

### **Team 1**

Mustapha Bouzaidi  
Isabelle Hanraads  
Sherwin Angelo  
Kevin Wong

# Inhoudsopgave :

## Inhoud

Missie en niet functionele requirements.....	3
Rollen.....	3
Project Plan.....	3
Glossary.....	4
Analyse:.....	8
Actor List.....	8
Use Case diagram:.....	9
Requirements.....	10
Het algoritme.....	12
Use case details.....	13
Actor A.I.....	13
Actor Gebruiker.....	14
Use Case List.....	18
Actor A.I.....	18
Record supplementary requirements.....	18
Prioriteiten use case:.....	19
User interface sketches:.....	20
Class diagram:.....	21
Sequence diagram:.....	22
State diagram:.....	27

# Genesis:

## Missie en niet functionele requirements

Het ontwerpen en bouwen van een taxisysteem op het spoor. Dit systeem zal zonder conducteurs, volledig automatisch, reizigers comfortabel van vertrek naar bestemmingspunt brengen. Taxi's kunnen aan elkaar koppelen om energie te besparen. Ook zal het systeem in de gaten houden dat er geen botsingen kunnen ontstaan.

## Rollen

Om het project zo georganiseerd mogelijk af te ronden worden er projectrollen aan iedere projectlid toegewezen. Dit wordt gedaan om het project binnen de afgesproken tijd te voltooien.

De projectrollen kunt u hieronder bezichtigen:

Rol	Wie
<b>Project manager</b>	Mustapha Bouzaidi
<b>Tijdsbewaker</b>	Isabelle Hanraads Sherwin Angelo
<b>Notulist</b>	Rotatie van het gehele team
<b>Glossariumbewaker</b>	Sherwin Angelo

## Project Plan

De planning ziet er op het moment als volgt uit:

18-09-07 tot 05-10-07	Projectinitiatie
05-10-07 tot 06-11-07	Analyse
06-10-07 tot 19-11-07	Design
19-11-07 tot 17-12-07	Realisatie
17-12-07 tot 02-01-08	Testen
02-01-08 tot 08-01-08	Oplevering

## Glossary

Naam	Type	Beschrijving
<b>A.I.</b>	Business actor, system actor	Computergestuurde actor dat het taxisysteem bestuurt.
<b>Aanrijding</b>	Business object	Een aanrijding tussen taxi's
<b>ArrayListComboBoxModel</b>	Design object, System object	Deze klasse is een abstracte klasse die alle methodes van de AbstractListModel implementeert Dit wordt voor het paneel klasse gebruikt om een array van de stations te maken.
<b>Baan</b>	Business object	Treinbaan waarop de taxi's rijden.
<b>Baaninstellingen</b>	System object	Instellingen m.b.t. de baan
<b>BaanInstellingen</b>	System object	De BaanManager klasse heeft het overzicht van de baan, het bevat sporen, stations en taxi's. BaanManager voer de volgende functie uit: <div style="margin-left: 40px;">De wisselsporen schakelen en terugschakelen.</div> <div style="margin-left: 40px;">De algemene instellingen aanpassen: snelheid van een taxi en wachttijd bij een station.</div> <div style="margin-left: 40px;">Bepaalt positie voor taxi objecten.</div> Bepaalt reisafstanden voor de ReisManager.
<b>BaanManager</b>	System object	Regelt de stations en spoorwissels op de baan.
<b>BaanManager</b>	System object	De BaanManager klasse heeft het overzicht van de baan, het bevat sporen, stations en taxi's. BaanManager voer de volgende functie uit: , De BaanManager klasse heeft het overzicht van de baan, het bevat sporen, stations en taxi's. BaanManager voer de volgende functie uit: , De wisselsporen schakelen en terugschakelen, De algemene instellingen aanpassen: snelheid van een taxi en wachttijd bij een station, Bepaalt positie voor taxi objecten, Bepaalt reisafstanden voor de ReisManager.
<b>Botsbeveiliging</b>	System object	Klasse voor het vermijden van collisions tussen taxi's.
<b>Controller</b>	System object, Design object	De controller verzorgt de communicatie tussen de systeem klasse en de gebruikers klasse. De communicatie gebeurt d.m.v. gegevens omzetting. Als de UserInterface gegevens nodig heeft om de baan te visualiseren dan worden de gegevens die afkomstig zijn van de Reismager of BaanManager omgezet door de controller klasse zodat het gevisualiseerd kan worden.
<b>Gebruiker</b>	Business actor	Menselijke actor die de simulatie gebruikt.
<b>Instellingscherm</b>	Design object, System object	Deze klasse tekent een paneel in een frame en wordt gebruikt voor de taxi instellingen. Je kan de snelheid van de taxi instellen en de reixigers instaptijd. De snelheid moet tussen 60 en 300 km/u liggen en de instaptijd mag vanaf 2 seconde en

		hoger zijn.
<b>Intelligentie</b>	System object	De object representatie van de A.I. actor in het systeem.
<b>Main</b>	System object	Klasse voor het starten van de applicatie/systeem.
<b>Main</b>	System object, system actor	Deze klasse zal van de klassen <code>UserInterface</code> en <code>ReisManager</code> een instantie maken. Hierdoor is het mogelijk om de simulatie te starten.
<b>Momentopnameladen</b>	Design object, Business object	Hiermee kunt u de bewaarde input van de simulator laden
<b>Momentopnameopslaan</b>	Design object, Business object	Hiermee kunt u de huidige input van de simulator bewaren
<b>Motor</b>	System object	Motor klasse waarmee de motor van de taxi aangestuurd word.
<b>Paneel</b>	Design object	Deze klasse zal alle gegevens op het scherm in een frame tekenen. Een voorbeeld van alle gegevens zijn, taxi's stations, knoppen, baan etc.
<b>Reis</b>	System object	Een Reis object is geassocieerd met de volgende objecten: 1 taxi 1 tot 5 reizigers 2 stations Een reis object bevat via een <code>ReisDetails</code> object een vertrek- en bestemmingspunt. Als een Reis object wordt aangemaakt dan wordt de afstand en tijd naar de vertrekpunt en bestemming berekend.
<b>ReisDetails</b>	System object	Bevat een vertrekpunt en bestemming ( <code>Station</code> objecten). Wordt gedeeld tussen <code>Reiziger</code> objecten en een reis object. Als een reiziger in het systeem wordt gevoerd dan is het niet geassocieerd met een <code>Reisobject</code> , dus wordt een vertrek- en bestemmingspunt van de reiziger in een <code>ReisDetails</code> object opgeslagen. Als een reisobject wordt aangemaakt dan wordt het <code>ReisDetails</code> object van zijn reizigers gebruikt.
<b>ReisManager</b>	System object, System actor	Het planningsalgoritme van het systeem en een actor voor het starten en besturen van reizen. Het algoritme beoordeelt reizigers in het systeem en start (indien nodig) nieuwe reizen.
<b>Reisstatus</b>	Design object	Deze klasse maakt een enumeratie status aan voor een taxi
<b>Reizerstat</b>	Design object, System object	Een Klasse die een paneel in een frame heeft die gebruikt wordt om het aantal taxi en stations in de simulator te voeren.
<b>Reizigerwachtljst</b>	Design object	Deze klasse maakt een berekening van de wachttijd van een reiziger
<b>Reiziger</b>	Business object, System object	Reiziger die een taxi aanvraagt om naar zijn bestemming te reizen.
<b>Reiziger</b>	System object	Representeert een reiziger in het systeem. Een reiziger wordt aangemaakt wanneer de "reiziger invoeren" functionaliteit in de simulatie gebruikt wordt. Een reiziger kan zich verplaatsen van/naar een station of taxi,

		daarnaast houdt het ook zijn wachttijd bij. De reistijd wordt bijgehouden door het geassocieerde Reis object.
<b>ReizigerManager</b>	System object	Regelt de reizigers in de simulatie.
<b>ReizigerStatus</b>	Design object	Deze klasse geeft de status bij van een reiziger
<b>Sensor</b>	System object	Sensor in taxi waarmee positie bepaald wordt.
<b>Simulatie</b>	Business object, System object	Werking van een intelligente taxi systeem op rails wordt op kleine schaal aangetoond.
<b>Simulatie</b>	Design object	Deze klasse is een laatste laag dat boven baanManager en Reismanager zit. Hij heeft hierdoor instanties van de reismanager en de baanmanager klasse. Deze klasse houdt zich voornamelijk bezig met het gemiddelde statistiek berekeningen en de momentopname laden en opslaan.
<b>Spoor</b>	System object	Een onderdeel van de treinbaan. Elk Spoor object heeft een verschillende lengte, bevat een verwijzing naar het volgende spoor object en heeft een attribuut dat het soort spoor aangeeft. Vaak verbindt een spoor uit de hoofdbaan met een wisselspoor of normale spoor. Taxi objecten kunnen zich op Spoor objecten bevinden.
<b>SpoorIterator</b>	Design object	Deze klasse geeft de status van een taxi terug. Of het op een spoor bevindt of op ene station
<b>SpoorType</b>	Design object	Deze klasse maakt een enumeratie van statussen voor een taxi
<b>Spoorwissel</b>	Business object, System object	Een spoorwissel dat een taxi naar een andere spoor kan leiden.
<b>Station</b>	Business object, System object	Een treinstation waarin reizigers taxi's kunnen aanvragen.
<b>Station</b>	System object	Een station object kan reizigers en taxi's bevatten. Een station zit op een Stationspoor object en kan benadert worden wanneer de wisselspoor naar dat station geschakeld wordt. Wanneer de status van een station gewijzigd is dan stuurt de betreffende station een notificatie naar de simulatie.
<b>Stationspoor</b>	System object, Business object	Een Stationspoor heeft alle eigenschappen van een Spoor object en bevat bovendien een Station object.
<b>StationStat</b>	Design object, System object, Business object	Deze klasse toont de station statistieken. De station statusstieken bestaat uit een station naam, reiziger op dat station en taxi op dat station.
<b>Statistieken</b>	Business object, System object	Verzameling statistische gegevens van entiteiten in de simulatie.
<b>Statistieken</b>	System object	Deze klasse zal alle gegevens van Taxi's, Stations en Reizigers opvragen van de ReisManager, er zal binnen de UserInterface klasse een mogelijkheid zijn deze statistieken stuk voor stuk op te vragen.
<b>Taxi</b>	System object	Een Taxi object kan reizigers bevatten en is bovendien geassocieerd met een reisobject dat het huidige vertrek- en bestemmingspunt van de taxi heeft. Een taxi berekent tijdens het reizen de positie op de baan en stuurt een notificatie naar de simulatie als het gewijzigd

		is. Bovendien wordt een notificatie naar de simulatie gestuurd als de status van de taxi is gewijzigd
<b>TaxiManager</b>	System object	Regelt de taxi's op de baan.
<b>TaxiReden</b>	Design object	Deze klasse maakt een enumeratie status aan van taxi voor de botbeveiliging
<b>Taxiroute</b>	Business object	Route dat een taxi doorloopt, naar vertrekpunt en vervolgens naar bestemming.
<b>Taxistat</b>	Design object	Deze klasse toont de taxi statestieken. De taxi statusstieken bestaat uit een taxi status waarbij je kan controlleren of de taxi op een station bevindt of op de spoor, aantal reiziger in taxi, reistijd van taxi, bestemming van de taxi en wachttijd per reiziger in een taxi.
<b>Taxistatus</b>	Business object	Status van een taxi, zoals positie, handeling, enz.
<b>Taxistatus</b>	Design object	Deze klasse maakt een enumeratie status aan voor de taxi. (RIJD, WACHT_OP_SPOOR, WACHT_OP_STATION, ONBEZET, BEZET, GEPAUZEERD_OP_SPOOR)
<b>TekenSpoor</b>	Design object	Deze klasse tekent het spoor aan de hand van de opgegeven waarde, dat tussentwee en acht ligt.
<b>User interface</b>	System object	Gebruikersinterface waarmee de gebruiker de simulatie kan bekijken en beïnvloeden.
<b>UserInterface</b>	System object, Design object, system actor	Houd zich bezig met de visualisatie. Door de UserInterface is de input en output van de simulatie mogelijk. Het functioneert door gebruik te maken van alle klassen die benoemd zijn in dit glossary.
<b>Wisselspoor</b>	System object, Design object	Een Wisselspoor heeft alle eigenschappen van een Spoor object, een extra verwijzing naar een andere spoor en de mogelijkheid om van stand te wisselen. De stand van de wisselspoor geeft aan met welke van de 2 sporen het verbindt (hoofdspoor of stationspoor) als een taxi langsrijdt.

# Analyse:

## Casus:

Movares wil een vervoersysteem dat reizigers comfortabeler van A naar B brengt en de gemiddelde wachttijd en reistijd verminderd. Dit systeem zal op het spoor gerealiseerd worden en in tegenstelling tot het huidige treinvervoer, dynamisch zijn, zoals taxivervoer. Dit houdt in dat er geen vaste vertrektijden meer zijn. Taxivervoer zal alleen aanwezig zijn waar nodig.

De wagons, of treintaxi's zullen een kleine groep mensen vervoeren, van maximaal 5 personen. De treintaxi's zullen alleen op vertrekpunt en eindbestemming stoppen. Reizigers zullen comfortabeler reizen omdat er geen mensen tussendoor in of uit stappen.

De veiligheid van de reizigers is ook van erg groot belang. Er wordt voor gezorgd dat treintaxi's onder geen geval kunnen botsen.

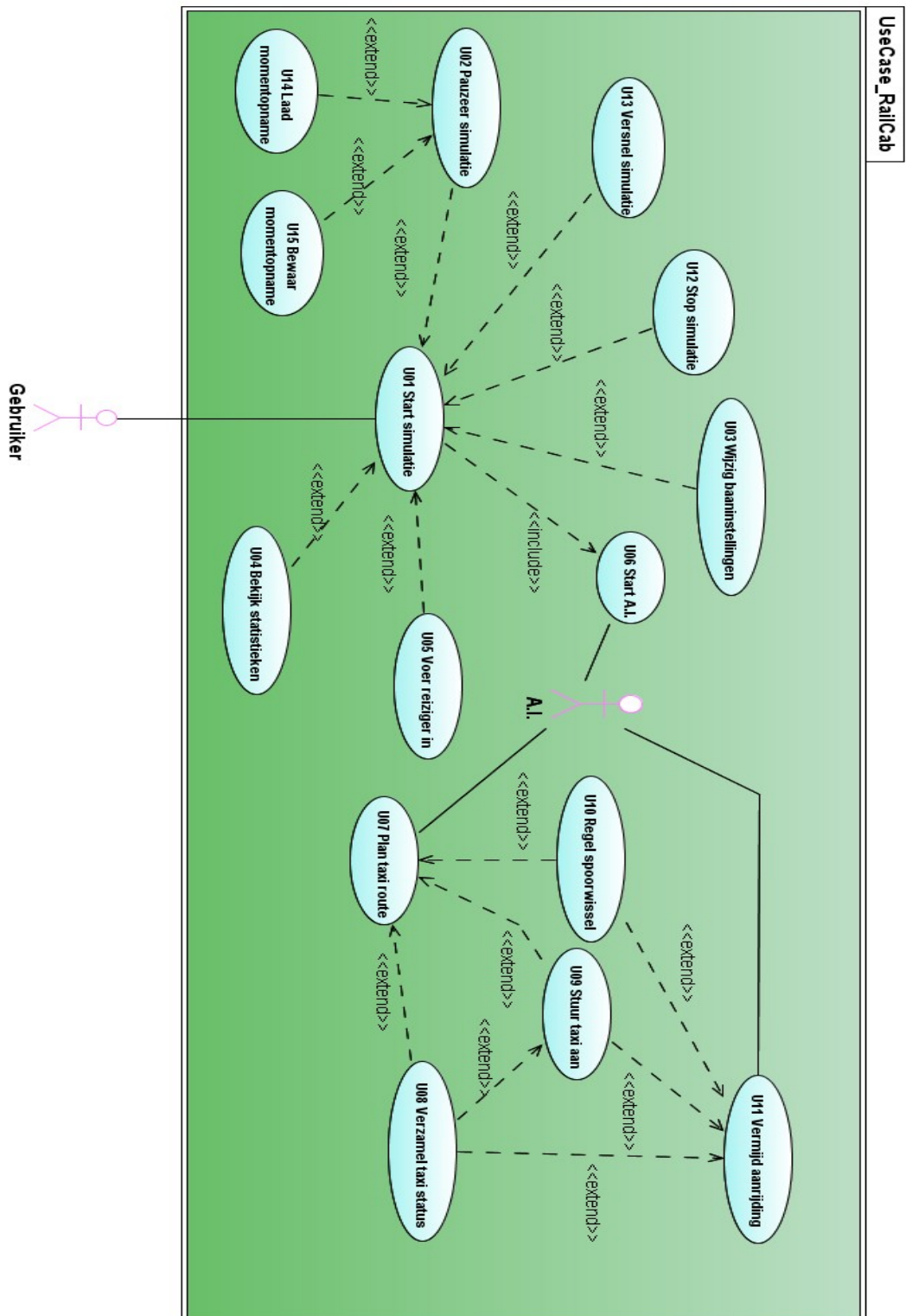
In dit project zal deze casus alleen in simulatie worden uitgewerkt. Er wordt gewerkt met een miniatuur versie. De baan zal in een cirkel lopen en op 8 plaatsen een 2e zijspoor hebben met een station daarnaast. Ook wordt er een softwarematige simulatie ontworpen waarin de miniatuur versie versneld kan worden afgespeeld en worden gepauzeerd.

## Actor List

Naam	Beschrijving
Gebruiker	Menselijke gebruiker die de simulatie kan bekijken en beïnvloeden.
A.I.	Intelligent subsysteem die de taxi's bestuurt.



## Use Case diagram:



## Requirements

Taxi's kunnen alleen vooruit rijden. In de praktijk zijn er aparte spoorwegen voor beide richtingen.

De simulatie werkt met 2 tot 8 taxi's.

De simulatie werkt met 2 tot 8 stations.

De volgende statistieken worden bijgehouden:

- Aantal wachtende passagiers per station.
- Wisselstanden
- Positie van taxi
- Aantal reizigers in station of taxi
- Aantal taxi's op baan en op station
- Aantal stations
- Wachtijd per reiziger en taxi.
- Reistijd per reiziger en taxi.
- Status per taxi.
- Gemiddelde bezetting van taxi
- Gemiddelde wachttijden en reistijden per reiziger en taxi.

Simulatie kan gepauzeerd worden, zodat meerdere reizigers makkelijk toegevoegd kunnen worden en statistieken bekeken worden.

Taxi's kunnen niet tegen elkaar botsen. Dit moet in de simulatie aangetoond kunnen worden door bijv. 2 taxi's op hetzelfde moment elkaar laten kruisen.

Taxi's kunnen niet met elkaar koppelen, omdat de voordelen niet opwegen tegen de toegevoegde complexiteit van het algoritme. De opdrachtgever moet wel akkoord gaan.

Afspraken over de interface:

- Een gebruiker kan een reiziger(s) toevoegen door een vertrekpunt, aantal reizigers en bestemming aan te geven.

- Taxi status weergeven (inclusief dreigende collision, waarbij de rechtdoorgaande taxi voorrang heeft) wordt op het scherm weergegeven.
- Instellingen veranderen zoals taxi wachttijd bij stations en taxi snelheid.

## Het algoritme

De complexiteit van de in software gevisualiseerde baan zal hetzelfde zijn als dat van de fysieke baan.

De in software gevisualiseerde baan kan meer stations en taxi's bevatten.

Gebruiker wordt gewaarschuwd bij onrealistische invoer van reizigers.

### **De volgende gegevens worden gebruikt voor de besturing van taxi's:**

Snelheid van taxi

Wachttijd van taxi wordt bij station bepaald aan de hand van de tijd die een reiziger nodig hebt om in een taxi te stappen.

Afstand van tussen stations

Lengte van station

Spoorwissel

Een lege taxi zonder ingeplande route blijft bij zijn eind punt tot dat er ruimte gemaakt moet worden dan rijdt hij door naar de volgende lege plek.

Rits mechanisme

### **De volgende factoren worden gebruikt voor de routeplanning van een taxi:**

Wachttijd van reiziger(s)

Aantal reizigers

Taxi bezetting

## Use case details

### Actor A.I.

Naam	<b>U06 Start A.I. (Included by: U01)</b>
Actor	A.I.
Preconditie	Simulatie is gestart.
Beschrijving	<ol style="list-style-type: none"> <li>1. Instellingen worden opgevraagd.</li> <li>2. A.I. wordt gestart als continu proces, aan de hand van de instellingen.</li> </ol>
Uitzondering	
Postconditie	A.I. is gestart.
Supplementaire Eisen	

Naam	<b>U07 Plan taxi route (Extends: U08, U09, U10)</b>
Actor	A.I.
Preconditie	A.I. is gestart. Reizigers beschikbaar
Beschrijving	<ol style="list-style-type: none"> <li>1. A.I. controleert de volgende factoren: Aantal beschikbare taxi's, aantal wachtende reizigers per station en de afstanden.</li> <li>2. A.I. rijdt taxi naar station om reizigers te vervoeren.</li> <li>3. A.I. rijdt naar station van bestemming.</li> <li>4. A.I. stopt de taxi als de bestemming is bereikt.</li> </ol>
Uitzondering	Geen reizen geen taxi's beschikbaar
Postconditie	Route voor één of meerdere taxi's bepaald.
Supplementaire Eisen	

Naam	<b>U08 Verzamel taxi status (Extended by: U07, U09, U11)</b>
Actor	A.I.
Preconditie	A.I. is gestart. Taxi(s) op baan
Beschrijving	<ol style="list-style-type: none"> <li>1. A.I. Controleert of taxi in beweging is of stilstaat.</li> <li>2. A.I. bepaalt positie van taxi aan de hand van sensoren en/of wiskundige berekeningen.</li> <li>3. A.I. controleert aantal reizigers in taxi.</li> <li>4. A.I. Controleert reistijd van taxi.</li> <li>5. A.I. geeft taxistatus door.</li> </ol>
Uitzondering	
Postconditie	Status taxi gecontroleerd.
Supplementaire Eisen	

Naam	<b>U09 Stuur taxi aan (Extends: U08. Extended by: U07, U11)</b>
Actor	A.I.
Preconditie	A.I. is gestart.
Beschrijving	<ol style="list-style-type: none"> <li>1. A.I. controleert taxi status.</li> <li>2. A.I. laat taxi vooruit rijden als dat gevraagd wordt.</li> <li>3. A.I. laat taxi stoppen als dat gevraagd wordt.</li> </ol>
Uitzondering	
Postconditie	Taxi is aangestuurd.
Supplementaire Eisen	

Naam	<b>U10 Regel spoorwissel (Extended by: U07, U11)</b>
Actor	A.I.
Preconditie	A.I. is gestart.
Beschrijving	<ol style="list-style-type: none"> <li>1. A.I. controleert spoorwissel.</li> <li>2. A.I. schakelt spoorwissel als dat gevraagd wordt.</li> </ol>
Uitzondering	Spoorwissel staat al op juiste positie.
Postconditie	Spoorwissel is naar de juiste positie geschakeld.
Supplementaire Eisen	

Naam	<b>U11 Vermijd aanrijding (Extends: U08, U09, U10)</b>
Actor	A.I.
Preconditie	A.I. is gestart.
Beschrijving	<ol style="list-style-type: none"> <li>1. A.I. controleert voorliggende taxi's binnen een bepaalde afstand.</li> <li>2. A.I. controleert overliggende taxi binnen een bepaalde radius.</li> <li>3. A.I. stopt taxi als het niet op de doorrijdende baan staat.</li> <li>4. Taxi hervat zijn voortgang als de andere taxi is gepasseerd.</li> </ol>
Uitzondering	
Postconditie	Aanrijding vermeden.
Supplementaire Eisen	Draait als een aparte proces en controleert continu.

### **Actor Gebruiker.**

Naam	<b>U01: Start simulatie (extends U02, U04, U03, U05 en includes U06)</b>
Preconditie	De simulatie is nog niet opgestart
Beschrijving	<ol style="list-style-type: none"> <li>1. Het gebruiker druk op het start knop om het systeem te starten</li> </ol>
Uitzondering	Het systeem kan geen verbinding maken met de fysieke baan.
Postconditie	De simulatie is gestart
Niet functionele eisen	

<b>Naam</b>	<b>U02: Pauzeer simulatie (is extended door U01 )</b>
<b>Preconditie</b>	De simulatie is gestart.
<b>Beschrijving</b>	1. De gebruiker drukt op pauze toets om het systeem te bevrozen.
<b>Uitzondering</b>	
<b>Postconditie</b>	Het systeem is gepauzeerd
<b>Niet functionele eisen</b>	

<b>Naam</b>	<b>U03: Wijzig baaninstellingen(is extended door U01)</b>
<b>Actor</b>	Gebruiker
<b>Preconditie</b>	De simulatie is gestart
<b>Beschrijving</b>	1. De gebruiker gaat naar het instellingen venster 2. De gebruiker wijzigt een of meerdere baan instellingen 3. De gebruiker kiest ok
<b>Uitzondering</b>	Baan instelling kan niet gewijzigd worden want het is in gebruik.
<b>Postconditie</b>	Baan instellingen zijn gewijzigd
<b>Niet functionele eisen</b>	

<b>Naam</b>	<b>U05: Voer reiziger in(is extended door U01)</b>
<b>Actor</b>	Gebruiker
<b>Preconditie</b>	De simulatie is opgestart.
<b>Beschrijving</b>	1. De gebruiker gaat naar de reiziger(s) invoer scherm 2. De gebruiker voert het aantal reiziger(s) in die met de taxi willen reizen. 3. De gebruiker selecteert het station waar de reiziger(s) opgepikt moeten worden. 4. De gebruiker selecteert het bestemming station van de reiziger(s). 5. Gebruiker kiest invoegen
<b>Uitzondering</b>	Er is een ongeldige waarde voor de reiziger(s) ingevoerd
<b>Postconditie</b>	De reiziger(s) zijn ingevoerd.
<b>Niet functionele eisen</b>	

<b>Naam</b>	<b>U12: Stop simulatie(is extended door U01)</b>
<b>Preconditie</b>	De simulatie is gestart
<b>Beschrijving</b>	1. De gebruiker drukt op het stop knop om de simulatie te stoppen.
<b>Uitzondering</b>	
<b>Postconditie</b>	De simulatie is gestopt
<b>Niet functionele eisen</b>	

<b>Naam</b>	<b>U04: Bekijk statistiek (is extended door U01)</b>
<b>Actor</b>	Gebruiker
<b>Preconditie</b>	De simulatie is gestart

<b>Beschrijving</b>	<ol style="list-style-type: none"> <li>De gebruiker kiest de soort statistiek die hij wil bekijken. Hij heeft de keuze uit reiziger, taxi en station. <ul style="list-style-type: none"> <li>Als de gebruiker voor reiziger(s) kiest dan kan hij de wachttijd van de reiziger(s) bekijken, hoeveel reizigers op een station wachten en waar een reiziger naartoe wilt of gaat.</li> <li>Als de gebruiker voor taxi kiest, dan kan hij zien waar een taxi bevindt en kan hij ook kijken of een taxi bezit of onbezet is. Hij kan ook de reistijd vanaf het vertrek naar de bestemming bekijken.</li> <li>Als de gebruiker voor station kiest dan kan hij hoeveelheid bezette of onbezette stations bekijken.</li> </ul> </li> </ol>
<b>Uitzondering</b>	
<b>Postconditie</b>	De gebruiker krijgt de statistiek te zien.
<b>Niet functionele eisen</b>	

<b>Naam</b>	<b>U04: Versnel simulatie (is extended door U01)</b>
<b>Actor</b>	Gebruiker
<b>Preconditie</b>	De simulatie is gestart
<b>Beschrijving</b>	1. Gebruiker druk op het versnellen knop om het systeem te versnellen
<b>Uitzondering</b>	De simulatie is al maximum versneld en kan alleen terug gezet worden op de standaard draaisnelheid.
<b>Postconditie</b>	De simulatie is versneld
<b>Niet functionele eisen</b>	

<b>Naam</b>	<b>U15: Bewaar momentopname (is extended door U02)</b>
<b>Actor</b>	Gebruiker
<b>Preconditie</b>	De simulatie is gestart en er zijn reizigers in het systeem ingevoerd.
<b>Beschrijving</b>	<ol style="list-style-type: none"> <li>Gebruiker gaat naar het moment opname menu.</li> <li>Gebruiker drukt op het bewaar momentopname knop.</li> </ol>
<b>Uitzondering</b>	
<b>Postconditie</b>	Momentopname is bewaard
<b>Niet functionele eisen</b>	

<b>Naam</b>	<b>U14: Laad momentopname (is extended door U02)</b>
<b>Actor</b>	Gebruiker
<b>Preconditie</b>	De simulatie is gestart
<b>Beschrijving</b>	<ol style="list-style-type: none"> <li>Gebruiker gaat naar het moment opname menu.</li> <li>Gebruiker kiest de optie, laad momentopname.</li> <li>Gebruiker selecteert welke moment opname hij wilt laden en drukt vervolgens op okay.</li> </ol>
<b>Uitzondering</b>	



<b>Postconditie</b>	Moment opname is geladen
<b>Niet functionele eisen</b>	

## Use Case List

### Actor A.I.

Naam	Beschrijving
<b>Start A.I.</b>	Intelligent subsysteem wordt gestart.
<b>Plan taxi route</b>	Beschikbare taxi's worden ingezet voor vervoer aan de hand van meerdere factoren.
<b>Verzamel taxi status</b>	Status (stop/rijd, positie, enz.) van taxi wordt gecontroleerd.
<b>Stuur taxi aan</b>	Taxi wordt aangestuurd. Een taxi kan rijden of stoppen.
<b>Regel spoorwissel</b>	Spoorwissel wordt geschakeld aan de hand van waar een taxi naartoe moet.
<b>Vermijd aanrijding</b>	Aanrijdingen tussen taxi's wordt vermeden door actieve controle en aansturing.

Naam	Beschrijving
<b>Start simulatie</b>	Het simulatie systeem wordt gestart
<b>Wijzig Baaninstellingen</b>	Alle baan instellingen kunnen hierin gewijzigd worden. Hierbij moet u aan de wissels en stations denken.
<b>Voer Reiziger in</b>	Reizigers die naar een bestemming willen kunnen hierin ingevoerd worden.
<b>Bekijk Statistieken</b>	Hier kunt u de statistieken van de reizigers, taxi's en stations bekijken.
<b>Pauzeer Simulatie</b>	Hier kunt u de simulatie op pauze zetten.
<b>Stop Simulatie</b>	Hier kunt u de simulatie stoppen
<b>Versnel Simulatie</b>	Hier kunt u de simulatie op een snellere snelheid laten draaien
<b>Bewaar momentopname</b>	Hier kunt u een momentopname maken van de simulatie, dit is enkel mogelijk als de simulatie op pauze staat.
<b>Laad momentopname</b>	Hier kunt u de momentopname van de simulatie laden, dit is enkel mogelijk als de simulatie op pauze staat.

## Record supplementary requirements

De simulatie dient met een druk op een knop te kunnen functioneren zodat het gebruikersvriendelijk wordt.

De simulatie wordt volledig in het Nederlands gerepresenteerd.

De simulatie kan met of zonder de fysieke baan functioneren.

De fysieke baan moet met maximaal 8 taxi's en 8 stations functioneren.

De simulatie is een desktop applicatie die op Unix, Windows en Mac kan werken.

Er kunnen maximaal 5 passagiers in een taxi

De simulatie heeft een gevisualiseerd beeld om de taxi's te bekijken.

## **Prioriteiten use case:**

### **Green:**

- U01: Start simulatie
- U05: Voer reiziger in
- U06: Start A.I.
- U11: Vermijd Aanrijding
- U07: Plan taxi route
- U10: Regel spoorwissel
- U08: Verzamel taxi status
- U09: Stuur taxi aan
- U04: Bekijk statistieken

### **Amber:**

- U02: Pauzeer simulatie
- U13: Versnel simulatie

### **Red:**

- U03: Wijzig baaninstelling
- U12: Stop Simulatie
- U14: Bewaar momentopname
- U15: Laad momentopname

## User interface sketches:

### *Instellingen venster*

**Baaninstellingen**

Algemeen Taxi

Snelheid taxi

Wachttijd taxi op station

OK Annuleren

### *Simulatie railcab UI*

**Simulatie RailCab**

Bestand Opties Help

Momentopname opslaan  
Momentopname laden  
Afsluiten Alt + F4

**Legenda**

- Station
- Taxi
- Spoorwissel
- 000 Reiziger(s)

**Reiziger(s) invoegen**

Aantal reizigers Vertrekpunt Bestemming Invoegen

Simulatie starten Simulatie versnellen Simulatie pauzeren Simulatie stoppen

### *Statistieken venster*



The screenshot shows a window titled "Statistieken Station" with a blue header bar containing standard window controls. The main content area displays the following information:

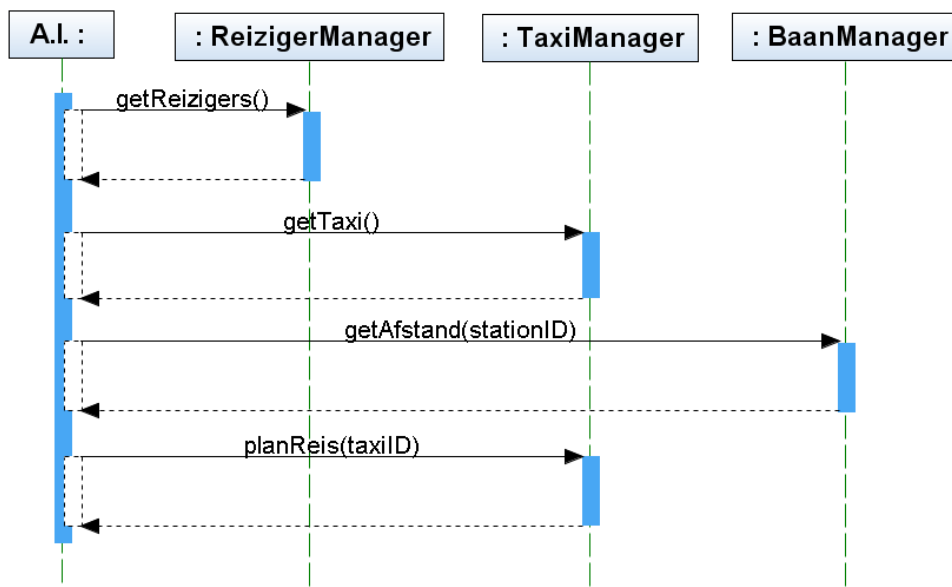
- Naam station:** Station #1
- Aantal reizigers in station:** 3
- Aantal taxi's op station:** 1
- Kies reiziger:** A text input field followed by a dropdown arrow button.
- OK:** A button in the bottom right corner.

### **Class diagram:**

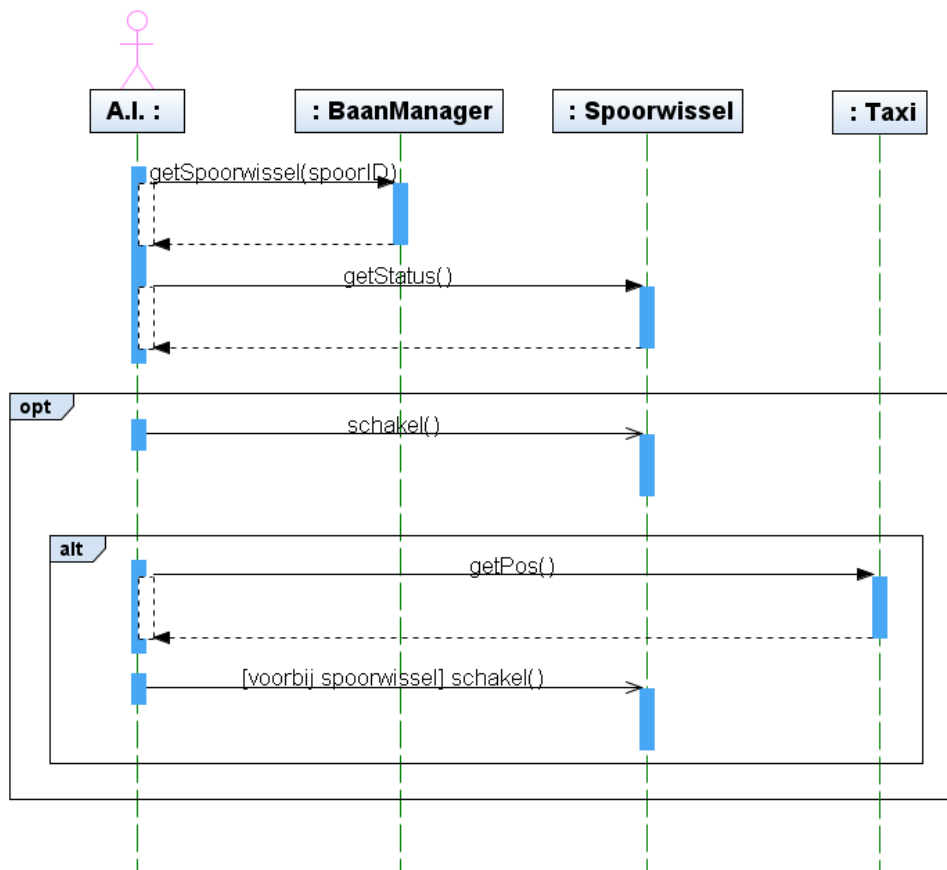
(zie bijlage 1)

## Sequence diagram:

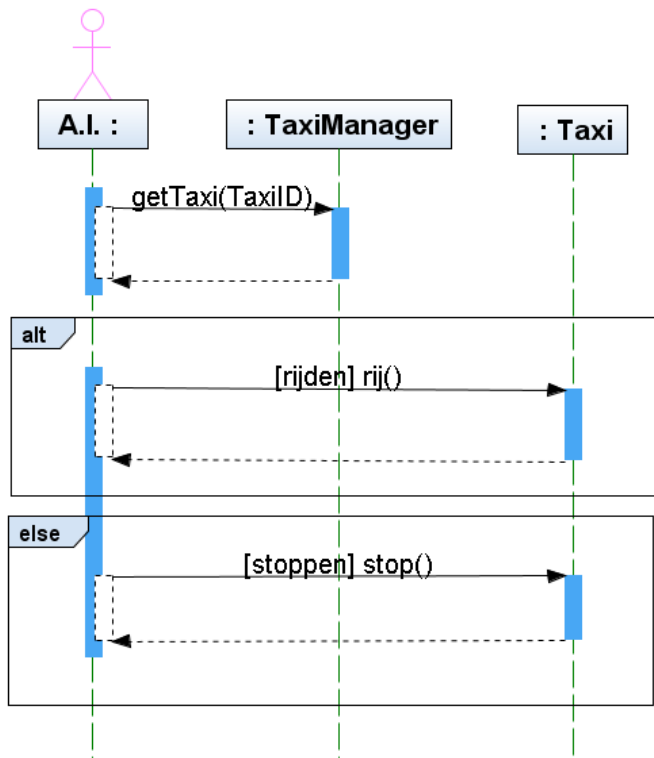
*Plan taxi route*



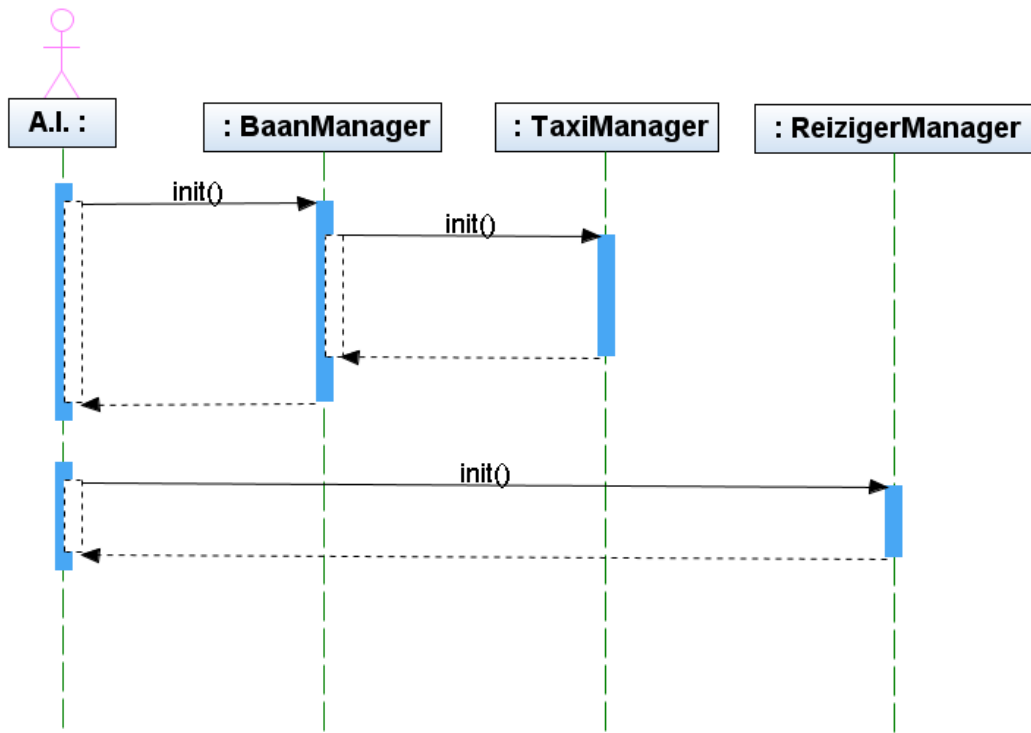
*Regel spoorwissel*



*Stuur taxi aan*

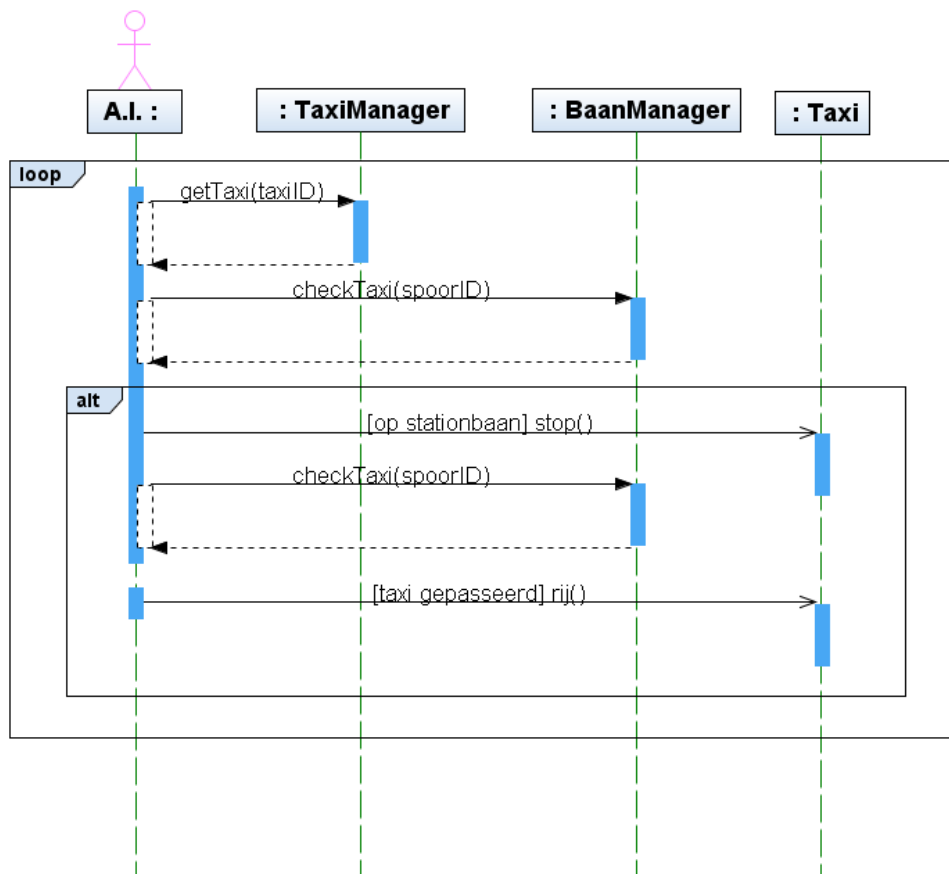


*Start A.I.*

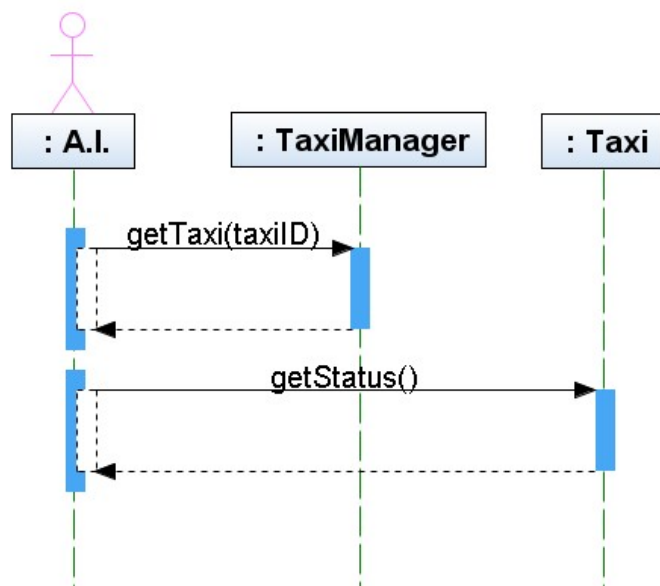


*Vermijd aanrijding*

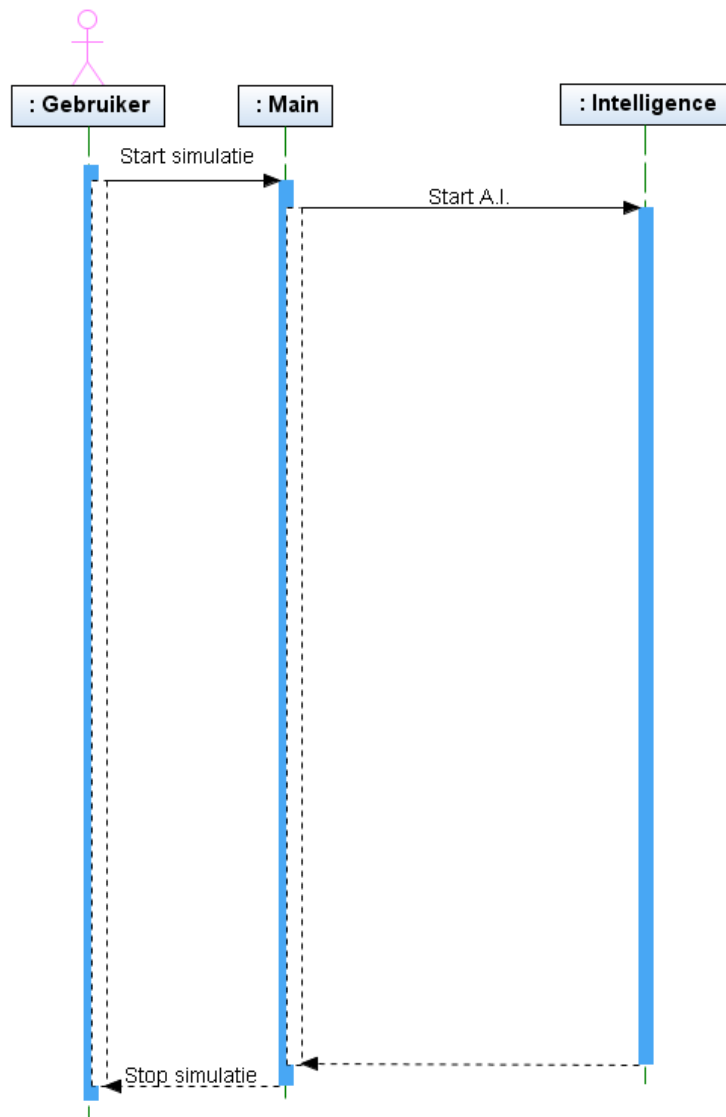




*A.I. verzamelt taxi status*

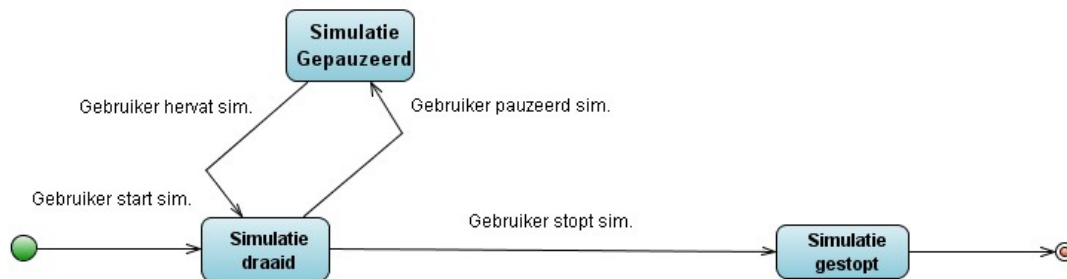


*Gebruiker*

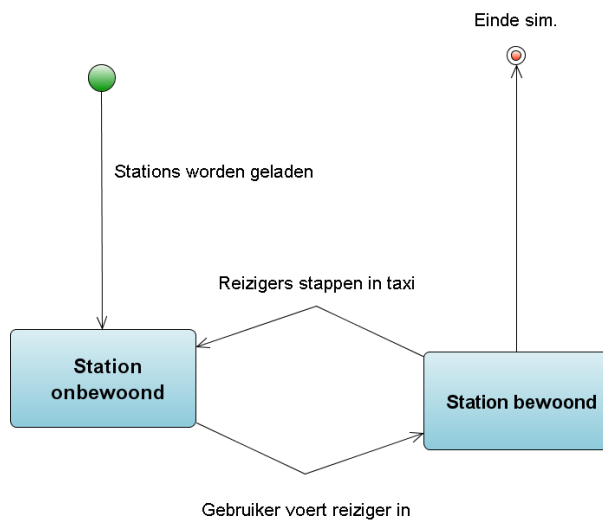


## State diagram:

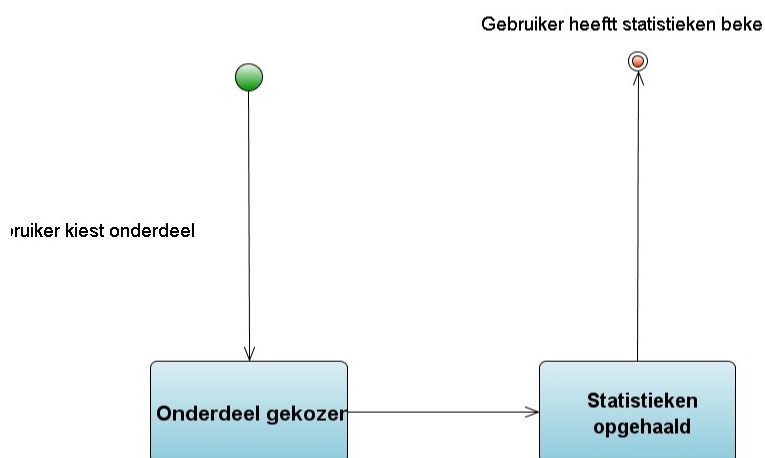
### *Gebruiker*



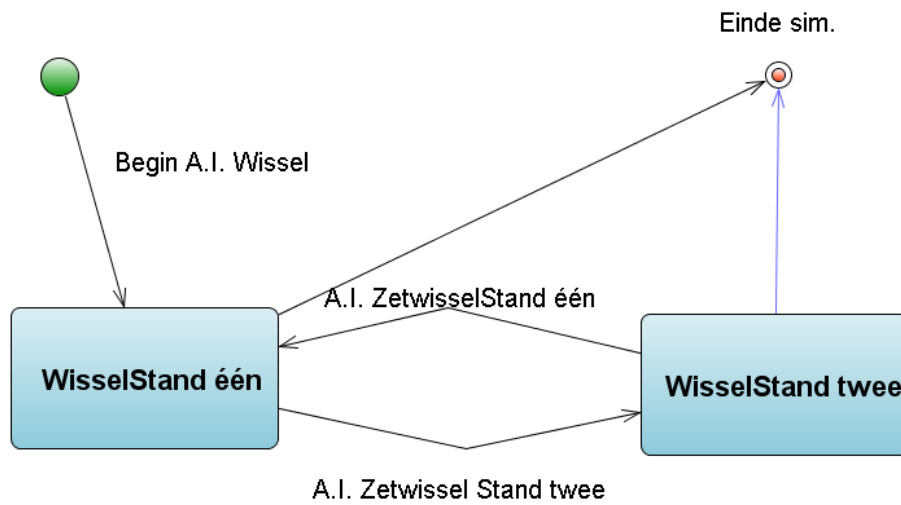
### *Station*



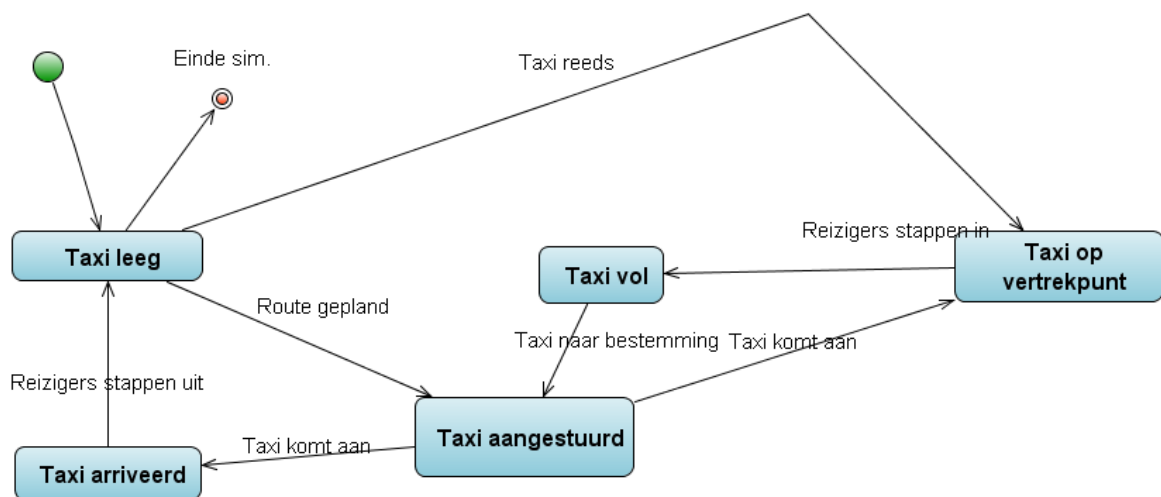
### *Statistieken*



## Wissel



## Taxi



## Reiziger

