zijn werk gaan doen. De trigger voor het proces van adviezen zit in de GPS functie. De GPS bepaald wanneer een gebruiker op de vijf minuten afstand is en waar de gebruiker aan komt. Als voorbeeld wordt er het station van Breda gebruikt.

Wanneer de gebruiker op vijf minuten afstand is zal de reisassistent de tabel aanspreken waarin alle mogelijkheden van Breda staan. In deze tabel staat bijvoorbeeld dat Breda een AHTG en een Burger King heeft. Tevens staat in deze tabel wat er in de omgeving van het station mogelijk is zoals een VVV kantoor of een kapper. Er zijn tabellen waarin de artikelen van de winkels staan beschreven. Dit zal voor de AHTG onder andere zuivel en koffie zijn en voor de kapper is het bijvoorbeeld haar knippen of hoofdhuid massage. Aan de hand van de winkel tabellen kunnen de tabellen opgeroepen worden waarin de details staan van de mogelijkheden zoals het adres van de kapper of de AHTG. Deze tabellen zullen voor de adviezen niet gebruikt worden maar voor het opvragen van de looproute zullen deze wel nodig zijn

De tabellen waarin de artikelen staan worden gebruikt voor het uitrekenen van de NGD samen met de tabellen waarin de profielen staan. Zoals aangegeven in de Case worden de NGD resultaten verwerkt in een matrix zodat deze sneller bereikbaar zijn. In de onderstaande afbeelding is te zien hoe deze matrix eruit zou zien. Om er voor te zorgen dat er rekening gehouden wordt met de tijd is er in de matrix een kolom verwerkt waarin staat beschreven welke artikelen wanneer toegepast kunnen worden. Een kop koffie wordt naar alle waarschijnlijkheid van 7 uur tot ongeveer 2 uur gedronken en patat wordt ongeveer vanaf 12 uur tot 20 uur gegeten. Deze tijden zijn gebaseerd op de werktijden van de doelgroep zijnde de zakelijke reiziger. Het is aannemelijk dat een student die om 5 uur de trein naar huis neemt een zak patat wil in plaats van koffie maar de reisassistent is ontworpen voor de zakelijke reiziger waardoor een zak patat om 5 uur onwaarschijnlijker wordt.

Voorbeeld Matrix

Hobby's & voedingswaar → Artikelen ↓	Van/tot tijd	Motorrijden	Gamen	Espresso	Broodje gezond
Zuivel	7.00 – 14.00	0.966	1.084	1.054	1.132
Koffie	7.00 – 14.00	0.644	0.735	0.909	1.120
Motorblad	7.00 – 20.00	0.451	0.822	1.238	∞
Bioscoop film	14.00 – 20.00	0.974	0.887	1.442	∞
Haar knippen	14.00 – 18.00	0.936	1.119	1.413	0.769

NGD Waarden zijn fictief

Tabel 9 Matrix voorbeeld 2

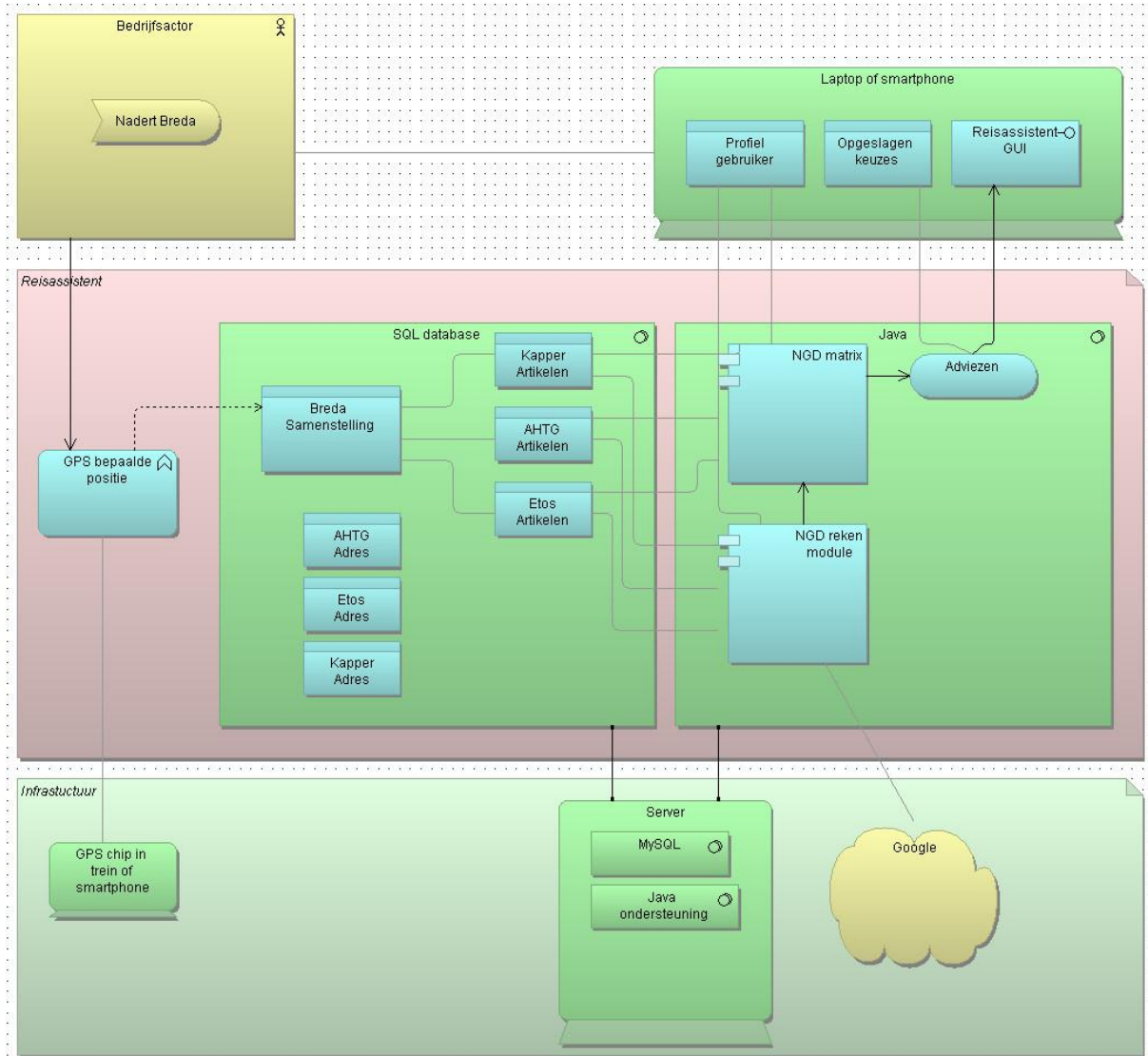
Wanneer de gebruiker dus op vijf minuten afstand van een station is gaat de reisassistent de adviezen voor het betreffende station uitrekenen. De reisassistent maakt gebruik van de matrix om dit proces te versnellen en de huidige tijd mee te kunnen nemen. De reisassistent vergelijkt eerste de matrix met de huidige tijd waardoor er een aantal mogelijke adviezen afvallen. Hierna worden de mogelijke artikelen gefilterd met de mogelijkheden op het betreffende station. Als laatste zal hij filteren op de hobby's en voedingswaren die de gebruiker heeft ingevoerd. Hierdoor blijven alleen de adviezen over die vallen binnen de betreffende tijds categorie, de mogelijkheden op het station en de voorkeuren van de gebruiker.

Van de overgebleven adviezen wordt er een selectie gemaakt van degene met een NGD waarde hoger dan 0.4. De gebruiker krijgt dan in het scherm van de reisassistent de top vijf adviezen te zien die uit te voeren zijn op het station. Tevens wordt er bij de adviezen rekening gehouden met eerder gekozen adviezen. Dit zal gebeuren door middel van een tabel waarin eerder gekozen adviezen staan opgeslagen. De top adviezen die uit de matrix komen worden vergeleken met adviezen die staan opgeslagen in de eerder gekozen adviezen. Indien er een advies tussenstaat die eerder gekozen is zal deze hoger komen staan dan andere adviezen.

### 6.2.2 Architectuur model

De architectuur van de reisassistent is gemaakt met behulp van de taal ArchiMate. Dit is een taal voor Enterprises architectuur en tevens geadopteerd door de The Open Group als een open standaard (39) (40). Als eerst zal er een tekstuele beschrijving van de architectuur worden gegeven en deze zal het gemaakte model in ArchiMate duidelijk maken.

Het onderstaande model geen schematisch weer hoe de beschrijving hierboven zal werken.



Afbeelding 13 Architectuur reisassistent

### 6.3 Database structuur

De informatie voor de reisassistent komt van verschillende bronnen die allen accurate informatie moeten verschaffen. In de onderstaande tekst worden deze bronnen beschreven en hoe zij de benodigde informatie gaan leveren.

Voor de reisassistent moet de volgende informatie beschikbaar zijn:

- **Profiel van gebruiker**  
Het profiel van de gebruiker heeft als bron de gebruiker zelf. De informatie die de gebruiker invult bij zijn profiel wordt opgeslagen op het apparaat waarop de reisassistent is geïnstalleerd en kan hierdoor altijd opgeroepen worden
- **Samenstelling van het station**  
De samenstelling van het station is beschikbaar op de website van de NS. Per station staat op ns.nl beschreven welke voorziening daar beschikbaar zijn. Dit geldt niet alleen voor de winkels waarover een betreffend station beschikt maar ook algemene voorzieningen zoals toilet of een NS hotspot. Deze informatie kan gebruikt worden om de samenstelling van een bepaald station te bepalen voor de reisassistent.
- **Artikelen van de winkels**  
De artikelen die de winkels op het station aanbieden zijn voor elke winkel verschillend. Elke winkel heeft wel een database waarin deze artikelen beschreven staan. Servex BV exploiteert diverse horeca- en retailformules op de stations van de NS. Dit doen ze onder merknamen als; Kiosk, AH to go, C'est du pain, De Restauratie, Smullers, New York Pizza, SWIRL'S en andere formules (41). Zij beschikken over de benodigde informatie over een merendeel van de stationwinkels. Resterende winkelketens die zijn gevestigd op het station zullen individueel benaderd moeten worden om achter de nodige artikel informatie te komen.
- **Station omliggende mogelijkheden**  
Omliggende mogelijkheden zullen eerst bepaald moeten worden aan de hand van afstand die een gebruiker kan lopen in 10 minuten. Aan de hand van deze informatie kan bepaald worden welke mogelijkheden interessant zijn voor een gebruiker en welke niet. De artikelen die de betreffende mogelijkheden aanbieden zullen in de database worden verwerkt om uiteindelijk te kunnen gebruiken in het advies.

De informatie die de verschillende bronnen aanbieden zullen opgeslagen moeten worden in een database die gebruikt kan worden door de reisassistent.

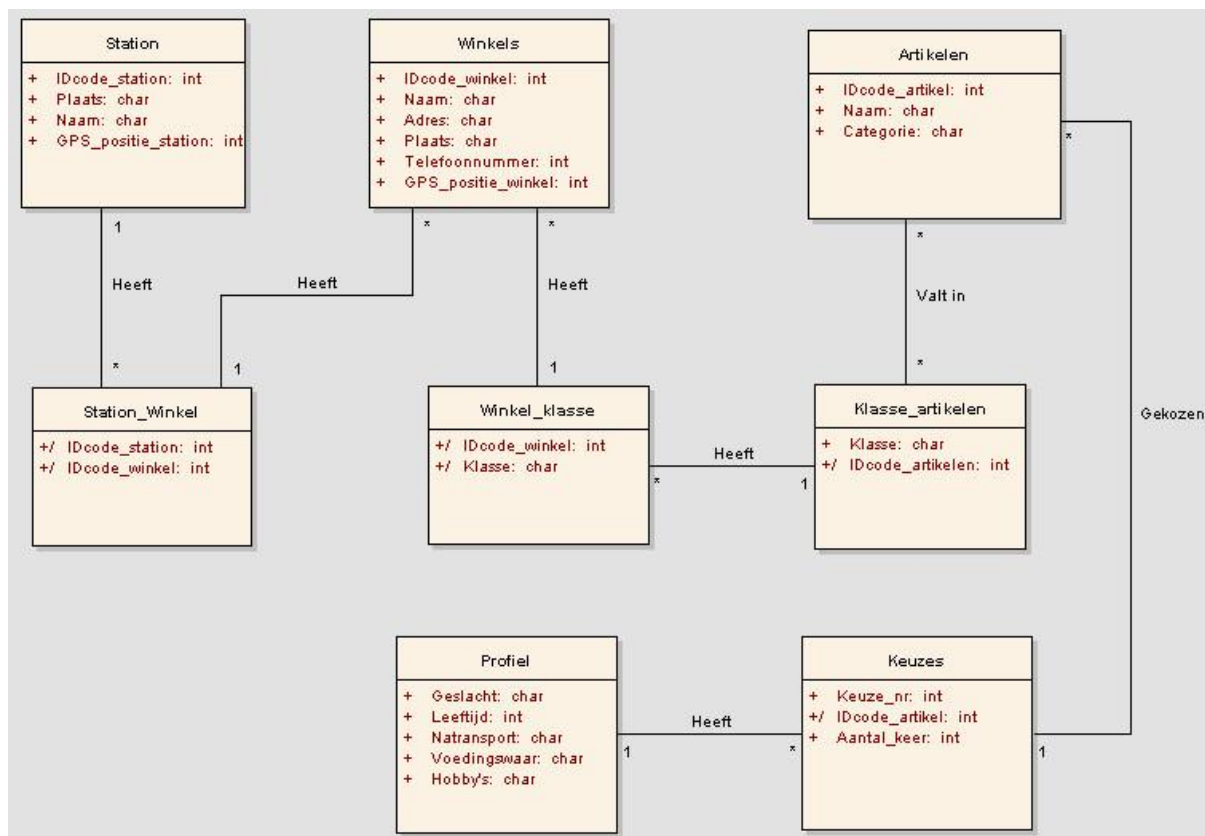
#### Soort database

Er zal voor de reisassistent gebruik gemaakt worden van een relationele database die benaderd kan worden met behulp van Structured Query Language(SQL). Deze vorm van databases heeft een aantal voordelen zoals een efficiënte dataopslag waardoor alles maar één keer opgeslagen hoeft te worden en Java heeft de mogelijkheid om met een relationele database een koppeling te maken.

#### Klassendiagram

Het klassendiagram beschrijft welke klasse, attributen en associaties er zijn voor de reisassistent. Op basis van onder andere dit model kan de reisassistent verwerkt worden in de database en Java code.

Voor de reisassistent zijn er twee verschillende plaatsen waarop informatie wordt opgeslagen. Het mobiele apparaat waarop de gebruiker werkt en de centrale server van de reisassistent kennen een database. Op het mobiele apparaat van de gebruiker wordt het profiel van de gebruiker opgeslagen. Op de centrale server staat de database waarin alle stations staan, de winkels van de stations en de artikelen die de winkels hebben. Op afbeelding 14 is te zien welke associatie de klassen hebben met elkaar.



Afbeelding 14 Klassendiagram

Op het mobiele apparaat van de gebruiker staan de klassen *Profiel* en *Keuzes*. Deze hebben de volgende functie:

- *Profiel*: In de klasse *profiel* staan alle gegevens van de gebruiker en deze kan hij wijzigen wanneer hij wil. Deze klasse wordt gebruikt om de NGD te bepalen samen met de klasse *Artikelen*.
- *Keuzes*: Deze klasse bevat alle gemaakte keuzes van de gebruiker. Deze klasse wordt gebruikt om eerder gemaakte keuzes terug te laten komen bij een volgend advies. Nadat het advies is samengesteld aan de hand van de NGD worden deze vergeleken met de al eerder gekozen adviezen. Indien adviezen overeenkomen met eerder gemaakte keuzen krijgen deze de voorkeur.

De centrale server staat de klassen *Station*, *Winkels*, *Artikelen*, *Station\_winkel*, *winkel\_klasse* en *Klasse\_artikelen*. Deze hebben de volgende functie:

- *Station*: In deze klasse staan alle station over het hele land met daarbij hun betreffende locatie. Tevens staat hier de GPS locatie van het station, deze wordt aangesproken door het GPS systeem om het juiste station gegevens te benaderen.
- *Winkels*: Deze klasse beschrijft de winkels die het betreffende station bezigt. Winkels zijn niet alleen de winkels op het station maar ook de winkels rond het station. Deze klasse wordt aangesproken wanneer het station is bepaald en kan gebruikt worden voor de looproute.
- *Artikelen*: De klasse *artikelen* bevat alle artikelen die beschikbaar zijn per winkel. Deze klasse is één van de onderdelen om de NGD te bepalen die in de matrix wordt verwerkt. De artikelen zijn benoemd op naam maar hebben ook een categorie die bij de NGD wordt gebruikt.
- *Station\_winkels*: Wanneer er bepaald is welk station de gebruiker nadert wordt er met deze klasse bepaald welke winkels het betreffende station bezit.
- *Winkels\_klasse*: Wanneer het station bepaald is wordt er met deze klasse bepaald in welke klasse de winkels vallen. Hierdoor kan bepaald worden welke artikelen de betreffende winkels bevatten.
- *Klasse\_artikelen*: In deze klasse staan de artikelen en in welke klasse deze vallen. Hiermee kan bepaald worden welke artikelen aan de gebruiker geadviseerd worden.

## 6.4 Uitgewerkte architectuur

In de volgende tekst en Unified Modeling Language(UML) modellen wordt de Use Case profiel invullen geheel uitgewerkt. Het gaat hier om de volgende modellen

- System Sequence Diagram(SSD)  
Dit model laat zien hoe de informatie uitwisselen is tussen de reisassistent en de gebruiker
- Functional User Interface(FUI)  
Dit model laat zien hoe de gebruiker taken kan uitvoeren met de reisassistent. Hierbij worden geen overbodige opmaak kenmerken gebruik.
- Extended System Sequence Diagram (ESSD)  
Dit model laat ook zien hoe de informatie uitwisseling is tussen gebruiker en de reisassistent. Tevens laat dit model zien hoe de informatie uitwisseling is tussen de reisassistent en de database.
- Extended Functional User Interface (EFUI)  
Dit model laat ook zien hoe de interface van de reisassistent eruit ziet maar deze laat ook zien hoe het dialoog is tussen gebruiker en reisassistent.

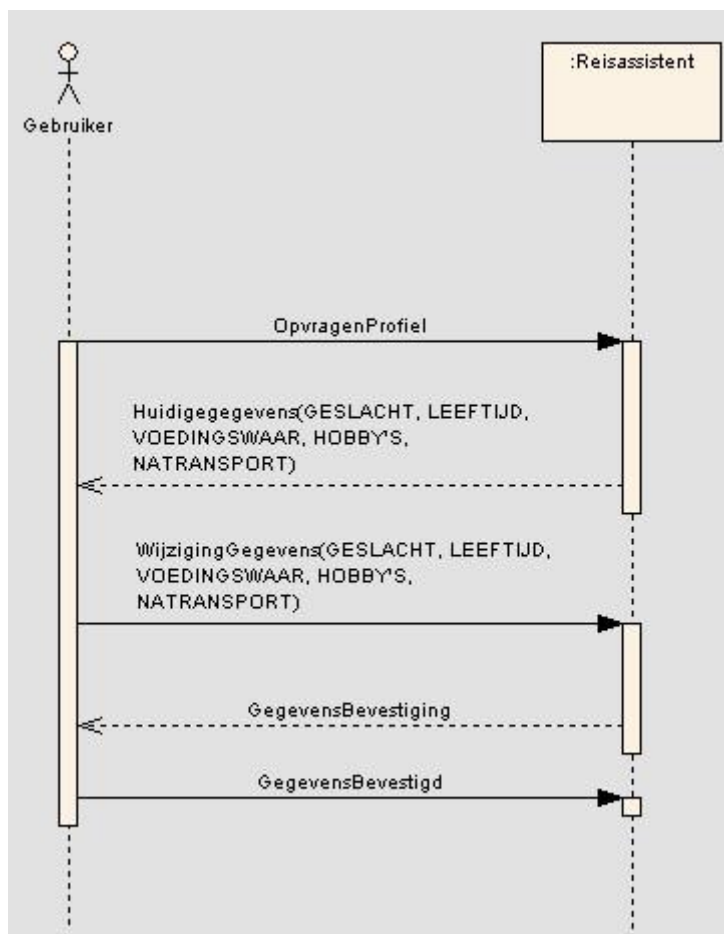
### 6.4.1 Use Case profiel invullen

Als eerste zal de Use Case Profiel invullen uitgewerkt worden. Met behulp van Unified Modeling Language(UML) modellen zal beschreven worden hoe de communicatie is tussen gebruiker, reisassistent en database. Tevens zal beschreven worden hoe de basis interface van de reisassistent eruit zal zien.

#### SSD

Wanneer de reisassistent voor het eerst opstart zal deze aan de gebruiker vragen om het profiel in te vullen. Indien de gebruiker zijn profiel wil wijzigen kan dit gedaan worden via het hoofdmenu. In dit geval is de gebruiker degene die de trigger is voor het proces. In afbeelding 15 is te zien hoe de informatie uitwisseling is.

De gebruiker vraagt het profiel op waarna de reisassistent de gevraagde gegevens toont. De gebruiker voert zijn wijzigingen in en klikt op de wijzigen knop. De reisassistent zal vragen of de gebruiker deze wijzigingen zeker weet en deze hierna opslaan.

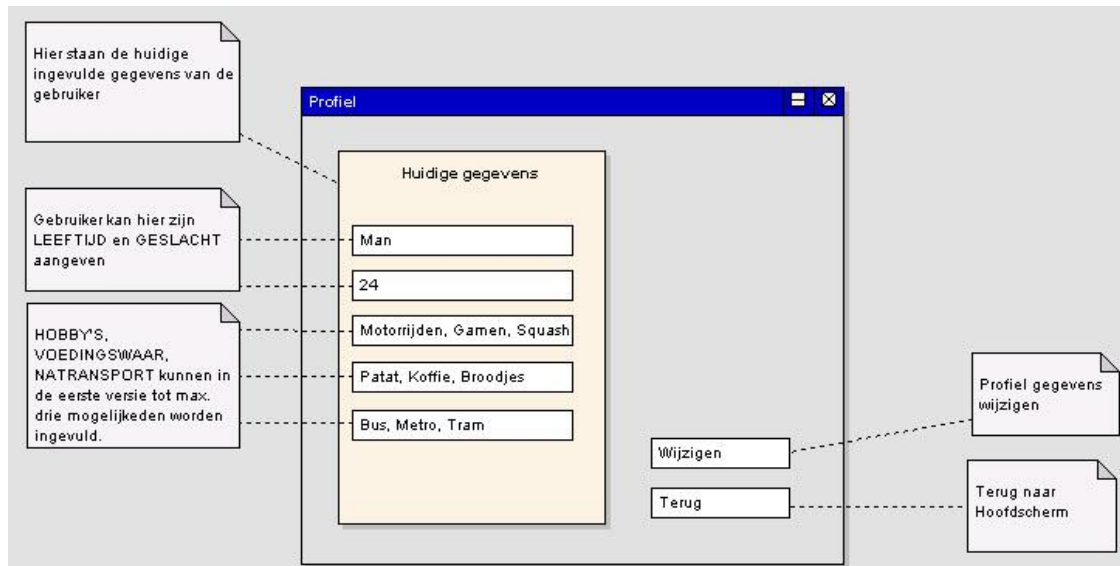


Afbeelding 15 SSD Invullen profiel



## FUI

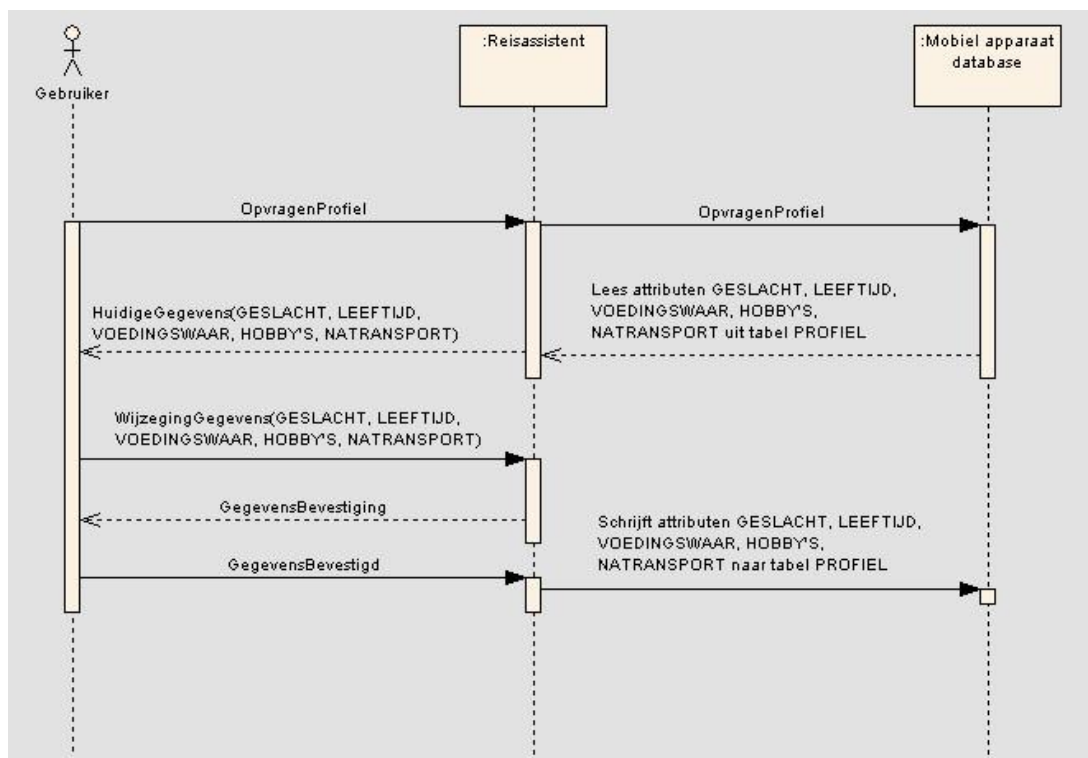
Het wijzigen van het profiel kan de gebruiker doen in het onderstaande scherm, afbeelding 16. Het laat alleen de informatie zien die de gebruiker wil weten en wijzigen. Hierdoor blijft het een rustig en overzichtelijk schema. In het scherm staat de initiële gegevens die de gebruiker kan wijzigen naar eigen inzicht. Na de wijzigingen kan de gebruiker op de knop wijzigen om de gegevens op te slaan.



Afbeelding 16 FUI Invullen profiel

## ESSD

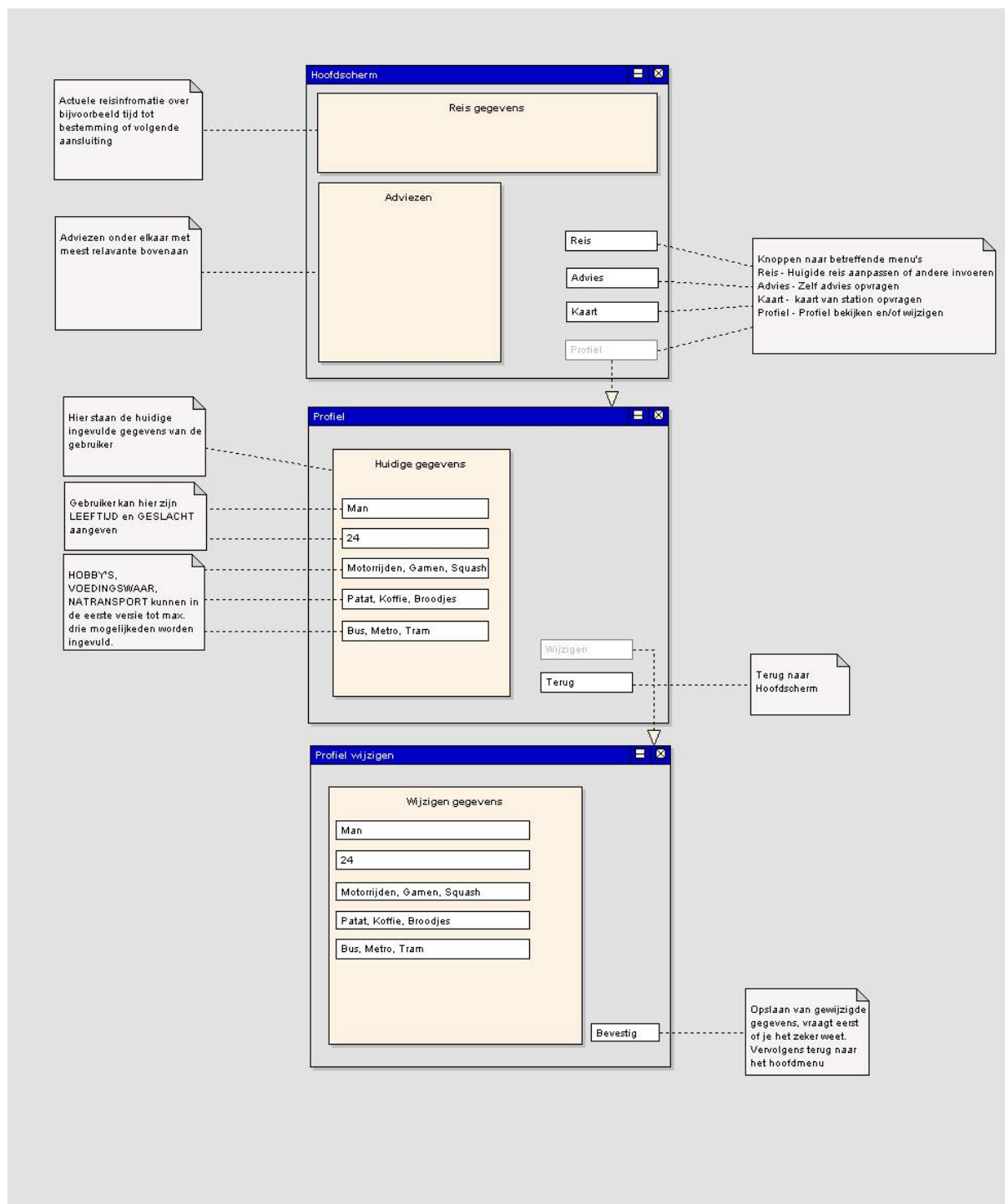
In afbeelding 17 is te zien hoe de reisassistent de gewijzigde data wegschrijft naar de database. Zoals vermeld in paragraaf 6.3 worden de gegevens opgeslagen op het mobiele apparaat van de gebruiker. De data wordt pas weggeschreven als de gebruiker de wijzigingen heeft bevestigd.



Afbeelding 17 ESSD Invullen profiel

## EFUI

Afbeelding 18 laat zien hoe het verloop is van de schermen door het proces van profiel wijzigen. De gebruiker begint in het hoofdscherm van de reisassistent. Hier kiest de gebruiker voor *Profiel* waar de informatie van zijn profiel staat. In deze pagina kan hij zijn wijzigingen aanbrengen en wanneer deze klaar zijn op *wijzigen* drukken. De reisassistent zal zoals beschreven bij de SSD eerst om een bevestiging vragen en indien de gebruiker dit goedkeurt de gewijzigde gegevens opslaan.



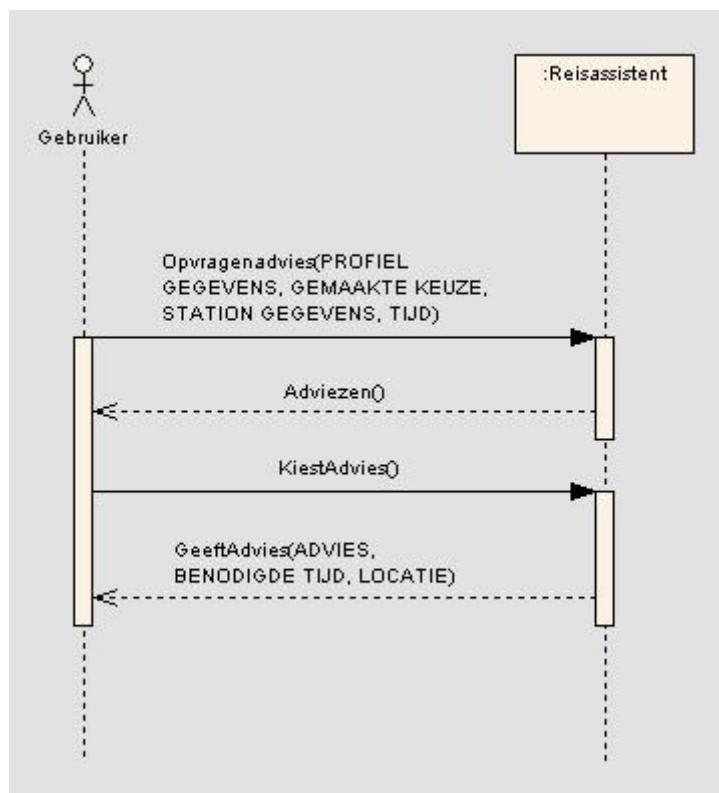
Afbeelding 18 EFUI Invullen profiel



#### 6.4.2 Use Case plannen wachttijd

##### SSD

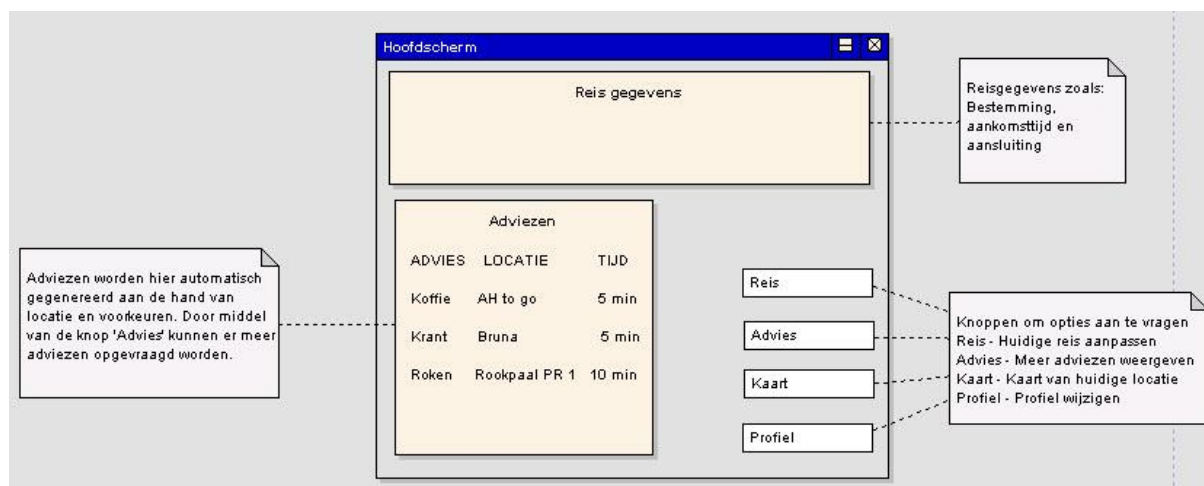
Het opvragen van advies gebeurt door de reisassistent automatisch indien de gebruiker in de buurt komt van een station. Indien de gebruiker de getoonde adviezen niet wil kan hij om meer adviezen vragen door op de knop advies te klikken (beschreven in afbeelding 19 de FUI). Dit scenario wordt in de onderstaande SSD beschreven. Wanneer de gebruiker het advies opvraagt gaat er informatie naar de reisassistent om deze uit te rekenen. De reisassistent zal dan adviezen teruggeven waaruit de gebruiker kan kiezen. Wanneer de gebruiker een advies heeft gekozen komt deze centraal in beeld en zal er specifieke informatie worden getoond.



Afbeelding 19 SSD Plannen wachttijd

##### FUI

De onderstaande afbeelding, afbeelding 20, is het hoofdscherm van de reisassistent waarin de reisgegevens en de adviezen getoond worden. De reisgegevens staan bovenin maar worden in dit verslag niet verder behandeld. De adviezen staan links en tonen de adviezen die gegenereerd worden aan de hand van de NGD.

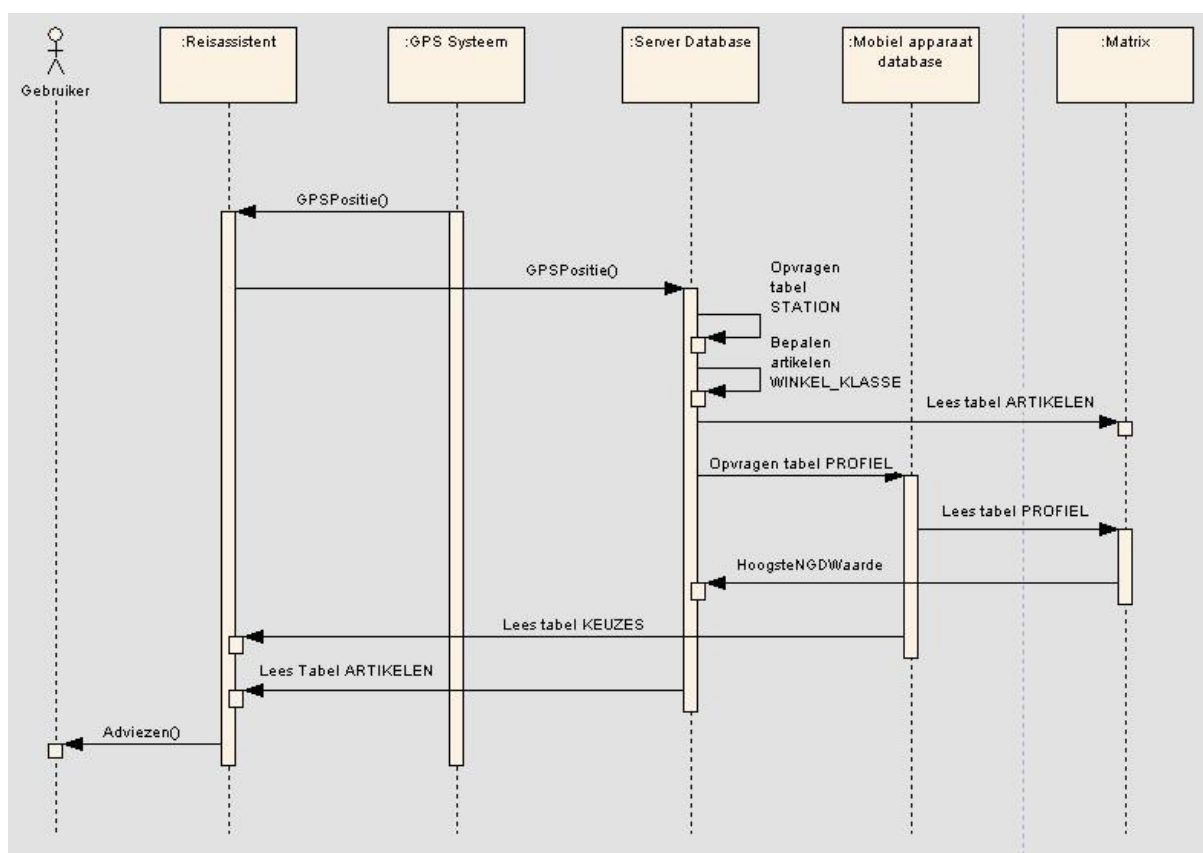


Afbeelding 20 FUI Plannen wachttijd

## ESSD

In de onderstaande afbeelding, afbeelding 21, is het scenario uitgewerkt waarin de reisassistent automatisch adviezen genereert wanneer er een station genaderd wordt. Via GPS wordt bepaald waar de gebruiker is en ook hoe ver hij van het station vandaan is. Deze positie gaat naar de server database waar de betreffende *Station* tabel wordt aangesproken. In dit scenario wordt wel gebruik gemaakt van de bekende reisinformatie zoals een reis van Rotterdam naar Gouda. Als de gebruiker in het voorbeeld van Rotterdam naar Gouda reist met een intercity komt hij lang stations waar hij geen informatie over nodig heeft. De trein stopt niet op Rotterdam Noord dus daar hoeft de gebruiker geen adviezen voor. Om er dus voor te zorgen dat de gebruiker pas advies krijgt wanneer hij dit nodig heeft komt er informatie uit de geplande bestemmingen van de gebruiker.

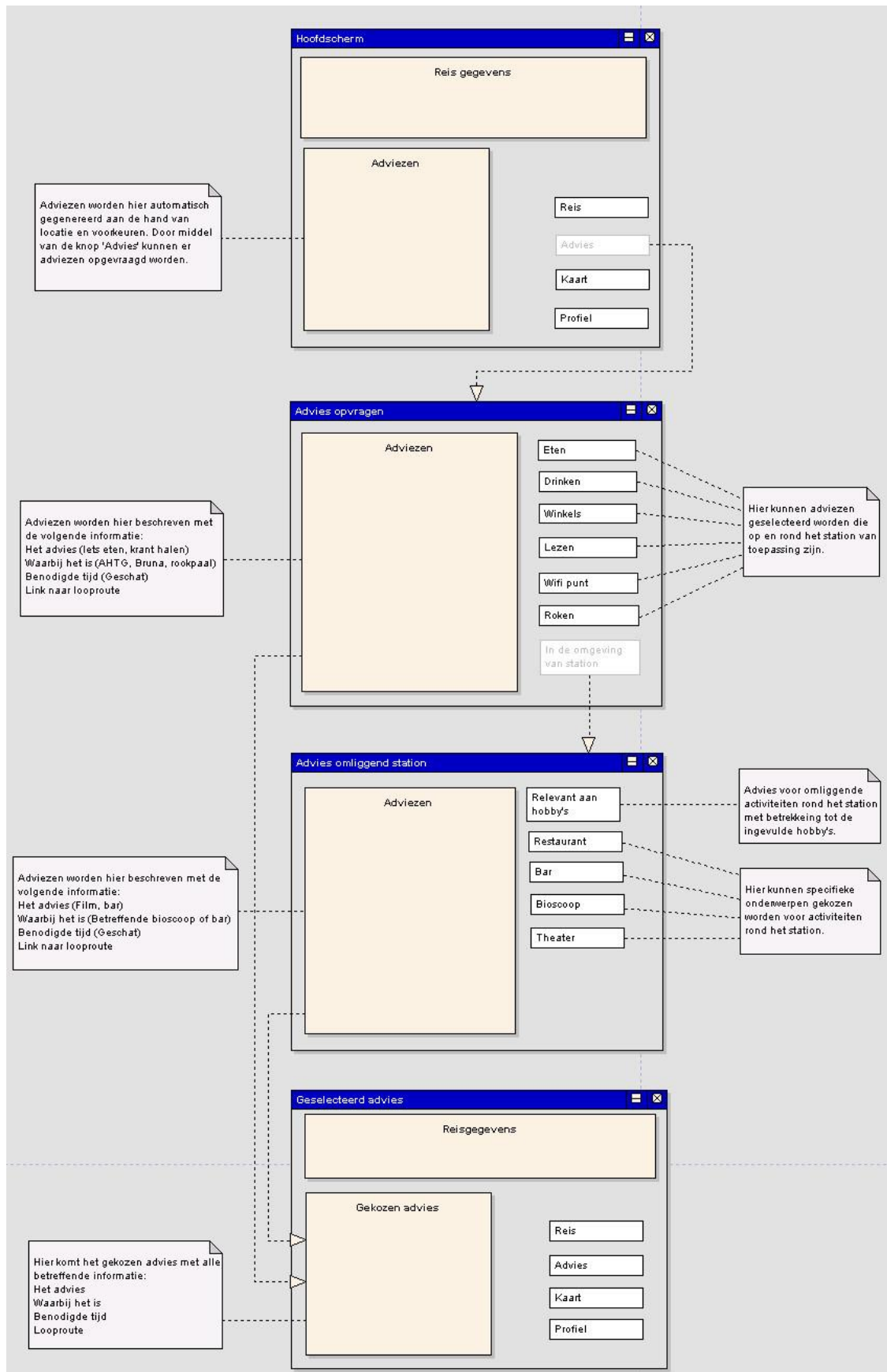
Gekoppeld aan de tabel *Station* zitten de tabellen *Winkels* en *Artikelen* die worden aangesproken om het advies uit te werken. Tevens wordt er een aanspraak gedaan op de tabel *Profiel* die op het mobiele apparaat staat. Hier wordt ook bepaald welke artikelen voor het station gelden door middel van de Klasse. Als eenmaal de betreffende artikelen zijn bepaald worden de NGD waarden bepaald aan de hand van de matrix. Deze waarde gaan naar de server database waar de betreffende artikelen en winkels worden geselecteerd. Als laatste worden de samengestelde adviezen vergeleken met de al eerder gekozen adviezen waaruit de uiteindelijke adviezen komen die de gebruiker te zien krijgt.



Afbeelding 21 ESSD Plannen wachttijd

## EFUI

In afbeelding 22 is te zien hoe het verloop van het proces loopt als de gebruiker meteen een advies kiest, meer advies wil of kiest op een categorie. De gebruiker krijgt linksonder te zien welke adviezen er zijn en indien hij er één goed vindt kan hij deze kiezen. Dit advies zal dan alleen komen te staan samen met de looproute. Indien de gebruiker nog meer advies wil kiest hij voor *advies*. Hierdoor komt hij in het scherm waarin meer adviezen staan en hij kan kiezen om adviezen per categorie te sorteren. Kiest hij hier een advies dan gaat hij weer terug naar het hoofdmenu. Kiest hij voor een categorie dan komen er adviezen die alleen in die categorie vallen. Voor *Omliggend station* zijn er nog een aantal categorieën beschikbaar. Uiteindelijk kiest de gebruiker hier een advies en komt hij weer in het hoofdmenu terecht.



Afbeelding 22 EFUI Plannen wachttijd

## 6.5 Conclusie

In dit hoofdstuk is behandeld hoe de functionele eisen van de reisassistent vertalen naar een technisch ontwerp. Hiermee wordt ook antwoord gegeven op de vierde hoofdvraag. Hoe moeten de functionele eisen op semantische wijzen gaan samenwerken? De Use Cases die zijn opgesteld in het functioneel ontwerp worden verder uitgewerkt aan met behulp van een aantal modellen. Tevens word de systeem architectuur uitgewerkt en hoe de achterliggende database gebruikt wordt.

De deelvraag wordt beantwoord aan de hand van de NGD uitwerking in de reisassistent. De functionele eisen zijn vastgesteld in het functioneel ontwerp en door middel van de NGD verwerking wordt er semantiek toegevoegd aan de reisassistent. Om deze semantiek te bereiken maakt de NGD formule gebruik van de voorkeuren van de gebruiken die in een profiel staan ingevuld en de mogelijkheden op het station. Door de twee woorden uit de beide bronnen in Google in te voeren komen er getallen waarmee de NGD waarde brekend kan worden en bepaald kan worden of deze twee woorden een semantisch verband hebben. Aan de hand van deze waarde wordt bepaald of het betreffende artikel wel of niet als advies aan de gebruiker moet worden getoond.



## 7 Het Proof of Concept

Dit hoofdstuk zal het Proof of Concept beschreven worden en hoe deze tot stand is gekomen. Het POC is een bewijs dat de NGD formule kan worden toegepast in een applicatie zodat er een waarde terug gegeven kan worden. Tevens wordt met het POC de hoofdvraag van dit project beantwoord.

Hoe moet een reisbewaker functioneren om op semantische wijze rekening te kunnen houden met de verschillende aspecten van een treinreis.

In dit hoofdstuk komt eerst aan bod welke functies het proof of concept zal hebben en hierna hoe dit is uitgewerkt.

### 7.1 Onderdelen Proof of Concept

Zoals beschreven in de BCL zal het proof of concept de volgende functie kunnen vervullen.

BCL Proof of Concept	
Basis	<ul style="list-style-type: none"> <li>• Database met 100 artikelen</li> <li>• Tekstveld waar gebruiker een hobby of voedingswaar kan invullen</li> <li>• Tekstveld kunnen uitlezen</li> <li>• Uitgelezen tekst en database kunnen verwerken met NGD</li> <li>• Een NGD waarde kunnen teruggeven</li> </ul>
Comfort	<ul style="list-style-type: none"> <li>• Met behulp van de NGD waarde een advies kunnen geven</li> </ul>
Luxe	<ul style="list-style-type: none"> <li>• Een interface in de huisstijl van de NS</li> </ul>

Tabel 10 BCL POC 2

Dit zijn de basis eigenschappen die zijn uitgewerkt en bewijzen dat de NGD een waarde kan teruggeven die kan aangeven of twee woorden een semantisch verband met elkaar hebben.

### 7.2 Google vs Yahoo

Voorafgaande aan dit hoofdstuk moet gemeld worden dat voor het POC gebruik is gemaakt van de hoeveelheid zoekresultaten van Yahoo. Dit komt omdat Google alle mogelijkheden blokkeert om met een externe applicatie connectie te maken met Google. Het is alleen mogelijk om met Google te werken als er gebruik gemaakt wordt van Google Application Programming Interface (API). Vroeger was deze connectie wel mogelijk maar met een beperking van 100 zoekquery's per 24 uur.

Tijdens dit project is er geen tijd geweest om deze aan te vragen dus is er gekozen voor Yahoo. Hiervoor is ook de aantal geïndexeerde pagina's aangepast naar die van Yahoo (12). Hierdoor kunnen de uitkomsten van de POC anders uitvallen maar dat de techniek werkt is alsnog bewezen.

### 7.3 Uitwerking POC

Om de rest van dit hoofdstuk wordt beschreven hoe de POC is uitgewerkt met behulp van de NGD. Tevens wordt beschreven hoe de database is gemaakt en hoe de onderdelen van de POC zijn opgesteld. Als laatste wordt beschreven hoe de POC uiteindelijk werkt.

### 7.3.1 Database

Voor het testen van de NGD formule is er gebruikt gemaakt van een test tabel met daarin 100 artikelen. De artikelen zijn willekeurig opgesteld in de vorm van voedselcategorieën en mogelijke hobby's. De tabel is gemaakt met behulp van MySQL om dezelfde relationele database te gebruiken die de echte reisassistent ook zal gebruiken. Op afbeelding 23 is een voorbeeld te zien van de tabel artikelen.

The screenshot shows the MySQL Query Browser interface. The SQL Query Area contains the query: `SELECT * FROM artikelen a;`. The Results window displays a table with 100 rows. The first column is 'ID' and the second column is 'Artikelen'. The data includes various items like 'Motomijden', 'Kitesurfen', 'Mode', 'Gamen', 'Broodjes', 'Squash', 'Koffie', 'Auto', 'Zuivel', 'Salade', 'Sportdrink', 'Snoep', 'Krant', 'Chocolade', 'Onbait', 'Tennis', 'TV', 'Fitness', 'Kickboxen', 'Boeken', 'Pasta', 'Frikandel', 'Kroket', 'Hamburger', 'Italiaans', 'Bar', 'Bloemen', 'Sigaretten', 'Sigaren', 'Aansteker', 'Shag', 'Computer', 'Internet', 'Mac Donalds', 'Burger King', 'Warm broodje', and 'Voetbal'.

Afbeelding 23 POC database

### 7.3.2 NGD formule naar code

Hieronder is te zien hoe de NGD formule vertaald naar Java code:

$$NGD(x, y) = \frac{\max\{\log f(x), \log f(y)\} - \log f(x, y)}{\log M - \min\{\log f(x), \log f(y)\}}$$

Vertaald naar code

```
NGD = (Math.max(Math.log(x), Math.log(y)) - Math.log(xy))
      / (Math.log(M) - Math.min(Math.log(x), Math.log(y)));
```

Om deze functie te vertalen naar Java code is er gebruik gemaakt van de Math functie die alle wiskundige functies bevat die nodig zijn om de NGD uit te rekenen. De functies *max* en *min* zijn vanzelfsprekend maar waar op gelet moet worden is de *log* functie. Voor deze functie moet gebruik worden gemaakt van het natuurlijk logaritme en het Briggse logaritme. Het Briggse logaritme wordt in Java aangeduid met de functie *log10* en het natuurlijk logaritme met de functie *log*.

### 7.3.3 Database connectie

De connectie met de database is ook gemaakt met behulp van Java. Hieronder is te zien hoe de connectie tot stand is gekomen. Er is gebruik gemaakt van een *Vector* om de artikelen één voor één te kunnen lezen. Door de artikelen één voor één uit de database te halen kunnen ze in Yahoo geplaatst worden om zoekresultaten te genereren.

```
public class dbconnectie {

    public Vector<String> getDatabaseInvoer() throws Exception
    {
        Vector<String> returnWaarde = new Vector<String>();

        Class.forName("com.mysql.jdbc.Driver");
        Connection con = DriverManager.getConnection(
            "jdbc:mysql://localhost:3306/artikelen", "root", "root");

        Statement stmt = con.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,
            ResultSet.CONCUR_READ_ONLY);

        ResultSet result = stmt.executeQuery("SELECT ID, artikelen FROM artikelen");

        while (result.next()) {
            returnWaarde.add(result.getString("artikelen"));
        }

        return returnWaarde;
    }
}
```

### 7.3.4 Tekstveld input gebruiker

In de onderstaande twee tekstvelden is de Java code te zien waarin het tekstveld wordt gegenereerd waarin de gebruiker zijn zoekwoord kan invullen. Als de applicatie wordt gestart verschijnt er een tekstveld waarin de gebruiker een zoekwoord kan invullen. Indien de gebruiker geen zoekwoord invoert komt er een melding waarin staat dat er geen zoekwoord is ingevoerd. Tevens staat in dit stuk code dat het aantal woorden uit de database die worden vergeleken wordt gelimiteerd met 20. Later in het hoofdstuk meer over dit onderwerp.

```
import javax.swing.*;

import dbconnectie.dbconnectie;

import java.awt.event.*;
import java.io.IOException;
import java.util.Vector;

public class Namer extends JFrame {
    public static void main(String[] args) {
        new Namer();
    }
    public dbconnectie db = new dbconnectie();
    public NGD ngd = new NGD();
    private JButton buttonOK;
    private JTextField textInvoer;

    public Namer() {

        this.setSize(325, 100);
        this.setTitle("Invoer zoekwoord");
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        ButtonListener bl = new ButtonListener();
        JPanel panell = new JPanel();
        panell.add(new JLabel("Je hobby of voedingswaarde: "));
        textInvoer = new JTextField(15);
        panell.add(textInvoer);
        buttonOK = new JButton("OK");
        buttonOK.addActionListener(bl);
        panell.add(buttonOK);
        this.add(panell);
        this.setVisible(true);
    }
}
```



```

public class ButtonListener implements ActionListener {
    public void actionPerformed(ActionEvent e) {
        double ngdWaarde = 0;
        double grootste = Double.MAX_VALUE;
        String dbwoord = "";
        String invoer = "";
        Vector<String> dbWard = new Vector<String>();

        if (e.getSource() == buttonOK) {
            invoer = textInvoer.getText();
            if (invoer.length() == 0) {
                JOptionPane.showMessageDialog(Namer.this,
                    "Je hebt niks ingevuld", "Geen zoekwoord",
                    JOptionPane.INFORMATION_MESSAGE);
            } else {
                try {
                    dbWard = db.getDatabaseInvoer();
                } catch (Exception e2) {
                    e2.printStackTrace();
                }
                for(int i = 0 ; i < 20 ; i++){
                    try {
                        ngdWaarde = ngd.getNGD(invoer,
                            dbWard.elementAt(i));
                        System.out.println(dbWard.elementAt(i) + " met
" + invoer + " geeft NGD " + ngdWaarde);

                        if(ngdWaarde < grootste) {
                            grootste = ngdWaarde;
                            dbwoord = dbWard.elementAt(i);
                        }
                    } catch (IOException e1) {
                    }
                }

                JOptionPane.showMessageDialog(Namer.this, "Het semantisch
verband tussen: " +
                    invoer + " en " + dbwoord + " is het grootst
met een NGD waarde van " + grootste, "Uitkomst",
                    JOptionPane.INFORMATION_MESSAGE);
                textInvoer.requestFocus();
            }
        }
    }
}

```

### 7.3.5 Hoeveelheid zoekresultaten extraheren

Door middel van de woorden van de database en de invoer van de gebruiker wordt er een functie(*getZoekResultaten*) opgevuld. Deze functie plaatst de zoekwoorden in Yahoo en voert een zoekquery uit met de betreffende woorden. Tevens extraheert deze functie het aantal zoekresultaten en maakt ze bruikbaar voor de NGD formule.

```
public static Double getZoekResultaten(String query) throws IOException
{
    double Resultaten = 0;

    URL yahoo = new URL("http", "proxy.logica.com", -1, "http://uk.search.yahoo.com/search?p=" +
        query);
    URLConnection yc = yahoo.openConnection();
    BufferedReader in = new BufferedReader(
        new InputStreamReader(
            yc.getInputStream()));

    String inputLine;

    while ((inputLine = in.readLine()) != null)
    {
        if (inputLine.contains("</strong> results for")) {

            inputLine.indexOf("</strong> results for");
            String result = inputLine.substring(inputLine.indexOf("resultCount") + 13, 1
            inputLine.indexOf("</strong> results for"));
            String result2 = result.replace(",", ""); 2
            Resultaten = Double.parseDouble(result2); 3

        }
    }
    in.close();
    return Resultaten;
}
```

<sup>1</sup> Hier worden de hoeveelheid zoekresultaten van de broncode geextraheerd.

<sup>2</sup> De zoekresultaten zijn nog in String vorm en bevatten komma's. Deze kunnen niet verwerkt worden en worden hier verwijderd.

<sup>3</sup> De resultaten worden hier verwerkt naar een Double zodat er mee gerekend kan worden in de NGD.

Aan het begin van het hoofdstuk staat vermeld dat Yahoo wordt gebruikt in plaats van Google omdat deze wel toegang biedt tot de broncode. Tevens wordt er vermeld dat Google een limiet plaatste op het aantal zoekquery. Tijdens het maken van het POC is er ondervonden dat Yahoo een zelfde restrictie op de zoekmachine heeft. Yahoo heeft een limiet van 150 zoekqueries per 24 uur van één IP adres. Omdat dit nog maar het POC is, is er voor besloten om de foutmelding maar uit te zingen en 24 uur te wachten voor de volgende test. Hieronder is de error melding te zien na te veel zoekquery bij Yahoo.

Server returned HTTP response code: 999

Ook is er een error tijdens het maken van de POC geconstateerd die onregelmatig optreedt. Uit debuggen is gebleken dat de broncode soms te lang duurt om te laden waardoor de Java code al begint met het lezen van de pagina voordat deze helemaal geladen is. Dit is tegen gegaan door in de code te verwerken dat de broncode eerst helemaal geladen moet zijn voordat de Java code begint.

### 7.3.6 NGD waarde krijgen

In de onderstaande code is te zien hoe de resultaten uit de *getZoekResultaten* functie worden verwerkt in de NGD formule. De *M* waarde is de hoeveelheid geïndexeerde pagina's van Yahoo en is volgende de website 'size of the World wide web' (12) opgesteld. Na berekend te hebben wat de NGD waarde is wordt de terug gegeven om te gebruiken in het tekst veld. Na het berekenen van de waarde wordt aan de gebruiker getoond welke woord combinatie van het door hem ingevoerde woord en het woord uit de database de kleinste NGD waarde retourneert. Dit representeert het grootste semantisch verband van de 20 woorden.

```
public double getNGD(String x, String y) throws IOException
{
    double x = getZoekResultaten(x);
    double y = getZoekResultaten(y);
    double xy = getZoekResultaten(x + "+" + y);
    double M = 52E9; // Math.pow(10, 10); // aantal geïndexeerde pagina's
    double NGD;

    NGD = (Math.max(Math.log(x), Math.log(y)) - Math.log(xy))
        / (Math.log(M) - Math.min(Math.log(x), Math.log(y)));

    return NGD;
}
```

### 7.3.7 Uiteindelijke werking POC

Hierboven is aangegeven welke code het mogelijk maakt om met een woord van de gebruiker en de woorden uit de database een NGD waarde te formuleren. Dit resulteert in een aantal stappen die zichtbaar zijn voor de gebruiker. In de onderstaande beschrijving is te zien wat de gebruiker ervaart met het POC:

De gebruiker krijgt eerste een scherm waarin hij een hobby of voedingswaar kan invullen te zien in afbeelding 24.



Afbeelding 24 Invoer POC

Indien de gebruiker geen woord intikt krijgt hij de melding dat er geen invulling is, te zien op afbeelding 25.



Afbeelding 25 Geen invoer POC

Wanneer de gebruiker wel iets invult en in dit geval 'Melk', gaat het POC melk vergelijken met de artikelen uit de database. Zoals vermeld, wordt het aantal combinatie beperkt tot 20. Niet alleen om de looptijd van het POC te verkorten maar ook om de POC meerdere keren te kunnen draaien op een dag.

De onderstaande resultaten worden in de console van Java gegenereerd. Er is te zien welke woordcombinatie welke NGD waarde genereert.

```
Motorrijden met Melk geeft NGD 0.7636787695395807
Kytesurfen met Melk geeft NGD 0.831100484386559
Mode met Melk geeft NGD 0.9803179152874113
Gamen met Melk geeft NGD 0.7027554060009447
Broodjes met Melk geeft NGD 0.39950173338579453
Squash met Melk geeft NGD 0.8993457293450694
Koffie met Melk geeft NGD 0.3161009032381753
Auto met Melk geeft NGD 0.8996115549387035
Zuivel met Melk geeft NGD 0.34743313371036816
Salade met Melk geeft NGD 0.46451802022520866
Sportdrank met Melk geeft NGD 0.5679270147271499
Snoep met Melk geeft NGD 0.4085683109708891
Krant met Melk geeft NGD 0.5446305134624885
Chocolade met Melk geeft NGD 0.33711170816820685
Ontbijt met Melk geeft NGD 0.3501135658912598
Tennis met Melk geeft NGD 0.976467888147703
TV met Melk geeft NGD 0.9755236396823658
Fitness met Melk geeft NGD 0.9572519841668451
Kickboxen met Melk geeft NGD 0.9548612910234436
Boeken met Melk geeft NGD 0.5465668059813893
```

Wanneer de 20 combinaties zijn doorgelopen wordt er een veld gegenereerd waarin staat welke combinatie de kleinste NGD waarde heeft tussen opgeleverd twee woorden die dus het grootste semantische verband vormen volgens de NGD. Van de 20 woord combinaties die hierboven staan aangegeven was dit de uitkomst: afbeelding 26.



Afbeelding 26 Uitvoer POC

## 7.4 Bevindingen tijdens uitwerking

### 7.4.1 Opvallende uitkomsten

Opvallend is uit deze test is dat de combinatie zuivel en melk een grotere NGD waarde oplevert dan koffie en melk. Hoewel het verschil niet groot is komt er toch een ander resultaat dan je als mens zou verwachten.

```
Koffie met Melk geeft NGD 0.3161009032381753
Chocolade met Melk geeft NGD 0.33711170816820685
Zuivel met Melk geeft NGD 0.34743313371036816
Broodjes met Melk geeft NGD 0.39950173338579453
```

Hieruit kan geconcludeerd worden dat in de geïndexeerde pagina's van Yahoo de combinatie koffie en melk vaker voorkomt dan de combinatie van zuivel en melk. Google beschikt over het algemeen over een betere zoekmachine dan Yahoo waardoor de resultaten wellicht anders kunnen uitpakken. Helaas is dit nog niet getest en wordt overgelaten aan de volgende uitvoerder.

#### 7.4.2 Verwachte uitkomsten

Ook mag gemeld worden dat de meest onwaarschijnlijke combinaties ook duidelijk worden aangegeven. Hieronder is te doen dat er duidelijk geen semantisch verband is tussen Mode en melk of Tennis en Melk.

Mode met Melk geeft NGD 0.9803179152874113 Squash met Melk geeft NGD 0.8993457293450694 Tennis met Melk geeft NGD 0.976467888147703
---

Tevens is het in de test voorgekomen dat bepaalde combinaties het resultaat infinity terug gaven waarvoor duidelijk geconcludeerd kan worden dat de betreffende woorden echt niks met elkaar te maken hebben.

Kytesurfen met Poep geeft NGD Infinity
--

#### 7.4.3 Spelling woorden

Tijdens het ontwikkelen van het POC is ondervonden dat de uitkomsten van de NGD formule kunnen variëren aan de hand van spelling. Zoals hieronder valt te zien geeft de combinatie kytesurfen,kitesurfen een andere NGD waarde dan de combinatie kytesurfen,kytesurfen.

Kytesurfen met kitesurfen geeft NGD 0.6026617009946523
--

Kytesurfen met kytesurfen geeft NGD 0.01439924605329423
---

Indien de reisassistent gebruik maakt van een tekstveld waarin de gebruiker zelf zijn hobby's en favoriete voedingswaar kan invullen is het aannemelijk om te concluderen dat spellingfouten zullen voorkomen. Voor de uitwerking van de reisassistent moet hier dus rekening mee gehouden worden.

#### 7.4.4 Wanneer semantisch verband

In het POC worden automatisch de NGD waarden uitgerekend. Maar wanneer geeft de NGD waarde aan dat er een semantisch verband is tussen twee woorden. Om dit te bepalen zullen er tests gedaan moeten worden op grote schaal om te bepalen wat de gemiddelde waarden zijn en wat uitzonderlijk kleine waarden zijn. Ook kan er gekozen worden om de drie kleinste waarde te selecteren en deze als advies te gebruiken.

#### 7.4.5 Afronden getallen

De getallen die door de POC gegenereerd worden geven gemiddeld 16 getallen achter de komma terug. In een verder uitwerking van de POC zal dit ingekort moeten worden tot 3. Een groot aantal getallen achter de komma geeft een valse indruk dat de berekening precies berekend wat het semantische verband is.

#### 7.4.6 Combinatie chocolade melk

Uit het POC is gekomen dat bijvoorbeeld Melk en chocolade een kleinere waarden terug geeft dan Zuivel en Melk. Dit komt waarschijnlijk doordat chocolade en melk al in twee vormen terug komen, zoals melkchocolade en chocolademelk, een snoepgoed en een drankje. Aan de hand van dit voorbeeld kan er gezegd worden dat de NGD techniek niet alle aspecten van semantiek kan indekken. Indien er een RDF en OWL database opgezet zou worden kan hierin wel onderscheid gemaakt worden.

#### 7.4.7 Cash resultaten

In een verder uitwerking van de POC kunnen de resultaten gecached worden. Hiermee worden de uitgerekende NGD waarden opgeslagen zodat alleen nieuwe waarden worden uitgerekend. Hiermee kan de matrix gesimuleerd worden.

## 8 Conclusie onderzoeksrapport

In het rapport worden in verschillende hoofdstukken de deelvragen beantwoord en in het hoofdstuk POC wordt de hoofdvraag beantwoord maar nog niet beschreven. Als eerste komen de deelvragen aan bod Hierna de hoofdvraag. Als laatste wordt beschreven hoe er een complete reisassistent gemaakt kan worden aan de hand van verschillende projecten binnen WT.

### 8.1 Antwoord deelvragen

#### 1. Welke aspecten van de reis wil de gebruiker terug zien in de reisassistent?

Om deze vraag te beantwoorden is er een gebruikers onderzoek gedaan binnen de doelgroep. Dit is gedaan met behulp van een enquête en interviews. Dit onderzoek was om een indruk te krijgen van hun wensen betreffende de reisaspecten die zij terug willen zien in de reisassistent.

De uitkomsten hiervan zijn dat de doelgroep als eerste graag terug wil zien dat zij te weten krijgen wanneer er een vertraging is en welke alternatieve reisopties er dan is. Als tweede wil de doelgroep weten wat zij kunnen doen op het station in de vorm van een advies. Hierna komt het aangeven van een looproute door het station met als laatste het bijhouden van hun reis. Aan de hand van deze uitkomsten zijn er functionaliteiten opgesteld die in de Case terug komen.

#### 2. Wat is de huidige status van semantische technologie?

Om deze vraag te beantwoorden is er een deskresearch gedaan en een literatuurstudie. Hierdoor is er gezocht op het internet met onder andere Google Scholar. Tevens zijn er een aantal boeken gevonden die de huidige semantische technologieën in detail beschrijven.

Uit dit stuk onderzoek is gebleken dat de beste manier voor de reisassistent om semantiek te bereiken is via de NGD formule. Deze formule brekend op basis van twee woorden die een aantal zoekresultaten geven in Google of er een semantisch verband is. De voordelen van deze techniek is dat alle woorden altijd bekend zijn en er geen uitgebreide database gemaakt hoeft te worden waarin alle woorden staan beschreven met hun semantiek. Tevens is voor de reisassistent het alleen maar nodig om te weten of twee woorden en semantisch verband hebben en niet wat dit verband is.

#### 3. Hoe vertalen de reisaspecten naar functionaliteiten in de reisassistent?

Om deze vraag te beantwoorden is er een Case opgesteld waarin wordt geschetst hoe een persona gebruik zou maken van de reisassistent en welke praktische onderdelen komen kijken bij de reisassistent. Tevens is er een functioneel ontwerp gemaakt waarin de functionaliteiten worden vertaald naar Use Cases om te gebruiken voor het technisch ontwerp. In de Case zijn de volgende functionaliteiten gedefinieerd aan de hand van het onderzoek:

1. Profiel in kunnen vullen
2. Je reis plannen
3. Het melden van vertragingen en een alternatieve route aangeboden krijgen
4. Adviezen krijgen om je wachttijd te vullen
5. Looproute kunnen ontvangen naar activiteit of aansluiting

In het functioneel ontwerp zijn deze functionaliteiten uitgewerkt naar Use Cases zijnde:

#### Use Cases

1. Invullen profiel
2. Opvragen reisinformatie
3. Opvragen informatie vertraging
4. Plannen wachttijd
5. Opvragen looproute

Het profiel geeft één van de twee woorden die nodig zijn voor de NGD formule. Het plannen van de wachttijd is de functionaliteit die voor de adviezen zorgt en maakt gebruik van een database waarin alle artikelen staan die beschikbaar zijn voor het betreffende station, dit is het tweede woord voor de NGD. Verder zijn er nog het verkrijgen van reisinformatie, het melden van vertragingen met alternatieve reisopties en het krijgen van een looproute. Deze functionaliteiten zijn niet verder uitgewerkt in dit rapport en dus buiten beschouwing gelaten.

#### 4. Hoe moeten de functionele eisen op semantische wijzen gaan samenwerken?

Deze vraag is beantwoord met het technisch ontwerp waarin alle onderdelen beschreven staan die nodig zijn om een advies te generen voor de gebruiker op semantisch basis. Om dit hoofdstuk te maken is er per onderdeel van de reisassistent beschreven hoe deze gaat functioneren en waar deze zijn informatie vandaan haalt. Om dit duidelijk te maken is er gebruik gemaakt van de Case om te bepalen per functionaliteit hoe deze werkt. Tevens is er gebruik gemaakt van UML modellen om schematisch weer te geven hoe de reisassistent gaat werken.

Het antwoord op deze deelvraag is dat door middel van het profiel, dat de gebruiker invoert met zijn hobby's en favoriete voedingswaar, en de database waarin alle mogelijke artikelen staan van een station er wordt bepaald wat het beste advies is voor de gebruiker. Met deze twee gegevens wordt er een NGD waarde gegenereerd die er voor zorgt dat de twee onderdelen met het grootste semantisch verband boven aan de adviezen komen te staan. De gebruiker kan dan kiezen uit het advies welke hem het meeste aanspreekt en deze wordt dan uitgebreider getoond in de reisassistent.

## **8.2 Antwoord hoofdvraag**

### Hoe moet een reisassistent functioneren om op semantische wijze rekening te kunnen houden met de verschillende aspecten van een treinreis.

Door middel van de deelvragen wordt ook de hoofdvraag beantwoord. Maar er is nog niet bewezen dat de gekozen techniek van de NGD ook daadwerkelijk kan werken in een applicatie. Om dit te bewijzen is er een Proof of Concept opgesteld waarin wordt aangetoond dat het mogelijk is om een semantisch verband automatisch te bepalen. In het POC wordt er een simpele database gemaakt met 100 mogelijke artikelen. In een tekstveld kan de gebruiker een hobby of voedingswaar invullen die wordt gebruikt voor de NGD formule. De woorden uit de database worden één voor één afgegaan met het ingevulde woord door de gebruiken en de kleinste NGD waarde wordt getoond aan de gebruiker.

Tijdens het maken van het POC zijn er een aantal bevindingen gedaan die belangrijk kunnen zijn voor het uitwerken van de echte reisassistent. Zo beperkt Google het gebruik van de broncode door externe applicaties en kennen Yahoo en Google een limiet van zoekqueries op één dag van één IP adres. Wel is met het POC bewezen dat de NGD methode daadwerkelijk kan werken om automatisch een semantisch verband te bepalen tussen twee woorden waarvan één door een gebruiker is ingevoerd en één uit een database komt.

Om de hoofdvraag in zijn geheel te beantwoorden:

De aspecten van de treinreis zijn bepaald als aan de hand van het onderzoek en het verkrijgen van informatie over wat te doen in wachttijd is op nummer twee geëindigd als meest belangrijke reiaspecten. Hieruit is de functionaliteit wachttijd plannen geformuleerd die er voorzorgt dat er een persoonlijk advies komt waarmee de gebruiker zijn wachttijd kan vullen. Dit advies wordt op semantische wijze samengesteld aan de hand van de NGD formule. Met behulp van een profiel en een database met artikelen van betreffende stations wordt er een semantisch verband berekend. Aan de hand van dit verband komt er een advies met het grootste semantisch verband tussen de twee woorden

## **8.3 Volledige reisassistent**

Zoals vermeld in de Case worden de andere functionaliteiten uitgewerkt door andere projecten binnen Logica. Als het mogelijk blijkt te zijn om al deze projecten samen te voegen tot één applicatie wordt het mogelijk om een reisassistent te creëren die alle functionaliteiten bezit die beschreven worden.

Het verkrijgen van reisinformatie bestaat al in de vorm van bijvoorbeeld de reisplanner Xtra van de NS en de 9292ov applicatie van 9292ov.nl. Om er voor te zorgen dat de reisassistent ook vertragingen kan melden kan er gebruik gemaakt worden van het onderzoek 'De NS travel agent' (42). Om de looproute te maken voor de reisassistent kan er gebruik gemaakt worden van het project 'A framework for context-aware applications using augmented reality' (43). Samen met dit project zouden al deze projecten een reisassistent vormen die geheel kan functioneren volgens de beschrijving in de Case.

## 9 Discussiepunten & Actiepunten

Tijdens het uitvoeren van dit project zijn er een aantal bevindingen gedaan die niet zijn uitgewerkt. Hieronder worden deze punten beschreven per onderdeel waar zij invallen. Er wordt begonnen met discussie punten voor het project, hierna problemen en als laatste aanbevelingen die aan de hand van dit project worden gedaan.

### 9.1 Discussie punten

#### Doelgroep specificatie

Tijdens het onderzoek is er gebruik gemaakt van de doelgroep zakelijke reiziger. Hoewel deze doelgroep samen met de opdrachtgever is opgesteld, is het nagelaten om de doelgroep verder te specificeren. Hierdoor is er gewerkt met een brede doelgroep. De doelgroep had nog verder uitgewerkt moeten worden met bijvoorbeeld een leeftijdscategorie.

#### Valide onderzoek

Voor het onderzoek is er een enquête uitgezet en zijn er interviews afgenomen. In totaal zijn er voor de enquête 24 respondenten vergaard en er zijn twee mensen geïnterviewd. Hoewel dit op zich geen slechte aantallen zijn is het voor een doelgroep als 'zakelijke reiziger' geen valide steekproef. Om een betrouwbaarheidsniveau te krijgen van 95% over een doelgroep van 50.000 mensen, wat een kleine hoeveelheid van de zakelijke reiziger zou zijn, is een totaal aantal respondenten nodig van 382 (44). Dit is nooit te halen in de beschikbare tijd maar een totaal aantal mensen van 26 is geen valide steekproef van de doelgroep 'zakelijke reiziger'.

#### Zal NGD altijd werken?

De NGD is een manier om semantiek te bereiken aan de hand van een formule. Hierdoor is de precieze uitkomst tussen twee woorden nooit bekend. Dus dan ontstaat de vraag of de NGD altijd betrouwbaar is om een advies op te baseren. Uit het POC bleek al dat de combinatie 'Melk' met de database een andere uitkomst had dan verwacht. Of dit te wijten is aan Yahoo of de formule is niet bekend.

### 9.2 Actiepunten

#### Aantal zoekquers Google/Yahoo

Tijdens het maken van het POC is er ondervonden dat Google alleen gebruikt kan worden door een externe applicatie met behulp van hun API. Hierdoor is er gebruikt gemaakt van Yahoo voor het POC. Bij nader onderzoek op de Google API is er gevonden dat zelfs met de API van Google het aantal gratis zoekquery is gelimiteerd tot 1000 per 24 uur (45). Indien er dus op grootschalig gebruik gemaakt moet worden van Google zal er contact opgenomen moeten worden met Google.

#### Aantal zoekopdrachten

In de Case is deze complicatie al benoemd en behandeld, daarom wordt de complicatie in dit stuk kort samengevat. Het gaat hier om aantal zoekopdrachten en de tijd die er kwijt is aan het verwerken van de zoekopdracht. De tijd die nodig is voor het uitrekenen van alle adviezen is onacceptabele indien er bij het generen van elk advies alle berekeningen opnieuw uitgevoerd moeten worden.

Om dit euvel te overkomen is er een matrix oplossing ontworpen waarin alle zoekopdrachten en hun NGD waarde worden opgeslagen. Deze matrix kan gebruikt worden door de reisassistent en dient als een index. Hierdoor hoeven de berekeningen niet elke keer opnieuw uitgevoerd te worden en is de rektijd voor adviezen een acceptabele tijd.

#### Gebruikte taal

Taal en Google zijn belangrijke aspecten bij de reisassistent maar het brengt ook complicaties met zich mee. De complicatie die hier besproken wordt zijn de verschillende talen die in de wereld worden gebruikt en die ook gebruikt worden op het internet. De reisassistent is in het Nederlands en wordt ook in Nederland gebruikt maar Google kan door het gehele internet zoeken. Hierdoor kunnen zoekresultaten anders terug komen dan wanneer alleen door Nederlandse pagina's wordt gezocht. Een voorbeeld hiervan is het woord *das*, in het Nederlands heeft dit woord een aantal betekenissen maar in het Duits is het een lidwoord. Als met dit woord door het gehele internet wordt gezocht kan dit meer pagina's opleveren dan bij bijvoorbeeld het woord *deur*, wat alleen in het Nederlands voor komt.

Waar ook rekening mee gehouden moet worden zijn het aantal woorden die in Nederland zijn ingeburgerd maar oorspronkelijk uit andere talen komen, woorden als: *Snowboard*, *Crème fraîche* of *Rottweiler*. Hoewel deze woorden in het Nederlands dagelijks gebruikt worden kunnen ze resultaten genereren die niet accuraat zijn. Als op het gehele internet wordt gezocht kan het er voor zorgen dat ook pagina's uit andere talen meegenomen worden.



Dit probleem lijkt eenvoudig opgelost door bij Google aan te geven alleen door pagina's te zoeken die in het Nederlands zijn. Toch is dit geen garantie dat alle resultaten ook daadwerkelijk alleen Nederlandse pagina's zijn. Tevens kan het zo zijn dat sommige pagina's de Nederlandse versie van het woord gebruiken. Zo vertaald Crème fraîche zich naar verse room en het is niet onwaarschijnlijk dat sites deze Nederlandse term gebruiken in plaats van de Franse. Tijdens het Proof of Concept zal er geen rekening gehouden worden met dit euvel maar indien verdere ontwikkeling van de reisassistent gerealiseerd wordt zal hier aandacht aan besteed moeten worden.

#### Doelgroep specificeren

In de paragraaf discussie punten staat dit punt genoemd waarna er een aanbeveling wordt gedaan betreffende dit punt. Als er in de toekomst meer onderzoek gedaan wordt naar de doelgroep 'zakelijke reiziger' is er een aanbeveling om deze specifieker te maken dan hij nu is. Dit zal gedaan moeten worden in samenwerking met de NS en andere betrokken partijen.

#### Meer gebruikeronderzoek

In de discussie punten staat dit punt ook genoemd en hier wordt er een aanbeveling gedaan over dit punt. Om een duidelijker beeld te krijgen van de wensen van de doelgroep zou er een uitgebreider onderzoek gedaan moeten worden naar een gespecificeerde versie van de doelgroep 'zakelijke reiziger'. Hiermee kan op een meer valide manier worden aangetoond wat de wensen zijn van de doelgroep en hoe deze verwerkt kunnen worden in de reisassistent.

#### NGD resultaten analyseren

Omdat de NGD waarde gebruikt gaat worden als basis voor een advies zal er onderzoek gedaan moeten worden naar welke uitkomsten de NGD formule terug geeft. Zoals het POC aangeeft is 'melk' al anders dan verwacht. Om er voor te zorgen dat de reisassistent adequaat gaat functioneren, zal er onderzoek gedaan moeten worden naar grootschalige resultaten van de NGD.

#### Onderzoek Java/Java ME

Om de reisassistent platform onafhankelijk te maken was één van de redenen om te kiezen voor Java. Wel moet erop gelet worden dat Java op een smartphone een andere versie kan zijn dan de normale Java zijnde Java ME (46). ME is een uitgekleden versie van de standaard Java en zal dus minder functies bevatten. Om er voor te zorgen dat de reisassistent functioneert zal er gekeken moeten worden naar het ontwerp van de reisassistent en wat er mogelijk is in Java ME.

#### Spelling van woorden

Tijdens het uitwerken van de POC is er ook ondervonden dat de spelling van woorden invloed kan hebben op de uitkomsten van de NGD waarde. Hier moet verder onderzoek naar gedaan worden hoe hier adequaat mee om te gaan om de werking van de reisassistent te garanderen.

#### RDF OWL chocolade

In het POC komt naar voren dat melk en chocolade al een groter verband vormen dan melk en zuivel. In dit geval is het dus beter om een RDF/OWL database te hebben die hier het onderscheid wel kan maken. Dit kan opgelost worden door de NGD waarden van chocolade en melk of zuivel en melk aan te passen in de matrix. Voordat dit gedaan wordt zal er onderzoek gedaan moeten worden naar welke combinaties onverwachte hoge waarden geven en wat hier de beste oplossing voor is.



## 10 Bronnenlijst

1. **Verhoeven, Nel.** *Wat is Onderzoek*. Amsterdam : Boom onderwijs, 2008.
2. **Stone, Debbie, et al.** *User Interface Design and Evaluation*. San Francisco : Morgan Kaufmann, 2005.
3. **van Steenis, H.** *Computable ICT woordenboek*. 2003.
4. **Noord, G.** Semantische interpretatie en ambiguïteit. [Online] 1998. <http://www.let.rug.nl/~vannoord/papers/breinmakers/node7.html>.
5. **Gasevic, Dragan, Djuric, Dragan en Devedzic, Vladan.** *Model Driven Engineering and Ontology Development*. London : Springer, 2008.
6. **van Dijk, Daniel.** *Designing a Semantic Web-Based Poster Picture Advisor*. Rotterdam : sn, 2008.
7. **Hulshof, Hans.** *Raden met een bedoeling: ambiguïteit in beeld*. Amsterdam : Universiteit van Amsterdam, 2005.
8. **Koolwaaij, Johan en Strating, Patrick.** *Service Framework for Mobile Context-aware Applications*. Enschede : sn, 2005.
9. **Dey, Anind en Abowd, Gregory.** *Towards a Better Understanding of Context and Context-Awareness*. London : sn, 1999.
10. **Schilit, Bill, Adams, Norman en Want, Roy.** *Context-Aware Computing Applications*. sl : 1st International Workshop on Mobile Computing Systems and Applications., 1994.
11. **Cilibrasi, Rudi en Vitanyi, Paul.** *Automatic Meaning Discovery Using Google*. Amsterdam : Universiteit van Amsterdam, 2007.
12. **de Kunder, Maurice.** The size of the World Wide Web. [Online] oktober 2009. <http://www.worldwidewebsite.com/>.
13. **Cilibrasi, Rudi en Vitanyi, Paul.** *The Google Similarity Distance*. Amsterdam : sn, 2007.
14. **van Harmelen, Frank, et al.** *Using Google Distance to weight approximate ontology matches*. Eindhoven : sn, 2008.
15. **Evangelista, Alberto en Kjos-Hanssen, Bjørn.** *Google Distance Between Words*. Connecticut : sn, 2006.
16. **Guiaux, Jelmer en Scheers, Marco.** *Semantic search*. Zoetermeer : sn, 2009.
17. **Van der Pals, Eddy.** *Semantic search voor Business Intelligence*. Rotterdam : sn, 2009.
18. **Pepping, Gerard.** *Het semantic web*. Groningen : sn, 2009.
19. **Berners-Lee, Tim.** Tim Berners-Lee Bio. [Online] 2009 . <http://www.w3.org/People/Berners-Lee/>.
20. **van Harmelen, Frank.** Frank van Harmelen personal website. [Online] 2009. <http://www.cs.vu.nl/~frankh/>.
21. **Luickx, Philippe.** *Toepassing van Semantisch Web technologie voor het beheer van beleidsinformatie binnen de Vlaamse Overheid*. Leuven : sn, 2008.
22. **van Harmelen, Frank.** Lecture Web Ontology Languages. sl : [http://videolectures.net/koml04\\_harmelen\\_o/](http://videolectures.net/koml04_harmelen_o/), 2004.
23. **W3C.** RDF/XML Syntax Specification (Revised). [Online] 2004. <http://www.w3.org/TR/rdf-syntax-grammar/>.
24. **Antoniou, Grigoris en van Harmelen, Frank.** *A semantic web primer*. London : MIT Press, 2008.
25. **Babylon Translation Software.** Babylon 8 Translation Software and Dictionary Tool. [Online] 2008. <http://www.babylon.com/>.
26. **Passin, Thomas.** *Explorer's Guide to the Semantic Web*. Greenwich : Manning, 2004.
27. **W3C.** RDF Vocabulary Description Language 1.0: RDF Schema. [Online] 2004. <http://www.w3.org/TR/rdf-schema/>.

28. —. OWL Web Ontology Language. [Online] 2004. <http://www.w3.org/TR/2004/REC-owl-features-20040210/>.
29. **AllSites LLC**. Online Dictionary, Language Guide, Foreign Language and Etymology. [Online] 2009. <http://www.allwords.com/>.
30. **Segaran, Toby, Evans, Colin en Taylor, Jamie**. *Programming the Semantic Web*. Sebastopol : O'Reilly, 2009.
31. **Algemeen Dagblad**. AD Binnenland. [Online] 2009. <http://www.ad.nl/ad/nl/1012/Binnenland/article/detail/446659/2009/11/26/Gratis-internet-in-de-trein.dhtml>.
32. **Nu.nl**. [Online] 2009. <http://www.nu.nl/internet/1345120/toch-draadloos-internet-in-de-trein.html>.
33. **Kenny, David en Marshall, John F**. *Contextual Marketing: The Real Business of the Internet*. sl : the Harvard Business Review, 2000.
34. **Stone, Bob en Jacobs, Ron**. *Successful Direct Marketing Methods*. New York : McGraw-Hill Companies, 2008.
35. **Hoffman, Donna F en Novak, Thomas P**. *How To Acquire Customers From The Web*. New York : Harvard Business Review, 2000.
36. **Jansen, Bernard en Resnick, Marc**. *An Examination of Searcher's Perceptions of Nonsponsored and Sponsored Links During Ecommerce Web Searching*. : JASIST, 2006.
37. **Haagse Hoge School**. *Business Alignment Reader*. Den Haag : Haagse Hoge School, 2007.
38. **Cockburn, Alistair**. *Writing Effective Use Cases*. Indianapolis : Addison-Wesley, 2002.
39. **Berrisford, Graham en Lankhorst, Marc**. *Using ArchiMate with an Architecture Method*. sl : Via Nova Architectura, 2009.
40. **Lankhorst, Marc, et al**. *ArchiMate verbindt architectuurdomeinen*. sl : Informatie, 2004.
41. **Servex**. [Online] Servex BV, 2009. <http://www.lekkerwerkenophetstation.nl/web/Servex/ServexBV.htm>.
42. **Hathie, Patrick**. *De NS Travel Agent*. sl : Logica, 2009.
43. **Uijtdewilligen, Freek**. *A framework for context-aware applications using augmented reality*. Twente : Universiteit van Twente, 2009.
44. **journalinks**. <http://www.journalinks.be/>. [Online] 2009. <http://www.journalinks.be/steekproef/?margin=5&confidence=95&population=4+233+861&response=50&sample1=100&sample2=200&sample3=300&confidence1=90&confidence2=95&confidence3=99>.
45. **Google**. Google SOAP Search API. [Online] 2009. [http://code.google.com/intl/nl/apis/soapsearch/api\\_faq.html](http://code.google.com/intl/nl/apis/soapsearch/api_faq.html).
46. **Sun**. Java ME. [Online] 2009. <http://java.sun.com/javame/index.jsp>.

## 11 Afbeeldingenlijst

Afbeelding 1 semantiek voorbeeld 1 .....	12
Afbeelding 2 semantiek voorbeeld 2 .....	13
Afbeelding 3 Lexicale ambiguïteit .....	19
Afbeelding 4 Semantisch web opbouw .....	22
Afbeelding 5 Voorbeeld triple .....	23
Afbeelding 6 Voorbeeld RDFS .....	23
Afbeelding 7 Voorbeeld Protégé .....	24
Afbeelding 8 Plaatsing Jena .....	25
Afbeelding 9 Verdeling enquête resultaten .....	26
Afbeelding 10 Eerste schets reisassistent .....	29
Afbeelding 11 Google maps .....	33
Afbeelding 12 Use Case diagram .....	41
Afbeelding 13 Architectuur reisassistent .....	51
Afbeelding 14 Klassendiagram .....	53
Afbeelding 15 SSD Invullen profiel .....	54
Afbeelding 16 FUI Invullen profiel .....	55
Afbeelding 17 ESSD Invullen profiel .....	55
Afbeelding 18 EFUI Invullen profiel .....	56
Afbeelding 19 SSD Plannen wachttijd .....	57
Afbeelding 20 FUI Plannen wachttijd .....	57
Afbeelding 21 ESSD Plannen wachttijd .....	58
Afbeelding 22 EFUI Plannen wachttijd .....	59
Afbeelding 23 POC database .....	63
Afbeelding 24 Invoer POC .....	67
Afbeelding 25 Geen invoer POC .....	67
Afbeelding 26 Uitvoer POC .....	68

## 12 Tabellenlijst

Tabel 1 Overwegingen conventionele technieken/NGD .....	28
Tabel 2 Gemiddelde rektijd Google .....	35
Tabel 3 Matrix voorbeeld 1 .....	36
Tabel 4 Uitwerking Profiel invullen .....	42
Tabel 5 Uitwerking Plannen wachttijd .....	44
Tabel 6 BCL reisassistent .....	44
Tabel 7 BCL POC .....	45
Tabel 8 Functionaliteiten naar Use Cases .....	45
Tabel 9 Matrix voorbeeld 2 .....	50
Tabel 10 BCL POC 2 .....	62

## 13 Afkortingenlijst

Afkorting	Volledig
AHTG	Albert Heijn To Go
API	Application Programming Interface
BCL	Basis Comfort Luxe
EFUI	Extended Functional User Interface
ESSD	Extended System Sequence Diagram
FO	Functioneel ontwerp
FUI	Functional User Interface
GPS	Global Positioning System
GUI	Graphical user interface
NGD	Normalized Google Distance
NS	Nederlandse Spoorwegen
OV	Openbaar Vervoer
OWL	Web Ontology Language
PID	Project Initiating Document
POC	Proof of Concept
RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
SPARQL	SPARQL Protocol and RDF Query Language
SQL	Structured Query Language
SSD	System Sequence Diagram
TO	Technisch ontwerp
UCD	Use Case Diagram
UML	Unified Modeling Language
VVV	Vereniging voor Vreemdelingenverkeer
W3C	World Wide Web Consortium
XML	Extensible Markup Language

## 14 Register

### 3

3G · 37

### A

Affiliate marketing · 34, 37  
Ambigüiteit · 18, 19, 27  
Antoniem · 24  
Application Programming Interface · 62, 79  
ArchiMate · 51  
Automatic contextuel reconfiguration · 20

### B

Bedrijfsperspectief · 33, 34  
Bezettingsgraad · 11  
brute force · 27

### C

Celsius · 20  
Context · 18, 19, 20, 27, 28, 37, 45, 49, 71  
Context-aware · 20  
Context-triggered · 20  
Contextual command · 20  
Contextual Marketing · 34

### D

Detaillering niveau · 34  
Direct marketing · 34  
doelgroep · 8, 16, 17, 18, 26, 46, 50, 70, 72, 73

### E

Enquête · 10, 16, 17, 26, 46, 70, 72, 85  
Etymologieën · 24  
Extended Functional User Interface · 54  
Extended System Sequence Diagram · 54  
Extensible Markup Language · 21, 79

### F

Fahrenheit · 20  
Folksonomie · 19, 22, 27  
Frank van Harmelen · 21, 22, 23, 27  
Functional User Interface · 54, 79

### G

Gebruikersperspectief · 33, 34, 41  
Gestructureerde data · 22  
Global Positioning System · 36, 79  
Google · 3, 8, 20, 21, 28, 29, 32, 34, 35, 36, 45, 60, 62, 66, 68, 70, 71, 72, 73, 79  
Graphical User Interface · 49

### H

Hyper Text Markup Language · 21  
Hyperoniem · 24  
Hyponiem · 24

### I

Interview · 17, 26

### J

Java · 24, 25, 28, 29, 37, 48, 52, 63, 64, 66, 68, 73, 83  
Java ME · 48, 73  
Jena · 25, 27, 28, 88

### K

Klasse · 19, 23, 24, 48, 52, 53  
Kassendiagram · 52  
Kunstmatige intelligentie · 18

### L

Laptop · 11, 16, 17, 29, 32, 37  
lexicale ambigüiteit · 19

### M

Matrix · 35, 36, 38, 45, 49, 50, 53, 72  
Meronym · 24  
MySQL · 63

### N

Negatieve correlatie · 21  
NGD formule · 8, 9, 28, 52, 60, 62, 63, 66, 70, 71, 73  
NGD waarde · 9, 21, 36, 46, 49, 50, 60, 62, 66, 67, 68, 71, 72, 73

Normalized Google Distance · 3, 8, 20, 79

---

## O

Object · 22  
 Ontologie · 18, 19, 21, 23, 24, 27, 28  
 OWL DL · 23  
 OWL Full · 23  
 OWL lite · 23

---

## P

Persona · 17, 32, 38, 70  
 Predicaat · 22  
 Primaire context types · 20  
 Protégé · 24, 25, 27, 28  
 Protégé-Frames · 24  
 Protégé-OWL · 24  
 Proximate selection · 20

---

## R

Resource Description Framework · 21, 22, 79  
 Resource Description Framework Schema · 23

---

## S

Secundaire context types · 20  
 Semantiek · 8, 12, 13, 18, 20, 21, 22, 23, 24, 25, 26, 27, 28, 36, 60, 70, 72, 88  
 Semantisch verband · 8, 9, 21, 60, 62, 66, 69, 70, 71  
 Semantische web · 18, 21, 22, 23, 25, 27  
 Servicegraad · 8, 11  
 Smartphone · 8, 16, 29, 36, 37, 48, 49, 73

Subject · 22  
 Synoniemen · 21  
 Syntactische ambiguïteit · 19  
 System Sequence Diagram · 54, 79

---

## T

Taxonomie · 19, 20, 27, 34, 36  
 The Open Group · 51  
 Tim Berners-Lee · 21, 27

---

## U

Unified Modeling Language · 54, 79  
 Use Case · 41, 42, 43, 44, 45, 54, 57, 79

---

## W

Web Ontology Language · 21, 79  
 WordNet · 24, 28  
 World Wide Web Consortium · 21, 79

---

## Y

Yahoo · 62, 66, 68, 71, 72, 88

---

## Z

Zakelijke reiziger · 8, 16, 17, 37, 50, 72, 73  
 Zoekquery · 35, 66, 72



---

## Bijlage

## A. Personas

### Persona 1

**Naam:** Tim de Vries



**Leeftijd:** 41 jaar

**Woonplaats:** Voorburg

**Opleiding:**

Tim heeft een lang opleidingstraject achter de rug. Hij is van de MAVO naar het MBO gegaan en uiteindelijk heeft hij zijn bachelor in Informatica gehaald aan het HBO. Daarna heeft hij nog een aantal cursussen gedaan in Java en Oracle.

**Huwelijke status:** Twee keer gescheiden

**Huidige functie:**

Tim is op dit moment werkzaam als consultant bij een grote ICT dienstverlener

**Toelichting werk:**

Tim werkt als consultant waarbij hij naar verschillende bedrijven moet om daar te werken aan de bestaande problemen. Hij zit in een team waarin mensen zitten met verschillende specialiteiten. Tim doet zijn werk al drie jaar en is er tevreden over zijn functie en de voldoening die hij hieruit haalt. Wel is hij lichtelijk een workaholic waardoor hij lange uren spendeert aan zijn werk.

**Overige vaardigheden:**

Tim werkt aan een aantal open source projecten mee. Hiermee vult hij zijn vrije tijd en probeert hij zijn programmeer vaardigheden verder uit te werken. Verder heeft Tim een interesse in fotografie waarin hij ook nog een aantal cursussen heeft gevolgd. Hij kan inmiddels zijn eigen foto's bewerken en verspreiden op een professionele manier.

**Reizen:**

In het verleden is Tim betrokken geweest bij een ernstig auto ongeluk. Hierdoor is Tim niet meer in staat om zonder angst in een auto te zitten. Tevens durft hij hierdoor zelf niet meer te rijden. Tim maakt veel gebruik van het OV om te reizen naar de verschillende werkplekken te reizen

**Overige informatie:**

Tim is zeer fanatiek bezig met zijn werk, vandaar dat hij al twee keer is gescheiden. Hij heeft echter uit alle twee de huwelijken 1 kind over gehouden, en heeft nu dus 1 dochtertje van 8, en een zoon van 5 jaar oud. Als Tim niet bezig is met zijn werk houdt hij van wielrennen, of pokeravonden met oude vrienden.

**Persona 2****Naam:** Lisa Peperoni**Leeftijd:** 28 jaar  
**Woonplaats:** Rotterdam**Opleiding:** Lisa komt oorspronkelijk uit Italië en heeft daar de middelbare school gedaan. In Nederland heeft ze eerst intensief Nederlands gestudeerd, daarna heeft ze communicatiewetenschappen gestudeerd. Haar studie heeft ze nog aardig wat cursussen gedaan in allerlei verschillende richtingen zoals praktijk diploma boekhouden en cursussen voor webdesigners**Huwelijke status:** Lisa woont op dit moment samen met Lucano, maar niemand weet echt zeker of ze nog steeds officieel bij elkaar zijn. Op dit moment is het een knipperlicht relatie, wat erg veel tijd kost van Lisa.**Huidige functie:**

Lisa is op dit moment vertegenwoordiger voor een distributeur van merk-kleding

**Toelichting werk:**

Na een aantal verschillende baantjes kwam Lisa erachter dat haar echte passie bij kleding ligt. Hierdoor is ze op zoek gegaan naar een baan waar ze constant met haar favoriete kleding in aanraking komt. Deze heeft ze gevonden en samen met haar opleiding weet ze een goede invulling te geven aan haar functie. Momenteel is ze van plan om lang bij deze werkgever te blijven maar volgt wel regelmatig een cursus om haar opties open te houden.

**Overige vaardigheden:**

Lisa is goed met internet maar haar verdere kennis van computers is echter zeer beperkt. Naast haar normale werk heeft ze soms opdrachten als fotomodel. Verder is Lisa erg goed in het communiceren met verschillende soorten mensen waardoor ze uitstekend kan functioneren in haar huidige baan.

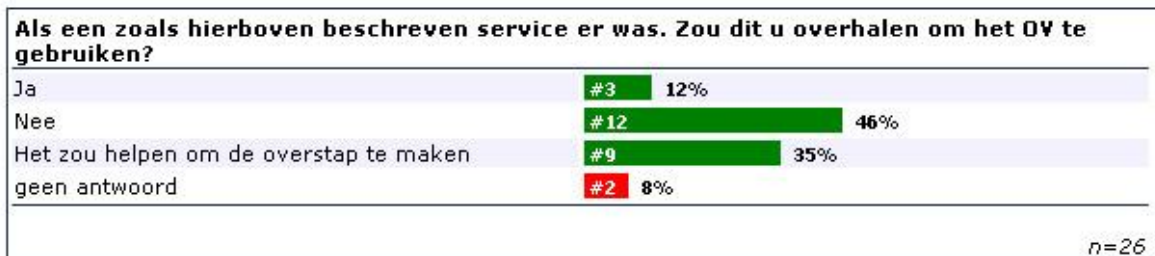
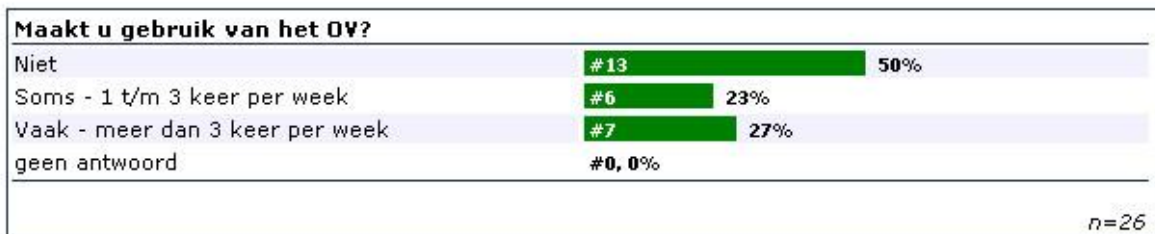
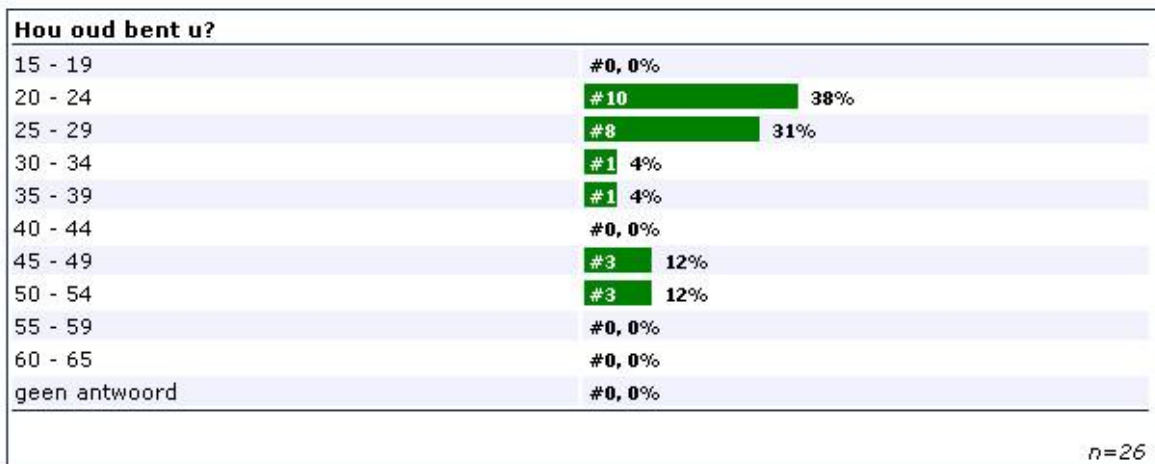
**Reizen:**

Lisa reist veel voor haar werk maar dit doet ze altijd met de auto. In haar studenten tijd heeft ze wel van het OV gebruik gemaakt maar hier was ze nooit echt tevreden over. Wel is ze milieubewust dus als ze de auto kan laten staan doet ze dat.

**Overige informatie:**

Lisa is zeer ambitieus, het liefst zou ze zo snel mogelijk beginnen met het opzetten van haar eigen bedrijf, maar ze weet dat het verstandiger is om eerst werkervaring op te doen. Naast haar werk heeft ze veel tijd nodig voor haar relatie, als ze echter vrije tijd heeft gaat ze vaak met vriendinnen een dagje weg naar bijvoorbeeld de sauna.

## B. Resultaten enquête



**Welke functionaliteiten moet de service bieden voordat u gebruik gaat maken van het OV? (MAX 4 kiezen)**

Bijhouden voortgang van de reis	#10	13%
Alternatief aangeven in geval van vertragi...	#24	32%
Waarschuwen wanneer u moet gaan inpakken	#6	8%
Advies op maat tijdens wachttijd	#14	18%
Plattegrond met de te lopen route	#13	17%
Welke activiteiten er te doen zijn op en r...	#6	8%
Tijd die een activiteit kan kosten	#3	4%
anders:	#0	0%
geen antwoord	#0	0%

n=76

**Wilt u alleen activiteiten op en rond het station of ook verdere gelegenheden?**

Alleen op het station	#6	23%
Daarbuiten maar op loopafstand	#16	62%
Daarbuiten maar op bus/tram/metro afstan...	#4	15%
geen antwoord	#0	0%

n=26

**Wat voor soort activiteiten? (MAX 4 kiezen)**

Restaurant	#15	17%
Supermarkt	#17	19%
Rookgelegenheden	#3	3%
Boek/tijdschriften/krant verkoop	#14	16%
Multimedia verkoop	#10	11%
Horeca	#16	18%
Sport en spel	#5	6%
Museum	#5	6%
anders:	#2	2%
geen antwoord	#0	0%

n=89

**Als u vertraging heeft, hoe wordt u dan nu geïnformeerd?**

Omroepberichten	#20	39%
Vertrekborden	#16	31%
Conducteur	#3	6%
Medepassagiers	#2	4%
Niet	#2	4%
anders:	#4	8%
geen antwoord	#0	0%

n=51

**Welke informatie over de vertraging krijgt u nu? (MAX 4 kiezen)**

Duur van vertraging	#17	45%
Wat zijn de alternatieve reisopties	#4	11%
Wat is de snelste alternatieve optie	#0	0%
Wat voor soort trein is mijn alternatieve ...	#1	3%
Wat kan ik doen in mijn wachttijd	#0	0%
Kom ik nog op tijd op mijn afspraak	#0	0%
Oorzaak van vertraging	#7	18%
anders:	#4	11%
geen antwoord	#0	0%

n=38

**Welke informatie over de vertraging wilt u weten? (MAX 4 kiezen)**

Duur van vertraging	#19	21%
Wat zijn de alternatieve reisopties	#16	18%
Wat is de snelste alternatieve optie	#24	27%
Wat voor soort trein is mijn alternatieve ...	#3	3%
Wat kan ik doen in mijn wachttijd	#5	6%
Kom ik nog op tijd op mijn afspraak	#14	16%
Oorzaak van vertraging	#9	10%
anders:	#0	0%
geen antwoord	#0	0%

n=90

**Zou u bereid zijn om te betalen voor een dergelijke service?**

Ja, een éénmalig bedrag van max 3 euro	#4	15%
Ja, een éénmalig bedrag van max 6 euro	#1	4%
Ja, een éénmalig bedrag van max 10 euro	#2	8%
Ja, een maandelijks bedrag van max 1 euro	#1	4%
Ja, een maandelijks bedrag van max 3 eur...	#2	8%
Ja, een maandelijks bedrag van max 5 eur...	#1	4%
Nee	#15	58%
geen antwoord	#0	0%

n=26

---

## C. Lijst semantisch applicaties

Hieronder is de lijst opgesteld met alle tot de schrijver bekende applicaties die zeggen semantiek te gebruiken.

- AdaptiveBlue
- Aduna
- Clear Forest
- Cognition Search
- Exalead
- Factbites
- Freebase
- Friend of a friend
- Hakia
- Jena
- Kosmix
- Lexxe
- Powerset
- SenseBot
- Spock
- Swoogle
- Talis
- Triplt
- TrueKnowledge
- Twine
- Wordnet
- Yahoo searchmonkey

