



Radboud Universiteit Nijmegen

Omgevingsbewuste systemen & Risico's

Master Thesis

Informatiekunde

Freek van Workum

v 1.0

Auteur	Freek van Workum
Studentnr	0652296
Afstudeernr	163IK
Begeleider	Dr. P. Van Bommel
Universiteit	Radboud Universiteit Nijmegen

Colofon

Auteur: Freek van Workum
Contact: freek@fvanw.nl
Titel: Omgevingsbewuste systemen en risico's
Versie 1 (definitief)
Afstudeernr: 163IK
Datum: 27-07-2012
Plaats: Nijmegen

Opleiding: Informatiekunde (NII)
Instituut: Faculteit der Natuurwetenschappen, Wiskunde & Informatica (FNWI)
Universiteit: Radboud Universiteit Nijmegen
Plaats: Nijmegen

Begeleider: Dr. P. van Bommel
Contact: p.vanbommel@cs.ru.nl

Voorwoord

Het schrijven van de master thesis is voor mij een leerzaam leertraject geweest. Het heeft bij mij het besef gebracht dat het belangrijk is om gedegen structuur aan te brengen alvorens aan te vangen met het onderzoek. Na overleg met Patrick van Bommel heb ik besloten om van het oorspronkelijke thesis onderwerp af te wijken en de keuze te maken voor omgevingsbewuste systemen.

Gedurende enige tijd ben ik naast mijn werkzaamheden op mijn werk bezig geweest om te werken aan dit onderzoek wat ik uiteindelijk tot een goed einde heb weten te brengen. Aan de start van het onderzoek had ik de nodige belangstelling voor de werking van omgevingsbewuste systemen en de mogelijke aandachtspunten bij dergelijke systemen. Gedurende het onderzoek heb ik antwoorden proberen te vinden op de door mij gestelde vragen en is de complexiteit van dergelijke systemen bij mij duidelijk geworden.

De afgelopen periode ben ik nauwlettend begeleid door Patrick van Bommel. Ik kon met mijn vragen bij hem terecht. Vaak stelde hij een aantal vervolgvragen waarmee ik dan weer verder kon. Ook wist hij mij de motiveren tijdens de lastige fases tijdens mijn thesis periode. Hiervoor wil ik hem dan ook hartelijk danken.

Ook mijn oud studie genoten en vrienden Willem Klomp, Roland Swinkels en Chiel Schutter wil ik bedanken voor het inhoudelijk sparren over menig onderwerp en de geboden ondersteuning op inhoudelijk vlak.

Ten slotte bedank ik mijn vriendin Sofie en mijn ouders voor de geboden support gedurende mijn scriptie tijd.

Samenvatting

Omgevingsbewuste systemen krijgen een steeds grote rol in onze moderne maatschappij. Gebruikers die bewust of onbewust in aanraking komen met een omgevingsbewust systeem hebben geen idee van de werking van een dergelijk systeem en de risico's die hierbij komen kijken.

De onderzoeksvraag in deze thesis is als volgt:

Wat is een omgevingsbewust systeem, welke risico's brengt een dergelijk systeem met zich mee en hoe zijn deze risico's te beperken?

Een omgevingsbewust systeem heeft een uitgebreider eisenpakket dan een systeem dat niet omgevingsbewust is. Zoals de naam aangeeft is een omgevingsbewust systeem zich bewust van de omgeving en dient hierop ontwikkeld en ingericht te zijn. Dit vergt een aantal extra systeemeisen of eisen die uitgebreider zijn dan bij een vergelijkbaar systeem dat niet omgevingsbewust is. Op basis van literatuurstudie is er in deze thesis onderzoek gedaan naar de betreffende eigenschappen.

Een omgevingsbewust systeem brengt meer en grotere risico's met zich mee vanwege de eigen beslissingen die een systeem maakt op basis van de verkregen informatie uit de omgeving. Hierbij verwacht de gebruiker of de omringende systemen binnen afzienbare tijd een reactie of zelfs een proactieve reactie. Hierbij speelt het tijdsaspect een grote rol. Een omgevingsbewust dient dan ook een weloverwogen beslissing te maken in een zo kort mogelijke tijd. Dit kan leiden tot verkeerde beslissingen met alle gevolgen van dien.

In de thesis wordt UML beschreven en de hiervoor reeds bestaande uitbreiding op UML, CAMEL. CAMEL levert twee metamodellen die ervoor zorgen dat een systeem zo gemodelleerd kan worden om de omgeving waar te nemen en vervolgens ook te kunnen reageren.

Op basis van de uit de literatuur afkomstige eigenschappen van omgevingsbewuste systemen, de risico's van omgevingsbewuste systemen en de CAMEL taal wordt aan de hand van twee praktijkvoorbeelden bekeken wat mogelijke pijnpunten van omgevingsbewuste systemen zijn. Dit op het gebied van moderne omgevingsbewuste gezondheidszorg en het omgevingsbewuste systeem in een moderne auto.

De eigenschappen worden vergeleken met de risico's en het onderlinge verband waar deze in kunnen voorkomen.

Uit de thesis is naar voren gekomen dat er aanzienlijke risico's kleven aan omgevingsbewuste systemen. Alhoewel CAMEL een handvat aanreikt om de omgevingsbewuste systemen te modelleren valt er over de uitvoering en implementatie nog veel te zeggen. Er zijn een aantal aandachtspunten waarop gelet zou kunnen worden om zo de kans dat de risico's tot uitvoer komen, te verkleinen. Het goed nadenken over het herkennen van gedrag, rekening houden met tijd en reactie staan hierbij centraal. Dit alles wordt overzichtelijk weergegeven in een meta-model omgevingsbewuste systemen en risico's.

Inhoudsopgave

Samenvatting.....	4
Inhoudsopgave.....	5
Hoofdstuk 1 – Inleiding.....	7
1.1 Probleemstelling.....	9
1.1.1 Hoofdvraag	9
1.1.3 Deelvragen.....	9
1.2 Verantwoording.....	10
1.2.1 Relevantie.....	10
1.3 Werkwijze.....	10
1.4.1 Afbakening.....	11
1.4.2 Variabelen	11
Hoofdstuk 2 - Omgevingsbewuste systemen.....	12
2.1 Wat is omgevingsbewust?.....	12
2.2 Wat is een omgevingsbewust systeem?.....	13
2.3 Eigenschappen van omgevingsbewuste systemen.....	14
2.3.1 Eigenschappen verwerkt in matrix.....	16
Hoofdstuk 3 – Risico's	18
3.1 Wat is een risico?.....	18
3.2 Risico Analyse.....	19
3.2.1 Identificatie.....	19
3.2.2 Evaluatie.....	19
3.2.3 Respons.....	19
3.3 Risico's van omgevingsbewuste systemen.....	20
3.3.1 Upgraden van woningen.....	20
3.3.2 Slecht voorbereide informatie uitwisseling.....	20
3.3.3 Systeemonderhoud.....	21
3.3.4 Ontwerp voor thuisgebruik.....	21
3.3.5 Sociale implicaties.....	22
3.3.6 Betrouwbaarheid.....	22
3.3.7 Systeemintelligentie.....	24
Hoofdstuk 4 – UML & Camel.....	25
4.1 UML.....	25
4.1.1 Wat is UML?.....	25
4.2 Wat zijn de componenten van UML?.....	26
4.2.1 Diagrammen.....	28
Usecase diagram.....	28
Klassendiagram.....	30
Objectdiagram.....	31
Sequencediagram.....	32
Communicatiediagram.....	33
Toestandsdiagram.....	34
Activiteitsdiagram.....	35
4.3 Context Awareness Modelling Language (CAMEL).....	36
4.3.1 Het doel van CAMEL.....	36
4.3.2 De Taal CAMEL.....	37
4.3.3 Monitoren van de omgeving.....	37
4.3.4 Reageren op verandering.....	39

4.3.5 Toepassen van CAMEL.....	40
Hoofdstuk 5 – Casus.....	41
5.1 Inleiding.....	41
5.2 Omgevingsbewuste systemen en risico's.....	42
5.2.1 Methode.....	44
5.3 Gezondheidszorg.....	45
5.3.1 Usecases.....	46
5.3.2 Klassendiagram / Objectdiagram.....	50
5.3.3 Sequencediagram.....	51
5.3.4 Omgevingsbewuste klassendiagrammen.....	52
5.3.4.1 Monitoren van de omgeving.....	52
5.3.4.2 Reageren op de omgeving.....	53
5.3.5 Risico's versus eigenschappen.....	54
5.4 Auto industrie.....	57
5.3.1 Usecases.....	58
5.4.2 Klassendiagram / Objectdiagram.....	61
5.4.3 Sequencediagram.....	64
5.4.4 Omgevingsbewuste klassendiagrammen.....	67
5.4.4.1 Monitoren van de omgeving.....	67
5.4.4.2 Reageren op de omgeving.....	68
5.4.5 Risico's versus eigenschappen.....	69
Hoofdstuk 6 – Conclusie.....	72
6.1 Vervolgonderzoek.....	75
Literatuur.....	76

Hoofdstuk 1 – Inleiding

In deze thesis zullen omgevingsbewuste systemen aan bod komen. Deze zijn sterk in opkomst en zorgen voor een personalisering van systemen, processen en gebruikersomgevingen op basis van reactie en interactie tussen systemen en omgeving. Doordat er steeds hogere eisen aan systemen worden gesteld waarbij ook de interactie met gebruiker(s) en omgeving steeds verder centraal komt te staan, is het interessant om je af te vragen hoe dergelijke systemen in elkaar zitten en op welke wijze ze verschillen van “traditionele systemen”. Dit vraagstuk komt naar voren op basis van een probleemstelling / vraag met deelvragen. Ook komt aan bod wat de risico's zijn van dergelijke omgevingsbewuste systemen.

Omgevingsbewuste systemen, in het engels ook wel pervasive / ubiquitous systems genaamd, worden doorgaans onder de noemer ambient computing geschaard in de engelse taal. Vaak worden deze termen ook als synoniemen gebruikt. Letterlijk vertaald naar het Nederlands betekenen deze termen het volgende:

pervasive: “altijd en overal”

ambient: “omgevings”

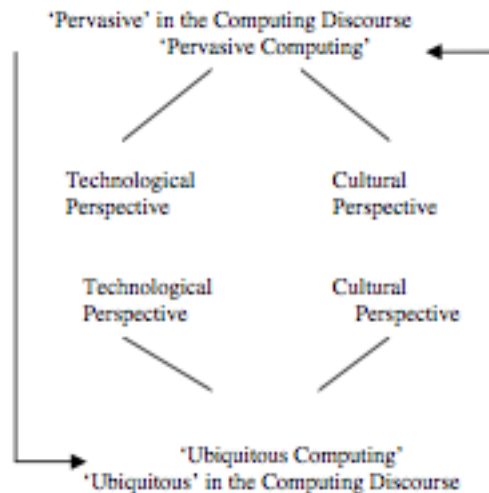
ubiquitous: “alomvertegenwoordigd”

Omgevingsbewuste systemen worden dan ook door bovenstaande termen benoemd. De systemen zijn altijd en overal beschikbaar(alomvertegenwoordigd) en houden rekening met de omgeving en veranderingen in de omgeving. Deze systemen zijn zoveel mogelijk benaderbaar en beschikbaar voor een groot aantal gebruikers en zijn ze zich bewust van de altijd veranderende omgeving en passen zich indien nodig hierop aan.

Door het vele gebruik van de term pervasive in verschillende vakgebieden is het een sterk context afhankelijk begrip. Er zijn door de jaren heen verschillende definities ontstaan van de term pervasive. De standaard vertaling betekent “verspreid”, “verdeeld” of “ergens doorheen gaand”.

De term wordt veel gebruikt bij bijvoorbeeld pervasive games. Hierin zijn twee takken in te onderscheiden. De technologische kant, waarbij de techniek als middel wordt gebruikt en de culturele kant waarbij de aandacht ligt op het spel zelf en de verhouding met de echte wereld.

In zowel de gaming tak alsmede in de computing tak wordt er veelvuldig gebruikt gemaakt van de term pervasive. De technologische tak verklaart hoe het resultaat bereikt kan worden en de culturele kant waarom dit dient te gebeuren. Ook in andere vakgebieden zoals de gezondheidszorg of de auto industrie wordt in meerdere mate gebruikt van omgevingsbewuste technieken. Dit zal nader worden uitgewerkt in de casus.



Figuur 1

De term pervasive is in 1998 geïntroduceerd door IBM. Hiervoor hadden onderzoekers bij Xerox PARC al technologie ontwikkeld die zij “ubiquitous computing” noemden. Hieruit is “ubiquitous gaming” voortgekomen wat uiteindelijk “pervasive gaming” is geworden. Het onderzoek bij Xerox is in 1988 gestart uit frustratie over tekortkomingen van de PC. Deze was te ingewikkeld en te lastig in gebruik. Ook vond men de PC te geïsoleerd van mensen en overige activiteiten. Hieruit is het idee ontstaan voor de PC in de 21e eeuw waarbij de gedachte van de mens centraal stond in de maatschappij met op de achtergrond de PC.

In 1998 is IBM aan de slag gegaan met pervasive computing uit het oogpunt van de toekomst waarin iedereen snel en makkelijk gebruik kan maken van bepaalde diensten. Dit met name in de e-business destijds.

Mark Bergman, hoofd pervasive computing van IBM destijds:

“Pervasive computing is about enabling people to gain immediate access to information and services anywhere, anytime, without having to scrounge for a phone jack. However, while mobility and wireless technology are a big part of it, it's really about making e-business personal. Thanks to the explosive growth of the Internet, people will soon expect to be able to engage in electronic business effortlessly”

Er is samenhang tussen termen als pervasive, ubiquitous en termen als mobile computing, ambient computing en environmental of embedded computing. Dit zijn namelijk termen die gebruikt worden als synoniemen, allen met de betrekking tot omgevingsbewustheid.

1.1 Probleemstelling

In deze thesis komt aan bod wat een omgevingsbewuste systeem is en welke (aanvullende) eigenschappen een omgevingsbewust systeem kenmerkt. Ook wordt gekeken naar de mogelijke additionele risico's van deze systemen. Er worden voorbeelden besproken van omgevingsbewuste systemen en de mogelijke risico's die omgevingsbewuste systemen met zich mee brengen. Voor het ontwikkelen van omgevingsbewuste systemen zullen een aantal eigenschappen aan bod komen die mogelijk afwijken van de eigenschappen van traditionele systemen, deze eigenschappen zullen worden benoemd om zo tot een heldere definitie van een omgevingsbewust systeem te komen.

Het is belangrijk dat er een goed beeld ontstaat van de kenmerken en benodigde eigenschappen van een omgevingsbewust systeem en het eventueel aanvullende aandachtspunten tijdens de ontwikkeling en operationele fase van een dergelijk systeem. Een omgevingsbewust systeem neemt namelijk zelfstandig beslissingen of voorziet de systemen die deze beslissingen nemen van informatie uit de omgeving. Deze informatie stroom is van groot belang, het weglaten van waardevolle informatie of filteren van de verkeerde informatie kan resulteren in een keuze die achteraf niet gewenst blijkt te zijn. Nou is dit bij het wel of niet activeren van sfeerverlichting in een woonkamer misschien niet van belang, in de gezondheidszorg of auto industrie kan een verkeerd besluit een veel grotere impact hebben. Deze mogelijke risico's en de impact hiervan komen in deze thesis aan bod. Dit resulteert in de volgende hoofdvraag:

1.1.1 Hoofdvraag

Wat is een omgevingsbewust systeem, welke risico's brengt een dergelijk systeem met zich mee en hoe zijn deze risico's te beperken?

Om achter de antwoorden op deze vraag te kunnen komen zal een aantal deelvragen beantwoord dienen te worden die duidelijkheid zullen verschaffen bij het beantwoorden van de gestelde hoofdvraag.

1.1.3 Deelvragen

Overzicht van de deelvragen:

- Wat is een omgevingsbewust systeem?
- Wat zijn de eigenschappen van een omgevingsbewuste systeem?
- Wat zijn de componenten van UML?
- Wat is CAMEL en wat hoe werken de metamodelen van CAMEL?
- Wat is een risico?
- Wat zijn risico's van omgevingsbewust systemen?
- Wat is het verband tussen risico's en eigenschappen van omgevingsbewuste systemen ?

Door antwoorden te verkrijgen op bovenstaande vragen kan er antwoord gegeven worden op de hoofdvraag.

1.2 Verantwoording

Er worden steeds hogere eisen gesteld aan systemen en met name interactie met deze systemen. Alles moet interactiever. Men wilt intelligente systemen die op basis van eerder gebruik bepalen wat gebruiker X of gebruiker Y aan functionaliteit tot zijn beschikking krijgt of welke afweging een systeem maakt.

Dit geheel valt onder de noemer omgevingsbewustheid. Omgevingsbewuste kunnen voordelen met zich meebrengen mits deze goed zijn ingericht. Er kleven echter ook bepaalde risico's aan zulke intelligente systemen. Deze risico's kunnen verstrekkende gevolgen hebben.

1.2.1 Relevantie

Het is voor bedrijven van belang om te weten wat omgevingsbewust inhoudt en waarmee rekening gehouden moet worden. De risico's van omgevingsbewuste systemen moeten in kaart worden gebracht en tijdens de ontwikkeling moet gelet worden op het beperken van deze risico's voor een optimaal resultaat. Hier is bij traditionele systemen genoeg over bekend. Bij omgevingsbewuste systemen is dit echter niet het geval. Derhalve zal dit in deze thesis onderzocht worden.

1.3 Werkwijze

In de thesis zal op basis van literatuurstudie, kennis worden vergaard over omgevingsbewuste systemen en de benodigde functionaliteit van dergelijke systemen. De risico's van omgevingsbewuste systemen worden in kaart gebracht en het verband tussen eigenschappen en risico's zal worden bekeken met behulp van UML/CAMEL. Dit alles wordt nader uitgewerkt in de twee casussen.

1.4 Theoretisch kader

De volgende kennisgebieden zijn van toepassing op deze thesis:

Informatie technologie:

- Omgevingsbewuste systemen
- Modelleren (UML + CAMEL)
- Modelleren van systemen en bijbehorende risico's

Risico's:

- Definitie van risico's in het algemeen
- Risico's van omgevingsbewuste systemen

1.4.1 Afbakening

In deze thesis zal gekeken worden naar omgevingsbewust systemen en de risico's hiervan. Dit zijn aan de ene kant systemen die ten allen tijde beschikbaar zijn, denk aan mobiele applicaties, aan de andere kant zijn het systemen die signalen oppikken uit de omgeving en op basis hiervan handelen. De eerstgenoemde definitie zal worden geabstraheerd in deze thesis.

De mobiele applicaties laat ik buiten beschouwing daar de keuze is gemaakt voor het inzoomen op de andere “vorm” van omgevingsbewuste systemen. De focus zal dus liggen op systemen die interactief en proactief handelen op basis op signalen die bijvoorbeeld door sensoren worden verkregen.

Op het gebied van risico's zal gekeken worden naar de risico's tijdens de verschillende stadia van een systeem. Met stadia wordt bedoeld het traject tussen ontwikkel fase tot en met operationele fase en wat een fout of onvolkomenheid tijdens het modelleren voor gevolg kan hebben. In de casus wordt getoetst in welke mate een bepaalde keuze tijdens het modelleren traject gevolgen kan hebben voor een risicovolle uitwerking hiervan in een later stadium.

1.4.2 Variabelen

De variabelen die aan bod komen in het onderzoek zijn als volgt:

Onafhankelijke variabelen:

- Modelleren van een omgevingsbewust systeem (CAMEL)
- Eigenschappen van een omgevingsbewust systeem
- Risico's

Afhankelijke variabelen:

- Risico's van een omgevingsbewust systeem

Weergegeven in een model ziet dit er als volgt uit:



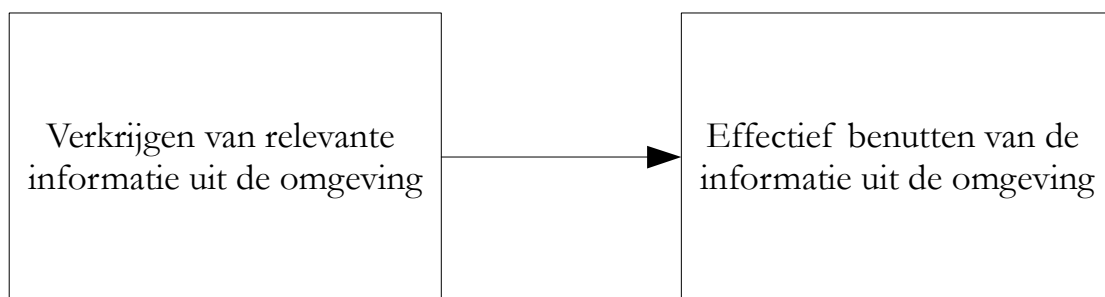
Figuur 2

Hoofdstuk 2 - Omgevingsbewuste systemen

2.1 Wat is omgevingsbewust?

Volgens de encyclopedie kent het woord omgevingsbewust de volgende betekenis: *“Het goed geïnformeerd zijn over de relevante ontwikkelingen in de omgeving van de organisatie en benut deze kennis effectief.”*

Het verkrijgen van informatie staat hierin centraal. De informatie uit de omgeving in het bijzonder. Vervolgens dient deze informatie dan ook effectief benut te worden. Deze twee zaken bevatten dan ook de kern van het omgevingsbewuste. Dit is weer te geven in onderstaand model (Figuur 3).



Figuur 3

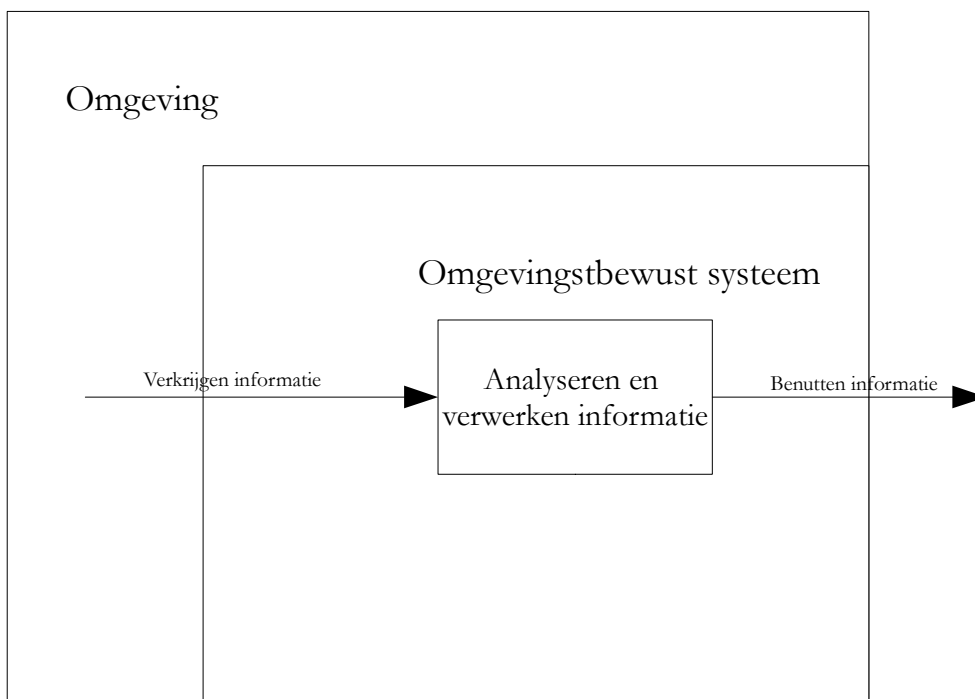
Omgevingsbewustheid is dan ook de het besef van de aanwezigheid van de omgeving.

2.2 Wat is een omgevingsbewust systeem?

Een omgevingsbewust systeem is een systeem dat informatie verkrijgt uit de omgeving en deze informatie analyseert om vervolgens op basis van de nuttige informatie een afweging te maken en/of door te sturen naar een achterliggend systeem.

Daar een dergelijk systeem in korte tijd erg veel input aan informatie krijgt, is het van belang dat er een goede schifting wordt gemaakt tussen de nuttige en minder nuttige informatie. Dit vergt aardig wat rekenkracht en programmeer werk vooraf. Ook is het zaak dat deze analyse niet ten koste gaat van de reactietijd van een omgevingsbewust systeem.

Een omgevingsbewust systeem dient in korte tijd de juiste afweging te maken en snel en adequaat te schakelen met diverse gerelateerde systemen die in verband staan met dit omgevingsbewust systeem.



Figuur 4

Toelichting: In bovenstaand model (figuur 4) is een omgevingsbewust systeem weergegeven met daarbij de omgeving. Het omgevingsbewust systeem verkrijgt informatie uit de omgeving om deze informatie vervolgens te analyseren en verwerken. De waardevolle of nuttige informatie wordt vervolgens doorgegeven om zo benut te kunnen worden door achterliggende systemen.

2.3 Eigenschappen van omgevingsbewuste systemen

Aan de hand van een literatuurstudie van een aantal artikelen over omgevingsbewuste systemen is een selectie gemaakt uit de meest voorkomende en genoemde eigenschappen van omgevingsbewuste systemen. In deze paragraaf worden de meest voorkomende eigenschappen van omgevingsbewuste systemen beschreven. Vervolgens zal in een matrix worden weergegeven hoe vaak en in welke artikelen deze eisen terug te vinden zijn.

Alomtegenwoordig (Pervasivity):

Voorzien van diensten aan de gebruiker, los van locatie en of apparaat.

De term pervasivity betekent alomvattendheid/alomtegenwoordig. Dat wil zeggen dat de dienst of applicatie ten alle tijde beschikbaar dient te zijn voor de gebruiker. Los van locatie of van eventueel platform waar de dienst op wordt benaderd.

Omgevingsbewustheid (Context-awareness):

Bewustheid van het systeem met betrekking op de omgeving.

Het systeem pikt signalen op uit de omgeving. Bijvoorbeeld middels sensoren detecteert het veranderingen of constateringingen uit de omgeving, waarbij het essentieel is dat de bruikbare signalen gefilterd worden uit de niet of minder bruikbare signalen.

Bovenstaande twee eigenschappen zijn specifiek kenmerkend voor een omgevingsbewust systeem. Naast de omgevingsbewustheid van een omgevingsbewust systeem is er uiteraard ook sprake van eigenschappen die voor kunnen komen in een niet omgevingsbewust systeem. Een omgevingsbewust systeem is immers een traditioneel systeem met extra functionaliteit en derhalve ook bijbehorende eigenschappen. De mate waarin een bepaalde eigenschap deel uitmaakt van een omgevingsbewust systeem kan echter afwijken van de mate waarin deze zelfde eigenschap op zou treden in een traditioneel systeem. De aanname dat de in dit hoofdstuk genoemde eigenschappen enkel en alleen voor kunnen komen in een omgevingsbewust systeem is dan ook niet waar.

Bruikbaarheid (Usability):

Mate waarin het systeem bruikbaar is, ervaren door de gebruiker.

De bruikbaarheid van een bepaald systeem, dienst of applicatie en de mate waarin dit door de gebruiker ervaren wordt.

Zelfonderhoudbaarheid(Autonomicity):

De mate waarin een systeem zelf onderhoudend of opererend is en zelfstandig gerund kan worden zonder te veel inbreng of sturing van derden.

Proactiviteit (Proactivity):

Het vooruitdenken of anticiperen van het systeem op de wensen van de gebruiker. Om snel te kunnen handelen en reageren is het voor een systeem van groot belang dat er snel geschakeld kan worden. Dit kan in grote mate bereikt worden door proactief te handelen.

Betrouwbaarheid (Dependability):

De betrouwbaarheid van het systeem houdt in hoe betrouwbaar een systeem is en in welke mate deze continuïteit kan waarborgen aan gebruikers.

Aanpasbaarheid (Adaptability):

De mate waarin een systeem zich kan aanpassen aan gebruikers, omgeving of andere systemen.

Prestatie (Performance):

De mate van presteren van een bepaald systeem. Hierbij kunnen een aantal maatstaven gehanteerd worden. Snelheid en reactietijd zijn twee van deze belangrijke maatstaven.

Beschikbaarheid (Availability):

De mate van beschikbaarheid van een bepaald systeem. Hoe meer of hoe langer een systeem beschikbaar is hoe beter. Men streeft hierbij naar 100%, in de praktijk zal dit echter lager uitvallen daar de maximale beschikbaarheid vaak niet realiseerbaar is.

Veiligheid (Security):

De mate van veiligheid van een systeem. Veiligheid, zowel authenticatie alsmede autorisatie zijn bij omgevingsbewuste systemen van groot belang. Een omgevingsbewust systeem neemt zelf bepaalde beslissingen die mogelijk een risico kunnen vormen op het gebied van veiligheid.

Instelbaarheid (Adjustability):

De mate waarin een systeem ingesteld of geconfigureerd kan worden.

2.3.1 Eigenschappen verwerkt in matrix

In de bestudeerde literatuur waren een aantal eigenschappen van omgevingsbewuste systemen veelvuldig aanwezig. Deze eigenschappen werden in de artikelen in meer of in mindere mate toegelicht en beschreven. Onderstaande matrix (figuur 5) toont een tiental artikelen en een elftal veelvuldig teruggekomen eigenschappen in deze artikelen. Per eigenschap is bekeken in welk artikel deze voorkwam. Het totaal is opgeteld en weergegeven in de onderste rij.

	Aanpasbaarheid	Alomverteenwoordigheid	Omgevingsbewustheid	Bruikbaarheid	Zelfonderhoudbaarheid	Proactiviteit	Betrouwbaarheid	Veiligheid	Beschikbaarheid	Prestatie	Instelbaarheid
Application Requirements for Middleware for Mobile and Pervasive Systems	x	x	x					x	x	x	
Futurephone Application Requirements for Middleware for Mobile and Pervasive Systems	x	x	x					x	x	x	
A Middleware for Context-Aware Agents in Ubiquitous Computing Environments				x		x		x	x	x	x
A Framework for Engineering Pervasive Applications Applied to Intra-vehicular Sensor Network Applications				x	x	x	x	x	x	x	
Requirements and Challenges for Building Service-Oriented Pervasive Middleware	x	x	x		x	x		x	x	x	
Survey of Requirements and Solutions for Ubiquitous Software	x	x	x					x	x	x	x
Knowledge Networks for Pervasive Services	x	x	x		x			x	x	x	
Pervasive Healthcare and Wireless Health Monitoring	x	x	x	x		x	x	x	x	x	
A Software Engineering Framework for Context-Aware Pervasive Computing	x		x	x						x	
A Generic Mobile Agent Framework for Ambient Intelligence	x	x					x	x		x	
Totaal	8	7	7	4	3	4	3	9	6	10	2

Figuur 5

Naar aanleiding van de getoonde matrix met daarin de tien bestudeerde artikelen is onderstaande tabel te vormen met daarin de eigenschappen en de frequentie waarmee deze zijn besproken in de betreffende artikelen. Prestatie, veiligheid en aanpasbaarheid staan hierbij in de top drie van meest voorkomende eigenschappen van omgevingsbewuste systemen. Op basis van deze literatuurstudie kan dan ook geconcludeerd worden dat dit de drie belangrijkste eigenschappen van omgevingsbewuste systemen zijn. Deze worden weergegeven in figuur 6.

Eigenschap	Voorgekomen in literatuur
Prestatie	10
Veiligheid	9
Aanpasbaarheid	8
Omgevingsbewustheid	7
Alomvertegenwoordigheid	7
Beschikbaarheid	6
Bruikbaarheid	4
Proactiviteit	4
Betrouwbaarheid	3
Zelfonderhoudbaarheid	3
Instelbaarheid	2

Figuur 6

Hoofdstuk 3 – Risico's

In dit hoofdstuk komen de risico's van omgevingsbewuste systemen aan bod. Er zal gekeken worden naar de risico's die een omgevingsbewust systeem met zich mee zou kunnen brengen. De definitie van de term risico wordt bekeken en hoe er met risico's kan worden omgegaan.

Betreffende deelvraag:

- Wat zijn risico's en hoe werkt risicobeperking en analyse?

3.1 Wat is een risico?

Volgens het van Dale woordenboek heeft het woord risico de volgende betekenis: *“ri·si·co het, de; o en m -'s gevaar van schade of verlies:~ lopen; op eigen ~”*

Het gevaar van schade of verlies. Dit betreft de term risico in de werkelijkheid. Alhoewel men binnen de informatie technologie zoveel mogelijk tracht de modellen exacte afspiegeling te laten zijn van deze werkelijkheid kan er om wat voor reden dan ook nog afwijking zitten tussen model en werkelijkheid. De vraag is dan ook of binnen de informatie technologie de definitie van de term risico identiek is aan de definitie gegeven in het van Dale woordenboek.

Bij een risico zijn twee factoren van belang:

- de kans (waarschijnlijkheid) dat het risico optreedt
- de impact (gevolgen) die het risico zal hebben

Door een van twee bovengenoemde factoren zoveel mogelijk te beperken kan het effect van een risico worden beperkt.

Aubert, Patry en Rivard beschrijven in [15] een aantal verschillende definities van de term risico. Hierbij houden zij rekening met de context waarin de term risico in geplaatst dient te worden. Deze context wordt met name bepaald door de sector waarin er naar risico's wordt gekeken. In de financiële sector kijkt men anders naar risico's dan in de medische wereld. Logischerwijs wordt er dan ook anders mee omgegaan.

In het bedrijfsleven kan men bijvoorbeeld verzekeringen afsluiten om zich tegen risico's te verzekeren. In de financiële sector bepalen de kans en impact van een risico vaak hoogte van rentes, termijn van leningen en dergelijke. In dergelijke gevallen hoeven risico's niet per definitie negatief te zijn of vallen de risico's in te calculeren, af te vangen of de impact te beperken.

Een andere definitie wordt genoemd door Forst, Allen, Porter en Bloodworth in [16]:

“Risks are uncertain future events which could influence the achievement of the organisations objectives, including strategies, operational, financial and compliance objectives.”

Uit deze definitie komt duidelijk naar voren dat het een risico een negatieve invloed kan hebben op vooraf gestelde doelen. Hoewel een risico niet per definitie negatief hoeft te zijn, wordt het begrip doorgaans wel met een negatief effect of negatieve uitwerking geassocieerd.

3.2 Risico Analyse

Risico analyse bestaat uit drie facetten:

- Identificatie
- Evaluatie
- Respons

Risico analyse is het traject van identificeren, evalueren en beperken van mogelijke risico's. Het reageren nadat een risico is voorgekomen, wordt respons genoemd. Frijns bespreekt deze facetten uitvoerig in [17].

3.2.1 Identificatie

Het identificeren van risico's is het vaststellen van risico's die mogelijk kunnen optreden. Hoe zorgvuldiger het identificeren van de risico's gebeurt, des te nauwkeuriger de latere risico evaluatie en respons hierop zullen zijn. De identificatie gebeurt op basis van ervaring en mogelijk op basis van simulatie, het nabootsen van een bepaalde situatie.

3.2.2 Evaluatie

Tijdens of nadat een risico zich heeft voorgedaan zal er gekeken dienen te worden wat de kans was dat dit risico op zou treden en wat het effect (impact) van dit risico is. Dit zijn de twee eerder genoemde factoren kans en impact zoals besproken in 3.1.

De evaluatie of analyse van risico's kan gebeuren op twee manieren. Kwalitatief en kwantitatief.

Kwalitatieve risico analyse: De kennis en ervaring van een risico analyse expert wordt gebruikt om het risico of de risico's te analyseren

Kwantitatieve risico analyse: Analyse wordt gebaseerd op basis van meetbare waarden. Vaak met behulp van een wiskundig model.

In geval van zeer grote belangen wordt er doorgaans gebruik gemaakt van kwantitatieve risico analyse. Het is dan uiteraard wel noodzakelijk dat er in het verleden data is verzameld waaruit op wiskundige basis kan worden afgeleid wat kans en impact van een dergelijk risico naar waarschijnlijkheid zal zijn.

3.2.3 Respons

Nadat de risico identificatie en evaluatie heeft plaatsgevonden, wil men gaan bepalen wat er te doen valt om de kans van herhaling van het risico en bijbehorende impact zo klein mogelijk te houden. Bij terugkerende risico's kan dit worden vastgelegd in procedures en processen. Bij minder vaak voorkomende en derhalve onvoorziene risico's zal er ad hoc gehandeld dienen te worden op basis van de (hopelijk) aanwezige kennis en ervaring.

3.3 Risico's van omgevingsbewuste systemen

Het invoeren en in gebruik nemen van omgevingsbewuste systemen kan een aantal risico's met zich mee brengen. Het is zaak om hierin onderscheid te maken tussen risico's van omgevingsbewuste systemen in thuis situaties en in bedrijfs situaties. De impact in een bedrijfs situatie of professionele organisatie is logischerwijs groter dan de impact in een thuissituatie. Eventuele thuiszorg en het monitoren van vitale lichaamsfuncties daargelaten. In deze paragraaf komen risico's aan bod waarbij onderscheid is gemaakt tussen een thuis en een bedrijfs situatie.

In [11] wordt door Edwards en Grinter een zevental voorbeelden van risico's van omgevingsbewuste systemen in een thuissituatie besproken. Deze risico's zijn als volgt:

3.3.1 Upgraden van woningen

De huidige woningen zijn niet uitgerust met omgevingsbewuste systemen en bijbehorende apparatuur. Experimentele woningen daargelaten. Een bestaande woning zal dan ook volledig uitgerust moeten worden met sensoren, processoren en bijvoorbeeld camera's om zo een omgevingsbewuste woning te kunnen worden. Er worden verschillende technologieën toegepast die stuk voor stuk in gebruik zullen worden genomen en niet als in één keer aangeschaft zullen worden. Dit vergroot de kans dat de apparatuur niet goed werkt vanwege de diverse fabrikanten waarvan deze afkomstig is.

Tevens zal het voor bewoners van een woning met een omgevingsbewust systeem een flink stap zijn om de transitie van een “dom” naar een “slim” huis te kunnen verwerken. De grote hoeveelheden nieuwe en complexe techniek die zijn toetrede doet in de woning kan voor de gebruiker te complex zijn waardoor problemen ontstaan waar hij geen wijs uit wordt en waar expertise voor nodig is om deze problemen te verhelpen of toe te lichten naar de gebruikers toe.

3.3.2 Slecht voorbereide informatie uitwisseling

Zoals toegelicht in het vorige nadeel waarbij onderdelen van omgevingsbewuste systemen los worden aangeschaft en afkomstig zijn van verschillende fabrikanten brengt dit het risico met zich mee dat deze losse onderdelen niet goed zullen samenwerken waardoor slechts een fractie van de volledige functionaliteit behaald zal worden. Het doel van het omgevingsbewuste systeem als geheel wordt dan voorbij gestreefd en de gewenste functionaliteit zal verdeeld zijn over losse eilanden die niet als geheel opereren. De individuele apparaten zullen dan ook ontwikkeld moeten worden met het voornemen om direct en volledig te kunnen samenwerken met alle andere apparaten zonder dat hier een uitgebreid configuratie en installatie proces aan vooraf dient te gaan. Edwards en Grinder verwachten dat er nieuwe standaard protocollen ontwikkeld zullen moeten worden om hier een standaard in te vinden zodat de verscheidene apparaten op eenvoudige wijze samen te laten functioneren.

3.3.3 Systeemonderhoud

Met de opkomst van de computer en zelfs meerdere computers in ieders huishouden is het onderhouden van deze systemen een hele klus geworden. Hardware en software moeten worden bijgewerkt en geupgrade, nieuwe apparaten gekoppeld worden en ieder apparaat moet voorzien zijn van een mogelijkheid om internet te kunnen. Bij vergelijkbare apparaten in het huishouden zoals een wasmachine of een droger gaat een gebruiker doorgaans ook niet zelf aan de slag om herstelwerkzaamheden uit te voeren. Daar hij hier de kennis niet voor heeft schakelt hij in zo'n geval vaak een expert in die het apparaat voor hem herstelt. Kan men dan verwachten dat bij een defect aan een geavanceerd omgevingsbewust apparaat dit wel zelf wordt opgelost of moet hier ook een expert voor worden ingeschakeld?

Een alternatief voor een zeer ervaren thuisgebruiker met kennis over de situatie, is een optie om de apparatuur op afstand te kunnen herstellen of configureren middels een derde partij die hiervoor de kennis in huis heeft. Uiteraard moet hier wel het veiligheidsaspect in mee worden genomen om te voorkomen dat onbedoeld het systeem bloot staat voor aanvallen van buitenaf.

3.3.4 Ontwerp voor thuisgebruik

Anders dan bij het ontwerpen van omgevingsbewuste systemen in bedrijven, hoeft er er hiervoor in een thuisomgeving niet op uitgebreide wijze aandacht aan besteed te worden zoals bij gebruik in een bedrijfsomgeving wel het geval is. In [12] wordt dit uitvoerig beschreven. Er zal wel door de thuisgebruikers geaccepteerd moeten worden dat de opkomst en het gebruik van omgevingsbewuste systemen geleidelijk zal gaan en het enige tijd kan duren voordat dit geruisloos geïntegreerd zal zijn in de woonomgeving. Dit zal volgens Edwards en Grinter in [11] op dezelfde manier gaan als de acceptatie en integratie van een vaste telefoonlijn en elektriciteit in het verleden. De uitvinder van de telefoon was heilig overtuigd van het succes van de uitvinding terwijl de verkopers er een hard hoofd in hadden. Uiteindelijk is gebleken dat Bell toch zijn gelijk heeft gekregen door de gebruikers die het nut van deze uitvinding toch naar waarde wisten te schatten. Hetzelfde gold voor draadloze netwerken voor computers waarvan men in eerste instantie niet dacht dat dit een groot succes zou worden. Met komst van de smartphone en andere portable apparaten zoals laptops is dit uiteindelijk toch het geval gebleken. Tezamen met SMS zijn dit tal van voorbeelden waarmee wordt aangetoond dat nieuwe technologie, zij het na enige tijd, uiteindelijk toch zijn intreden doet in het huishouden.

3.3.5 Sociale implicaties

In [12] hebben Abowd en Mynatt de sociale risico's van omgevingsbewuste systemen aan de kaak gesteld. Met name het privacy vraagstuk is hierin uitvergroot. Bij de intrede en het gebruik van omgevingsbewuste systemen waarbij allerlei sensoren en computers de bewegingen van gebruikers op de voet volgen, is dit een terecht vraagstuk. Edwards en Grinter halen in [11] echter nog een aantal risico's aan op sociaal niveau. Dit gaat om het de onderwerpen die zij betiteld hebben als het besparen van werk en het goede ouderschap. Het besparen van werk, in dit geval in het huishouden, is met bijvoorbeeld de komst van de wasmachine een stuk realiteit geworden. Hetzelfde geldt voor andere huishoudelijke apparaten die ten behoeve van de gebruiker zijn uitgevonden met het doel om taken uit handen te nemen en zo werk te besparen. Door het feit dat met behulp van deze nieuwe apparaten het werk in het huishouden aanzienlijk is verminderd, is men wel anders gaan denken volgens Edwards en Grinter. Men is makkelijker gaan denken over deze werkzaamheden. Vaak worden deze nu simpelere taken vooral door vrouwen uitgevoerd. Met de komst van de televisie in het huishouden is de rol van de ouder centraal komen te staan. Welke programma's morgen er door kinderen wel en niet worden gekeken en hoe lang mag er televisie gekeken worden. Hier is meer sociale controle in het gezinsleven voor nodig met alle bijbehorende afspraken van dien. Hetzelfde gold voor de intrede van de mobiele telefoon. Hierdoor zijn kinderen zelfstandiger geworden en heeft ze meer verantwoordelijkheid gegeven.

De besproken voorbeelden in [11] tonen aan dat de komst van techniek in het huishouden een grotere invloed heeft dan alleen het bieden van bepaalde functionaliteit. Over deze invloed zal door ontwikkelaars van de techniek van te voren nagedacht dienen te worden.

3.3.6 Betrouwbaarheid

Naast de bestaande technische apparaten in het huishouden zal er ook aan de nieuwe omgevingsbewuste systemen een bepaalde verwachting worden gesteld van betrouwbaarheid en stabiliteit. Bestaande apparaten als een televisie of televisie decoder zijn in dien mate ontwikkeld dat het systeem vrijwel nooit vastloopt of crasht. Dit is anders dan bij een desktop PC of een laptop die een stuk frequenter kuren vertoont. Dit verschil is terug te herleiden naar een aantal verschillen in het ontwikkeltraject van dergelijke apparaten:

- **Ontwikkelcultuur:** ontwikkelaars houden rekening met de benodigde stabiliteit. Het is vaak ook niet mogelijk om ter plekke updates of verbeteringen door te voeren. Hiervoor moet een apparaat vaak retour gestuurd worden wat een fabrikant logischerwijs zoveel mogelijk probeert te voorkomen.
- **Technische benadering:** ontwikkelaars van bijvoorbeeld televisie decoders ontwikkelen een systeem dat als geheel functionaliteit toevoegt, zonder afhankelijk te zijn van andere onderdelen of systemen in de omgeving. Hierdoor is dit systeem stabiel daar dit als geheel uitvoerig is getest en derhalve de voortreffelijke betrouwbaarheid levert. Desktop computers bevatten veel zelfstandige onderdelen die ook nog eens dienen samen te werken met andere apparaten en netwerken. Hierdoor is het afhankelijk van andere systemen en fabrikanten waardoor het niet de gebundelde functionaliteit als goed getest geheel zelf kan leveren. Bij het falen van een bepaald onderdeel van een systeem is de wens om de rest van het wel operationeel te houden. Er dient dan wel een vorm van redundantie ingebouwd te worden. Iets wat een simpele integratie en ingebruikname niet ten goede komt.

- Verwachting van de markt: de markt en dus ook de gebruikers van apparaten hebben een bepaalde verwachting van moderne apparatuur. Voor nieuwe apparaten liggen deze verwachtingen aanzienlijk hoger dan dat bij de PC het geval was. Hierbij is van oudsher een acceptatie voor crashes ingeslopen. Bij nieuwe apparaten en systemen wordt dit simpelweg niet door de gebruikers geaccepteerd.
- Regelgeving: Aan veel apparaten wordt er door de overheid eisen gesteld aan de kwaliteit en veiligheid van apparaten alvorens deze verkocht mogen worden.

Bovenstaande vier punten hebben bijgedragen aan de betrouwbaarheid van nieuwe apparaten. Al deze punten zullen ook in acht genomen dienen te worden bij de ontwikkeling van omgevingsbewuste systemen om zodoende de betrouwbaarheid te kunnen waarborgen.

3.3.7 Systeemintelligentie

Bij het werken met omgevingsbewuste systemen bestaat het risico's dat het systeem middels sensoren bepaalde signalen opvangen en vervolgens hiernaar onjuist handelen. Er zit een groot verschil tussen de werkelijkheid en wat een omgevingsbewust systeem als werkelijkheid ervaart. Voorbeeld hiervan is bijvoorbeeld dat een omgevingsbewust systeem bepaalde functionaliteit inschakelt omdat het opmerkt dat de chip / kaart van een gebruiker zich in een bepaalde ruimte bevindt. Dit terwijl de kaart daar mogelijk is achtergelaten en is vergeten door de gebruiker. Het omgevingsbewuste systeem trekt conclusies en voert acties uit op basis van wat hij opvangt uit de omgeving. Bij bepaalde twijfel of onduidelijke signalen uit de omgeving zal het systeem over een bepaalde intelligentie dienen te beschikken waarmee hij deze situaties met goed gevolg kan inschatten en afhandelen. Er zijn verschillende stadia van intelligentie van een omgevingsbewust systeem. Deze stadia staan beschreven in [13]. Hierin heeft Dey de volgende situaties beschreven:

- Het systeem interpreteert op basis van informatie verkregen via sensoren er een bepaalde situatie is en handelt hiernaar. Bijvoorbeeld de aanwezigheid van een ID kaart in een bepaalde ruimte.
- Het systeem telt een aantal bevindingen op op basis verkregen informatie. De aanwezigheid van meerdere mensen in een ruimte zou kunnen betekenen dat er een vergadering plaatsvindt.
- Het systeem handelt op basis van de eerdere opgetelde bevindingen en de hieraan verbonden conclusie. Bij het vaststellen van meerdere mensen in een ruimte, interpreteert het systeem dit als een vergadering en meent dat bepaalde digitale aantekeningen van een gebruiker met alle andere gebruikers in de ruimte gedeeld dienen te worden.
- Het systeem handelt pro-actief en stelt de gebruiker die met meerdere mensen in een bepaalde ruimte is de vraag of bepaalde digitale aantekeningen gedeeld moeten worden met de andere gebruikers of deelt deze aantekeningen zelf zonder dit eerst te vragen.

Er kan in dergelijke situaties dan ook onderscheid gemaakt worden tussen een een lage, middel of hoge mate van benodigde intelligentie van een systeem. Is het simpelweg verkrijgen van informatie van sensoren voldoende om te handelen. Moet er informatie gecombineerd worden of moet hierna ook nog een bepaalde actie uitgevoerd worden. Zowel de gebruiker alsmede het omgevingsbewust systeem moet weten wat het kan verwachten en wat er van zichzelf verwacht wordt. Hierin ligt een uitdaging voor ontwikkelaars van omgevingsbewuste systemen. De intelligentie van het systeem moet zoveel mogelijk voorspelbaar, begrijpelijk en herstelbaar zijn.

Hoofdstuk 4 – UML & Camel

In dit hoofdstuk komt aan bod waarop gelet moet worden tijdens het modelleren en daarbij rekening houden met (uiteindelijke) risico's. Hierbij staat UML centraal.

4.1 UML

UML staat voor Unified Modelling Language. Deze modelleer taal is afkomstig uit de jaren negentig. UML is voortgekomen uit een samenwerkingsverband van 17 bedrijven. De gebundelde samenwerking heeft de naam Object Management Groep (OMG) gekregen. OMG deelt accreditaties uit aan bedrijven zodat zij een certificaat verkrijgen om hun vaardigheden op UML gebied aan te tonen aldus Jos Warmer & Anneke Kleppe – Praktisch UML 3de editie [14].

De oorspronkelijke versie van UML heeft versie 1.0 toebedeeld gekregen. Dit vond plaats in 1997. Dit is later doorontwikkeld tot UML 2.0, afronding hiervan was in 2004. Over de evolutie van de taal volgt later in dit hoofdstuk meer.

4.1.1 Wat is UML?

UML is een modelmatige taal om objectgeoriënteerde analyses en ontwerpen voor een informatiesysteem te kunnen opstellen. De makers van UML benadrukken dat het een taal betreft een geen methode. Het draait binnen UML niet om de werkwijze en het standaardiseren hiervan. Wel draait het om het standaardiseren van de begrippen en de uiteindelijke diagrammen waarin de begrippen terugkomen. Hierdoor kan iedereen die UML wenst te gebruiken, zijn of haar eigen werkwijze hanteren om tot een bepaald diagram te komen.

Niveau van modelleren:

UML hanteert verschillende niveau's die kunnen worden toegekend aan de modellen. Dit om vast te kunnen stellen op welk niveau iemand modelleert. Binnen UML worden deze niveau's "Modelling Maturity Levels" (MMLs) genoemd.

UML hanteert de volgende MML's:

Level 0 - Geen specificatie:

De specificatie bevindt zich volledig in het hoofd van de ontwikkelaar. Van documentatie is geen sprake. Hierdoor ontstaan conflicten tussen gebruikers en ontwikkelaars.

Level 1 - Tekstuele specificatie:

De specificatie wordt vastgelegd in 1 of meerdere documenten, puur en alleen in natuurlijke taal. Hierin worden afspraken tussen gebruiker en ontwikkelaar vastgelegd. Natuurlijke taal is echter op verschillende manieren te interpreteren, het biedt dan ook nog steeds ruimte tot discussie. Ook het bijwerken van de documentatie na eventuele wijzigingen is hier een probleem.

Level 2 – Tekst met diagrammen:

Level 2 is vrijwel gelijk aan level 1. Er is prake van tekstuele specificatie met daarbij enkele diagrammen om de tekst te verduidelijken. De nadelen die bij level 1 golden, gelder hier nogsteeds.

Level 3 – Modellen met tekst:

Bij dit niveau wordt de specificatie vastgelegd in een aantal modellen. Hiermee bedoelt men "een

diagram of tekst geschreven in een niet natuurlijke taal waarvan de betekenis helder en eenduidig is”

Level 4 – Exacte modellen:

Op niveau 4 zijn de modellen het belangrijkste onderdeel van het systeem. Tevens hebben de modellen een samenhang zodat men ook kan spreken van 1 enkel model, bestaande uit verscheidene onderdelen. Wel is er nog sprake van tekstdocumenten om de modellen te ondersteunen en een en ander toe te lichten. De modellen zijn van dergelijke kwaliteit dat ze direct in contact kunnen worden gebracht met de achterliggende (te ontwikkelen) code. Een goede uitgangspositie voor ontwikkelaars.

Level 5 – Alleen modellen:

Op niveau zijn de modellen volledig en precies genoeg om alle details te schrijven. De code kan volledig uit de modellen gegenereerd worden. De modelleer taal is op dit niveau een soort van hoger niveau programmeertaal geworden.

4.2 Wat zijn de componenten van UML?

Use case diagram(I):

Toont hoe een systeem gebruikt kan worden door externe entiteiten, bijvoorbeeld gebruikers.

Klassediagram(II):

Toont de statische structuur van het systeem, gespecificeerd in klassen en onderlinge relaties tussen klassen.

Objectdiagram(II):

Toont de structuur van statische objecten (instanties van klassen) en onderlinge relaties.

Sequencediagram(III):

Toont de volgorde in tijd van boodschappen / communicatie binnen het systeem.

Communicatiediagram(III):

Toont de onderlinge communicatie tussen objecten en hun samenwerking om het doel te bewerkstelligen

Toestanddiagram(III):

Toont de toestanden waarin een object zich kan bevinden gedurende zijn cyclus.

Activiteitsdiagram(III):

Toont de activiteiten die door een deel van het systeem worden uitgevoerd. Tevens worden deze diagrammen gebruikt om bedrijfsprocessen en workflow te modelleren.

Componentdiagram(III):

Toont hoe de verdeling in componenten binnen een systeem en de relaties tussen deze componenten.

Deploymentdiagram(III):

Toont hoe de softwarecomponenten in een bepaalde systeemconfiguratie worden gebruikt.

Het usecasediagram(I) wordt ook wel het eisendiagram genoemd. Hierin wordt vastgesteld wat de functionele eisen van een systeem zijn. Oftewel, wat moet het systeem kunnen c.q. waar dient het systeem op functioneel niveau aan te voldoen? Per wenselijk handeling wordt een zogenaamde usecase opgesteld waaruit afgeleid kan worden wat de gebruiker van het systeem verwacht.

Het klassediagram(II) en objectdiagram(II) vormen de statische diagrammen. Zij geven immers de statische toestand van het systeem weer. Deze klassen kunnen worden aangeroepen in de vorm van objecten.

Het sequence(III), communicatie(III), toestand(III) en activiteitsdiagram(III) worden de dynamische diagrammen genoemd. Deze diagrammen tonen de volgorde van handelingen, toestanden en activiteiten binnen het systeem. Dit verschilt constant, vandaar de naam dynamische digrammen.

Tenslotte betreffen het component(III) en deploymentdiagram(III) de implementatiediagrammen. Deze diagrammen worden betrokken bij de implementatie (laatste fase) van de systeemontwikkeling. Diagrammen in fases:

De beschreven diagrammen kunnen op ieder moment tijdens het ontwikkelen van een systeem worden opgesteld. De praktijk leert echter dat de diagrammen zich verder ontwikkelen en zo ook uitgebreid worden naar mate de ontwikkeling van een systeem vordert. De negental beschreven diagrammen komen voor in verschillende fases van het ontwikkel traject van een systeem.

In [14] wordt onderscheid gemaakt tussen:

- Domeinmodel of -diagram: model of diagram waarin alleen aspecten uit de werkelijkheid waarin het systeem moet gaan werken (domein), geen automatiseringsaspecten.
- Applicatiemodel of -diagram: model of diagram waarin naast domeinaspecten ook keuzes naar voren komen die betrekking hebben op hoe de gebruiker met dit systeem kan gaan werken. Een applicatiemodel is een uitbreiding op het domeinmodel maar beide zijn verschillende ontwikkel stadia van een en hetzelfde model.
- Implementatiemodel of -diagram: model dat exact weergeeft hoe de implementatie van het systeem eruit zal gaan zien. Het bevat alle aspecten uit de vorige stadia met daarbij aspecten van de implementatie. Dit zo veel mogelijk uitgewerkt zodat deze zo goed mogelijk kunnen worden omgezet naar code.

Onderstaande tabel geeft weer in welk stadia de diagrammen kunnen voorkomen:

	domein	applicatie	implementatie
usecasediagram	nee	ja	nee
klassediagram	ja	ja	ja
objectdiagram	ja	ja	ja
sequencediagram	beperkt	ja	ja
communicatiediagram	beperkt	ja	ja
toestandsdiagram	ja	ja	ja
activiteitsdiagram	Ja, workflow	ja	ja
deploymentdiagram	nee	nee	ja
componentdiagram	nee	nee	ja

Figuur 7

4.2.1 Diagrammen

Usecase diagram

Use case-diagrammen beschrijven de relaties en afhankelijkheden tussen een groep van use cases en de actoren die deelnemen aan het proces.

Belangrijk om op te merken is dat usecase diagrammen niet geschikt zijn om het ontwerp te representeren en niet het inwendige van een systeem kunnen beschrijven. Usecase diagrammen zijn bedoeld om de communicatie met de toekomstige gebruikers van een systeem te vergemakkelijken en zijn in het bijzonder behulpzaam bij het vaststellen van welke benodigde functionaliteit een systeem moet hebben. Usecase diagrammen vertellen wat het systeem moet doen maar specificeren niet hoe dit gerealiseerd moet worden.

Een usecase beschrijft vanuit het standpunt van de actoren een of meerdere activiteiten in een systeem die een concreet, tastbaar resultaat oplevert(en).

Usecases zijn beschrijvingen van kenmerkende interacties tussen de gebruikers van een systeem en het systeem zelf. Zij representeren de externe interface van het systeem en specificeren een soort pakket van eisen die het systeem moet uitvoeren.

Bij het werken met use cases is het belangrijk om enkele eenvoudige regels in acht te nemen:

- Iedere use case is gerelateerd aan tenminste één actor
- Iedere use case heeft een initiator (bijv. een actor)

Use cases kunnen ook relaties met andere use cases hebben. De drie meest karakteristieke relaties tussen use cases zijn:

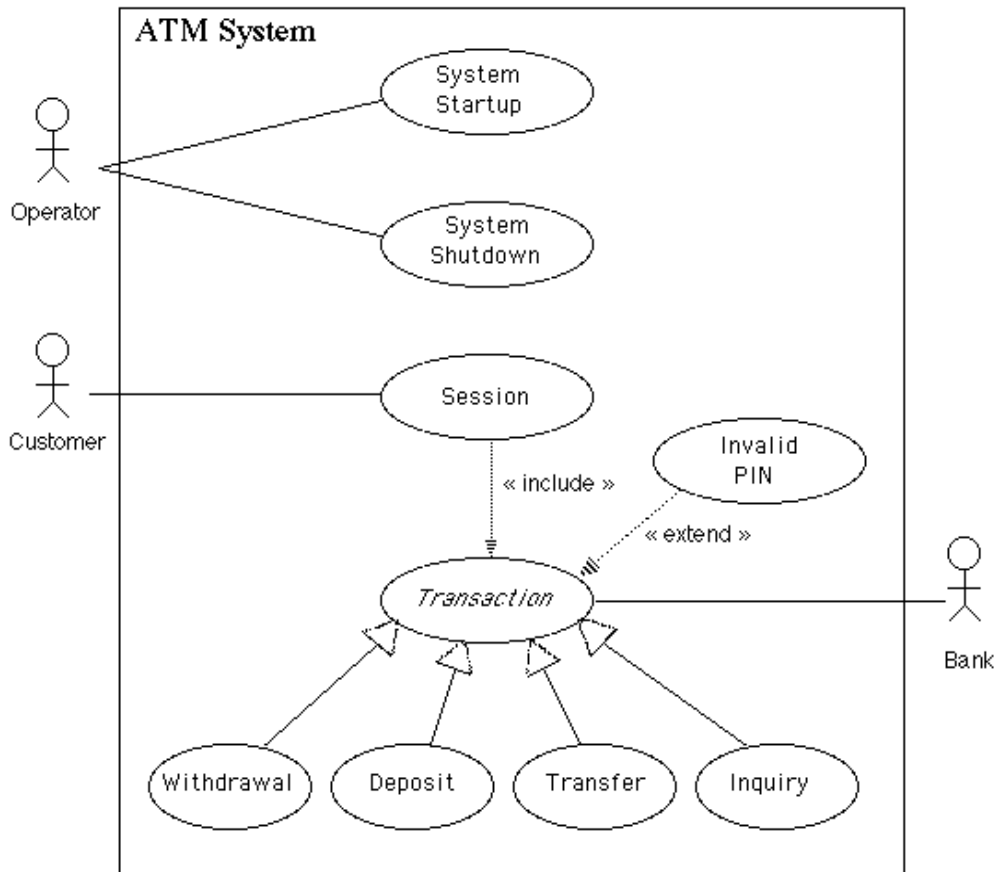
- Include: geeft aan dat een usecase zich binnen een andere use case afspeelt
- Extend: geeft aan dat in bepaalde situaties, of op een zeker moment (ook wel het uitbreidingspunt genoemd) een usecase uitgebreid zal worden met een andere.
- Generalisatie: geeft aan dat een use case de karakteristieken erft van de “super”-usecase, en sommige ervan kan herdefiniëren of nieuwe kan toevoegen, op eenzelfde wijze als bij de overerving bij klassen het geval is.

Actor:

Een actor is een externe entiteit (buiten het systeem) die samenwerkt met het systeem door deelname aan (en veelal door initiëren van) een usecase. In de dagelijkse werkelijkheid kunnen actoren mensen zijn (bijvoorbeeld gebruikers), andere computersystemen of externe gebeurtenissen.

Actoren representeren niet de fysieke mensen of systemen, maar hun rol. Dit impliceert dat als een persoon met het systeem samenwerkt op verschillende manieren (een persoon kan immers meerdere rollen aannemen) hij door meerdere actoren voorgesteld zal worden. Bijvoorbeeld een persoon die telefonische klanten ondersteuning geeft en orders invoert van de klant in het systeem. In dit geval gaat het om een persoon in de rol van de medewerker van de afdeling ondersteuning en een rol van de medewerker van de afdeling verkoop.

Afbeelding: voorbeeld van een usecase diagram:



Figuur 8

Toelichting model:

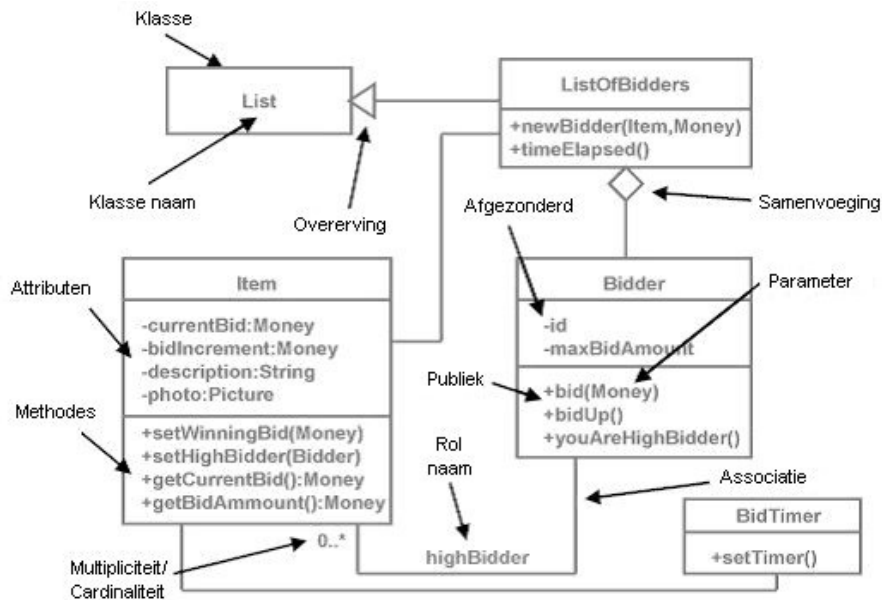
In bovenstaand voorbeeld zijn drie actoren te zien. De customer(klant), de operator (beheerder) en de bank. Klanten kunnen middels de activiteit sessie een bepaalde transactie uitvoeren. Deze transactie kan bestaan uit vier verschillende subactiviteiten (opnemen, storten, overmaken, opvragen). De bank heeft inzage in de transacties van de klanten. De beheerder is, in dit eenvoudige voorbeeld, puur en alleen afhankelijk voor het aan en uitzetten van het systeem.

De begrippen include en extend zijn reeds toegelicht. Dit geeft aan dat dat ze deel uitmaken van een andere, reeds bestaande, usecase.

Klassendiagram

Het klassendiagram beschrijft de klassen die een systeem vormen en het statische verband tussen de klassen. De klassen worden gedefinieerd in termen van hun naam, attributen (of gegevens), en gedrag (of methodes). De statische relaties zijn associatie, aggregatie, en overerving.

Afbeelding: voorbeeld van een klassendiagram in UML



Figuur 9

Toelichting model:

Object: Een specifieke entiteit of concept die een betekenis heeft in een applicatie domein.

Klasse: Een definitie van een verzameling potentiële objecten die dezelfde gegevens, gedrag, en relaties hebben.

Attributen: Een gegevenswaarde, binnen een object, die gedefinieerd in een klasse die een betekenis heeft binnen het applicatie domein.

Gedrag: Een door een object geleverde service die gedefinieerd wordt door een klasse.

Methode: De implementatie van gedrag in een object-georiënteerde programmeertaal.

Associatie: Een relatie tussen klassen.

Samenvoeging: Een geheel/deel verband tussen klassen.

Overerving: Een generalisatie/specialisatie verband tussen klassen.

Cardinaliteit/Multipliciteit: Het minimum en maximaantal objecten die aan een associatie of een samenvoeging deelnemen. De meest voorkomende (interessante) zijn `0..*`, `0..1`, `1..*`, en `1..1`.

Polymorfisme: De capaciteit om een bericht naar een voorwerp te verzenden zonder zijn specifieke klasse te kennen.

Objectdiagram

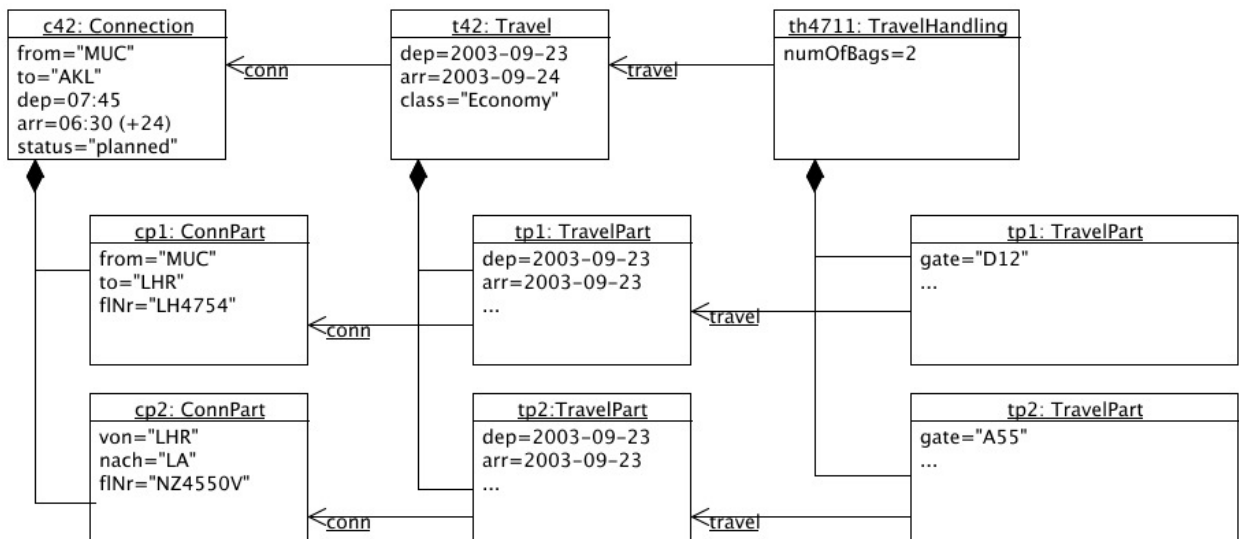
Een objectdiagram is een instantie van een klassendiagram. Doorgaans wordt een objectdiagram niet gebruikt, enkel en alleen om een wat moeilijker te begrijpen klassendiagram goed toe te lichten of te verduidelijken.

Verschil tussen klassen- en objectdiagram:

Beide diagrammen tonen de elementen waar het systeem uit bestaat en de onderlinge structurele relaties. Het klassendiagram toont de klassen, attributen van klassen operaties en associaties. Dit betreft de structuur waaraan de objecten uit klassen gebonden zijn.

Een object diagram is een weergave op een bepaald moment van objecten die gecreëerd zijn volgens de structuur van het klassendiagram. Een objectdiagram is dan ook te herkennen aan de attribuut waarden die zijn waar te nemen in het model.

Afbeelding van een objectdiagram.

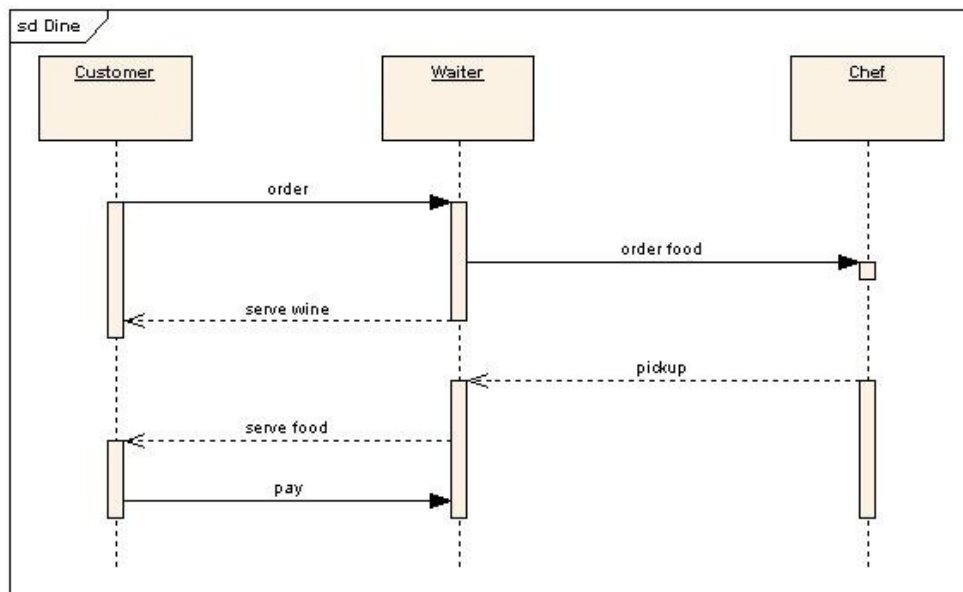


Figuur 10

Sequencediagram

Een sequencediagram beschrijft hoe groepen objecten in het verwezenlijken van het systeemgedrag samenwerken met hierbij de nadruk op het tijdsaspect. Deze samenwerking wordt uitgevoerd als reeks berichten tussen objecten. Het sequencediagram beschrijft de gedetailleerde implementatie van één enkele use case (of één variatie van één enkele usecase). De sequencediagrammen zijn niet nuttig om het gedrag binnen een object te tonen.

Afbeelding: Sequencediagram – plaatsen van bestelling in restaurant:



Figuur 11

Toelichting model:

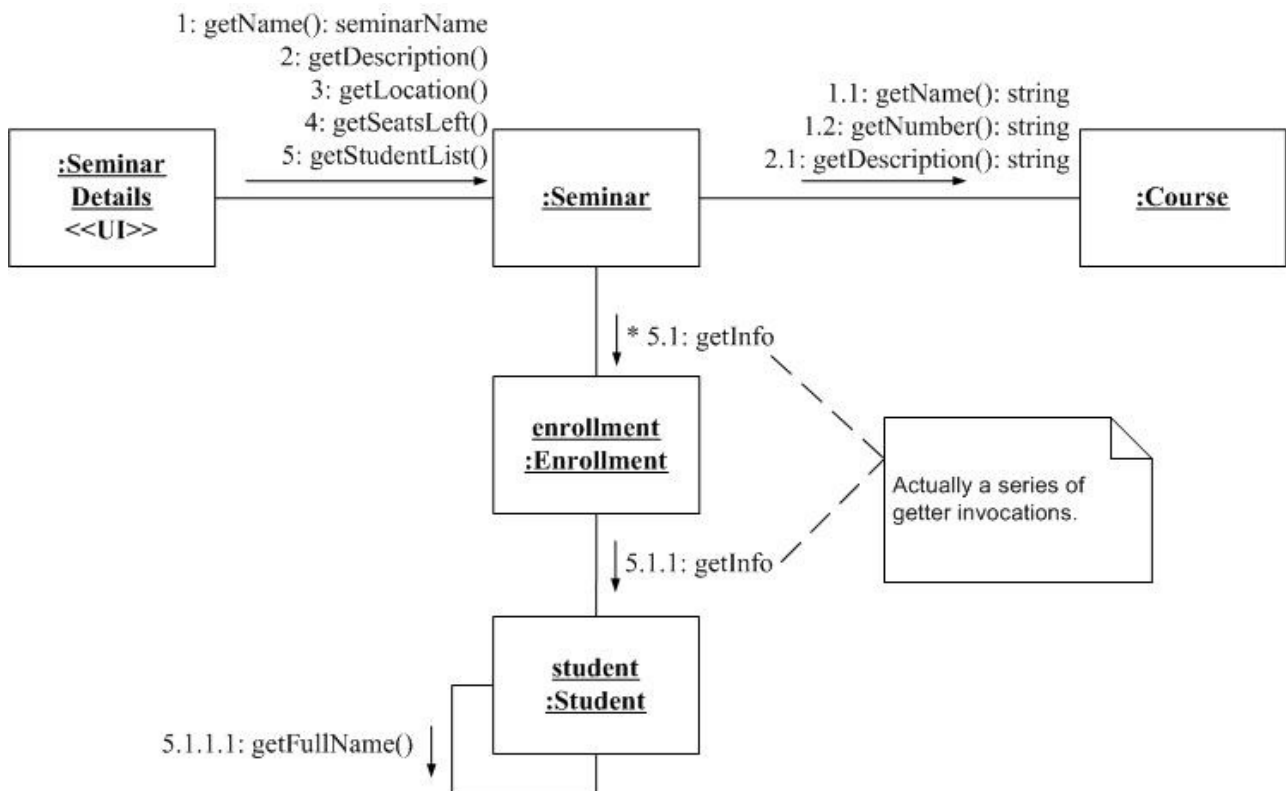
- 1 -Klant bestelt (bericht: Klant->Plaatsen bestelling->Ober)
- 2 -Ober geeft bestelling door aan Chef (bericht: Ober->Eten bereiden → Chef)
- 3 -Ober serveert Wijn (bericht: Ober-> Serveert wijn->Klant)
- 4 -Chef geeft aan ober aan dat eten geserveerd kan worden (bericht: Chef → Serveren-> Ober)
- 5 -Ober serveert eten aan klant (bericht: Ober-> Serveert eten->Klant)
- 6 -Klant rekt af bij ober (bericht: Klant-> Rekent bestelling af->Ober)

In bovenstaand voorbeeld gaat het om drie entiteiten: klant, ober en chef. In een ander voorbeeld kunnen dit onderdelen van een systeem zijn. Bijvoorbeeld verschillende klassen binnen een en hetzelfde systeem die onderling berichten uitwisselen. In het getoonde voorbeeld zijn de berichten het plaatsen van een bestelling of het afrekenen. Dit zou ook data overdracht kunnen zijn tussen systeemonderdelen. Het voornaamste is bovenstaande afbeelding en in sequencediagrammen in het algemeen is de volgorde waarin de berichtgeving onderling plaatsvindt tussen entiteiten. Het systeem dient hierop vooraf ingericht en ontworpen te zijn om zo het bericht verkeer op juiste wijze te kunnen afhandelen.

Communicatiediagram

In een communicatiediagram wordt exact dezelfde informatie weergegeven als in een sequencediagram. Het tijdsaspect is echter minder goed zichtbaar in een communicatiediagram. Een andere manier om het communicatiediagram te gebruiken is als overzicht van alle boodschappen die tussen objecten van bepaalde klassen uitgewisseld worden. In een dergelijk diagram worden alle boodschappen uit iedere interactie samengevoegd tot een overzicht. Het diagram dient dus als verzamelplaats voor alle boodschappen uit de sequence en/of communicatiediagrammen. Alhoewel dit in theorie wellicht handig lijkt, leert de praktijk anders. Zelfs bij een communicatiediagram met een beperkt aantal klassen kan dit zeer onoverzichtelijk worden. Wanneer dit het geval is, is het verstandiger om per klasse een communicatiediagram op te stellen. Een dergelijk diagram komt tot stand door een klasse te tekenen met daaromheen de omliggende klassen. Vervolgens wordt alle bericht uitwisseling tussen de middelste en omliggende klassen in kaart gebracht. Dit wordt herhaald totdat iedere klasse de middelste positie heeft vervuld. Uiteindelijk is het hele traject van bericht uitwisseling tussen klassen in kaart gebracht.

Afbeelding: communicatiediagram



Figuur 12

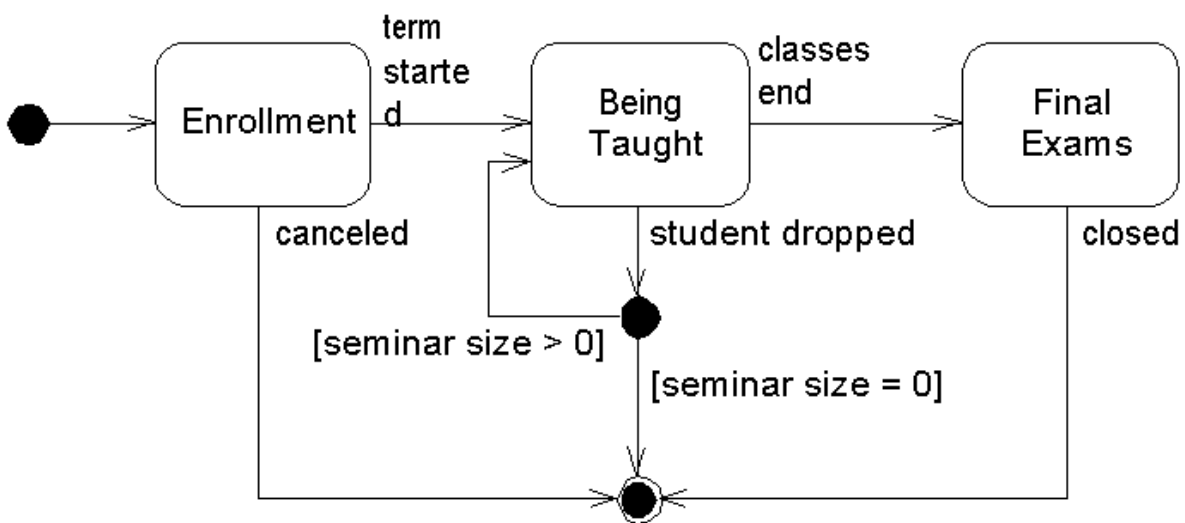
Toelichting model:

Bovenstaande afbeelding is een communicatie model dat de communicatie tussen enkele klassen toont. De centrale klasse in het geheel is de seminar klasse. Studenten die zich willen opgeven (enrollen) voor een seminar die bij bij een bepaald vak hoort (course) kunnen zich opgeven middels een enrollment. Deze berichten kunnen iteratief plaatsvinden doordat er meerdere studenten zich kunnen opgeven voor verschillende seminars. Zodra de enrollments voor alle seminars voltooid zijn, kan er een grafische weergave in lijstvorm getoond worden om zo alle opgegeven studenten voor een bepaalde seminar te bekijken.

Toestandsdiagram

Toestandsdiagrammen verschillen binnen UML van de toestandsdiagrammen binnen de traditionele softwareontwikkeling. Het verschil zit hem in het feit dat een toestandsdiagram bij UML bij een enkele klasse hoort en niet bij een volledig systeem. Een toestand wordt volgens [bron] gedefinieerd als een stabiele staat waarin een object zich enige tijd bevindt. In een toestandsdiagram wordt een toestand aangegeven middels een rechthoek met afgeronde hoeken. In dit kader staat de naam van de toestand vermeld.

Ieder object van de klasse waarbij het toestandsdiagram hoort, bevindt zich altijd in 1 van de toestanden. Een overgang van toestand A naar toestand B wordt een transitie genoemd. Een overgang van een toestand naar dezelfde toestand, wat uiteraard ook tot de mogelijkheden behoort, wordt een zelftransitie genoemd.



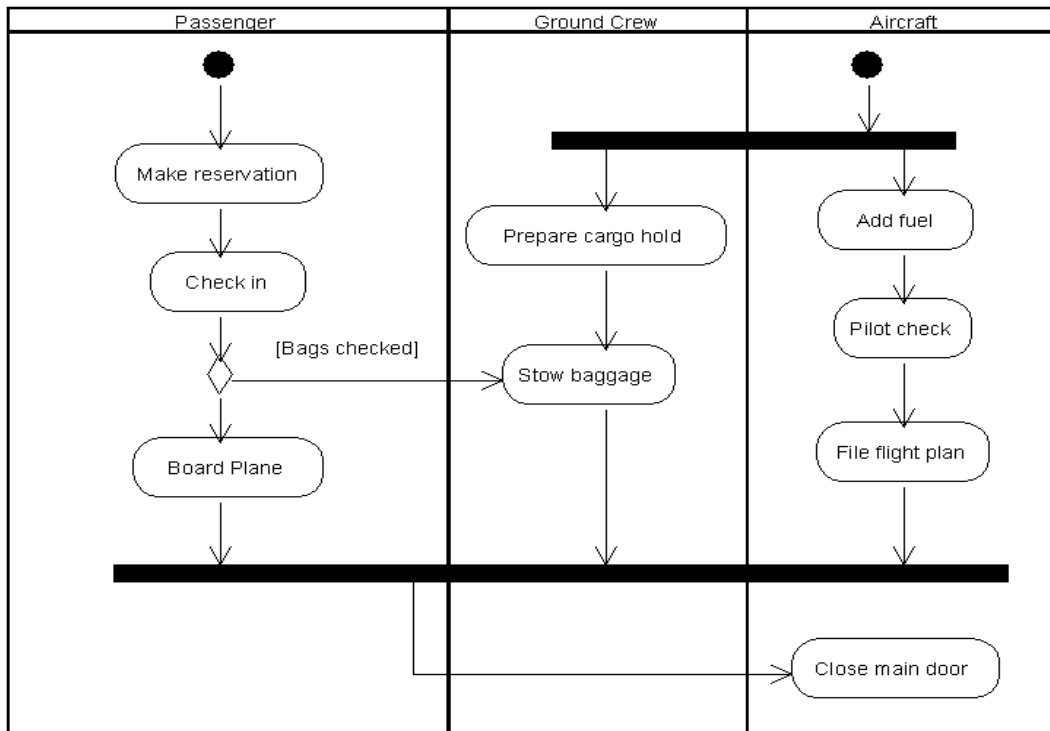
Figuur 13

Toelichting bij model:

In bovenstaand model is de toestand van een student weergegeven. Het inschrijven en volgen van een vak tot en met het examen. Binnen dit model moet de klasse student zich altijd in 1 van de genoemde toestanden bevinden.

Activiteitsdiagram

Een activiteitsdiagram is een proces gerichte beschrijving van (een deel van) een systeem. Het diagram wordt onder meer gebruikt om om een werkstroom te beschrijven. Tevens kan hiermee ook een operatie beschreven worden. Een activiteitsdiagram kan meer dan Een object beschrijven. Het nut van activiteitsdiagrammen ligt voornamelijk in het feit dat ze eventueel parallellisme van activiteiten kunnen weergeven.



Figuur 14

Toelichting model:

Op bovenstaand model zijn een deel van de activiteiten te zien die een passagier, de bemanning en een vliegtuig (parallel) doorlopen tijdens het vertrekken van een vlucht.

4.3 Context Awareness Modelling Language (CAMEL)

4.3.1 Het doel van CAMEL

CAMEL biedt volgens [BRON] ontwikkelaars de mogelijkheid om bestaande systemen uit te breiden met omgevingsbewustheid of nieuwe systemen te ontwikkelen die ook over de omgevingsbewuste eigenschap gaan beschikken. CAMEL biedt een uitbreiding op UML waaruit uiteindelijke code gegenereerd kan worden.

Het doel van CAMEL is het omgevingsbewust maken van systemen zodat de systemen zich bewust zijn van de omgeving en kunnen reageren op aanpassingen in de omgeving. Bij omgevingsbewuste systemen staan zowel het waarnemen van veranderingen in de omgeving alsmede het reageren op (wijzigingen in) de omgeving.

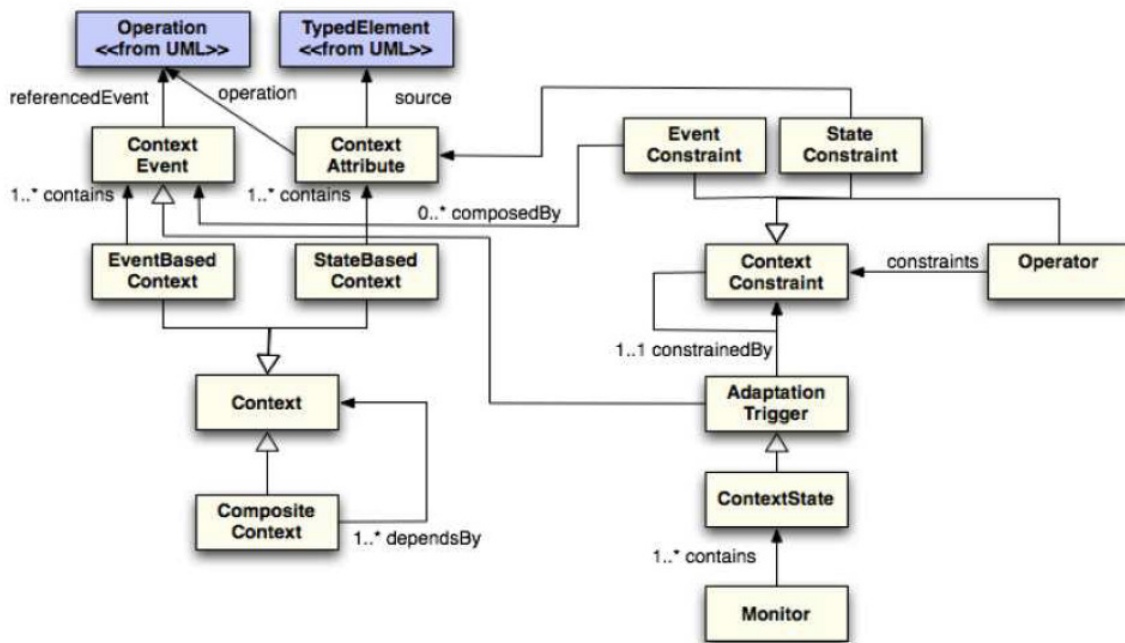
Naast het doel om een systeem omgevingsbewust te ontwikkelen zijn er uiteraard ook andere eigenschappen waaraan een systeem zal moeten voldoen. De ontwikkelaar zal een goede balans moeten zien te vinden om beide doelen op goede wijze te verwerken en integreren in een nieuw te ontwikkelen systeem. Om dit mogelijk te kunnen maken, biedt CAMEL uitkomst. CAMEL tracht een helder model gebaseerde ontwikkeling te realiseren waarmee de ontwikkelaar in staat wordt gesteld om rekening te houden met de omgevingsbewustheid van een systeem

4.3.2 De Taal CAMEL

CAMEL maakt gebruik van het Eclipse modelleer framework (EMF) en biedt naast een modelleer tool ook een mogelijkheid om uit de UML modellen rechtstreeks code te genereren op basis van een bestaand of ontworpen data model. Dit laatste zal worden geabstraheerd in deze thesis. De focus ligt op het toepassen van CAMEL en het hiermee uitbreiden van een bestaand UML model om zo omgevingsbewustheid te creëren.

Aan de hand van een tweetal meta modellen zal CAMEL worden toegelicht.

4.3.3 Monitoren van de omgeving



Figuur 15: Waarnemen / monitoren van de omgeving

Toelichting figuur 15:

De blauwe elementen zijn reeds bestaande UML onderdelen. De witte elementen zijn door CAMEL specifiek toegevoegde elementen. Het doel van dit meta model is het kunnen waarnemen van veranderingen in een de staat van een vooraf gedefinieerde omgeving.

Het meta model getoond in figuur 1 laat zien hoe de omgeving kan worden waargenomen en hoe kan worden omgegaan met veranderingen in de omgeving. Informatie afkomstig uit de omgeving wordt gemodelleerd middels “StateBasedContext” en “EventBasedContext”. StateBasedContext is een verzameling van statische gegevens uit de omgeving. Een verzameling attributen die middels ContextAttribute in verband worden gebracht met bestaande attributen uit het oorspronkelijke UML model. EventBasedContext is een gebeurtenis in de omgeving of een serie van gebeurtenissen die mogelijk van belang zijn voor het systeem.

Uit de staat van de omgeving wordt bepaalde informatie gehaald die in het “ContextAttribute” attribuut kan worden verwerkt. “ContextAttribute” wordt gekenmerkt door een naam en een relatie met een “UML:TypedElement”, dit is een bestaand attribuut dat in het reguliere UML model reeds

aanwezig is. Denk hierbij bijvoorbeeld aan bestaande data uit een systeem dat overeenkomstig is met een bepaald attribuut dat afkomstig is uit de omgeving.

“EventBasedContext” is bedoeld voor dynamische waarnemingen van de omgeving . Het bestaat uit een verzameling van waarnemingen weergegeven door “ContextEvent”. “ContextEvent” heeft ook weer een naam en relatie met een “UML:Operatie” waarbij de gebeurtenis gerelateerd is aan de desbetreffende methode van het systeem.

Door meerdere waarnemingen van de omgeving, zowel statisch als dynamisch, kan door het systeem een samengestelde werkwijze van de omgeving worden vastgesteld.

“Monitor” vertegenwoordigt de verzameling voor logisch gerelateerde aanpassingstriggers. Een trigger kan worden afgeleid op basis van relevante informatie uit de omgeving afkomstig uit de statische en dynamische gebeurtenissen in de omgeving die door de monitor worden waargenomen. Deze informatie wordt vertegenwoordigd door de “ContextState” die op zijn beurt de realisatie van “AdaptionTrigger” moet voorstellen.

Elke staat van de omgeving is gerelateerd aan “ContextConstraint” die de voorwaarden bevat waaraan de betrokken omgevings informatie moet voldoen. CAMEL bevat drie soorten “ContextConstraints” namelijk “StateConstraint”, “EventConstraint” en “Operator”.

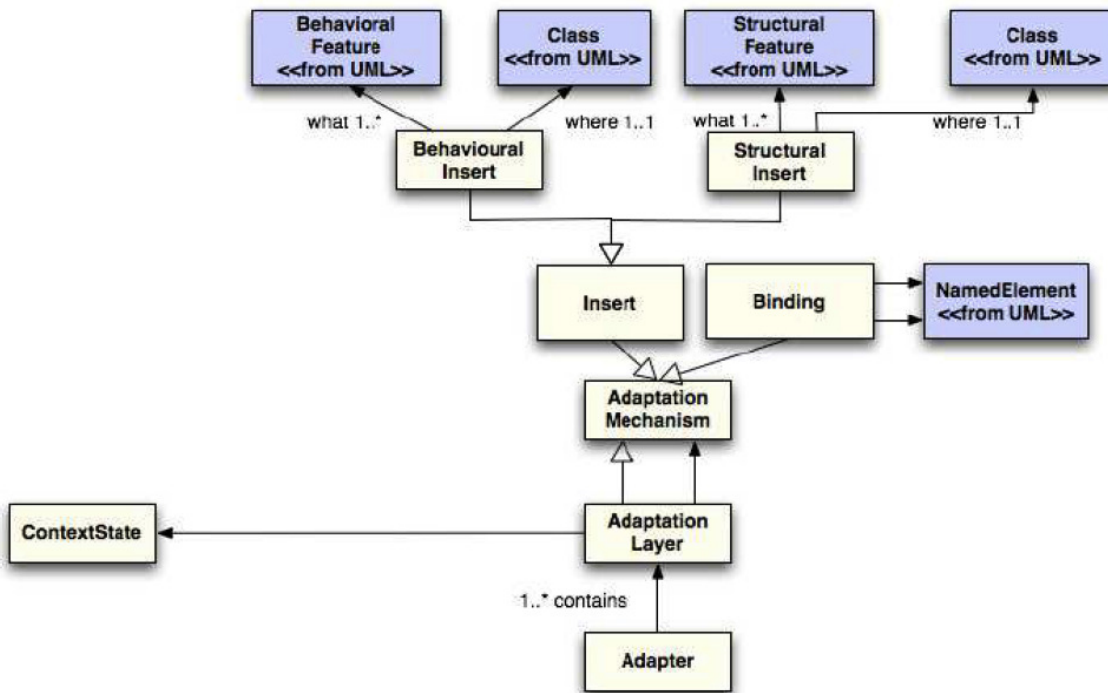
“ContextConstraint” refereert aan de gebeurtenissen in de omgeving. “StateConstraint” aan de staat van de omgeving en “Operator” zorgt ervoor dat de samengestelde staten en gebeurtenissen gecombineerd kunnen worden en dat hier een tijdelijke of logische toestand aan kan worden toegekend.

Door het ontleden van dit meta model kan de volgende geconcludeerd worden:

1. De omgeving is gedefinieerd.
2. Er treedt een verandering op in de omgeving, deze verandering wordt door het systeem waargenomen.
3. Wanneer deze verandering voldoet aan bepaalde waarden of eigenschappen zoals vastgelegd in de constraint zal deze verandering, samen met de laatste bekende toestand van de omgeving worden doorgegeven aan het systeem waardoor er een nieuwe staat van de omgeving ontstaan. Tevens kan er zo door het systeem gereageerd worden op deze wijziging in of van de omgeving.

Deze laatste handeling wordt toegelicht in het volgende meta model en bijbehorende uitleg.

4.3.4 Reageren op verandering



Figuur 16. Triggeren van aanpassingen

Figuur 16 laat het uitvoeren van een aanpassing van het systeem op de omgeving zien. Dit in navolging van het in figuur 1 toegelichte metamodel waarbij is toegelicht hoe een systeem aanpassingen van de omgeving kan waarnemen. Vervolgens wordt in dit tweede metamodel toegelicht hoe hierop gereageerd kan worden door het systeem.

Wederom zijn de blauwe elementen de bestaande UML onderdelen en zijn de witte de onderdelen die door CAMEL worden aangedragen.

Er worden door CAMEL twee inserts aangedragen, namelijk “BehaviouralInsert” en “StructuralInsert”. Dit zijn nieuwe waarnemingen van of veranderingen in de omgeving die aan het systeem kunnen worden doorgegeven. Hierbij wordt onderscheid gemaakt tussen gedrag (niet structureel) en structurele toevoegingen (inserts). Er zal vooraf vastgelegd dienen te worden waar de grens ligt tussen structureel en niet structureel. De vraag is vervolgens of dit vooraf door de ontwikkelaar wordt gedefinieerd of dat het systeem hier op zelfstandige basis een afweging in kan maken of mogelijk een combinatie van beide. Dit zal nog aan bod komen in de twee uit te werken casussen in hoofdstuk 5.

Een “Binding” combineert nieuwe informatie uit de omgeving met reeds aanwezige informatie uit het systeem om zo een nieuwe toevoeging aan het systeem tot stand te kunnen brengen. Ook dit zal worden toegelicht in de praktijkvoorbeelden in hoofdstuk 5.

Zodra er een wijziging wordt waargenomen kan er middels de “AdaptionLayer” in combinatie met de in figuur 1 toegelichte “ContextState” gezorgd worden voor een aanpassing. Deze wijziging bestaat zogezegd uit de huidige staat van de omgeving (ContextState) plus een aanpassing hierop.

4.3.5 Toepassen van CAMEL

Over het toepassen van CAMEL is in de aanwezige literatuur vrijwel niets te beginnen. Na het bestuderen van de metamodellen blijkt het volgende: de metamodellen van CAMEL kunnen gekoppeld worden met een bestaand klassendiagram uit de UML modelleer taal.

Het is dan ook zaak dat deze klassendiagrammen volledig zijn uitgewerkt. In hoofdstuk 5 zal aan de hand van twee praktijkvoorbeelden die met behulp van UML zijn uitgewerkt, getoetst worden of en in welke mate CAMEL hierop aansluit. Hierbij wordt ook het risico aspect dat in hoofdstuk 3 aan bod is gekomen, meegenomen.

Hoofdstuk 5 – Casus

5.1 Inleiding

In dit hoofdstuk wordt de theorie welke eerder aan bod is gekomen in deze thesis toegepast op een tweetal praktijkvoorbeelden c.q. casussen. De domeinen omgevingsbewuste systemen en risico's komen samen in de casussen om het verband tussen theorie en praktijk aan te tonen in deze twee specifieke voorbeelden.

Het doel hiervan is aan te tonen dat er de nodige risico's kleven aan het invoeren en met name gebruiken van omgevingsbewuste systemen. Nog meer dan bij traditionele systemen zal er rekening moeten worden gehouden met onder andere veiligheid en gevolgen van bepaalde beslissingen en handelingen van het omgevingsbewuste systeem. Aan de hand van de casussen zal aan de hand van praktijkvoorbeelden gekeken worden welke risico's een dergelijk systeem met zich meebrengt op basis van de in hoofdstuk 2 vastgestelde eigenschappen van een omgevingsbewust systeem, de in hoofdstuk 3 aan bod gekomen risico's en de eventuele uitkomst van CAMEL hierin.

5.2 Omgevingsbewuste systemen en risico's

In hoofdstuk twee zijn op basis van de gedane literatuurstudie een aantal eigenschappen naar voren gekomen die van groot belang zijn voor (omgevingsbewuste systemen). Onderstaande tabel was de uitkomst van de literatuurstudie en toont de frequentie waarmee de eigenschappen in de literatuur voorkwamen. Op de top 3 van de onderstaande eigenschappen zal worden ingezoomd. Naar aanleiding van de twee uitgewerkte casussen zal worden gekeken hoe deze eigenschappen in de praktijkvoorbeelden een rol spelen en wat hierbij mogelijke risico's zijn.

Eigenschap	Voorgekomen in literatuur
Prestatie	10
Veiligheid	9
Aanpasbaarheid	8
Omgevingsbewustheid	7
Alomvertegenwoordigheid	7
Beschikbaarheid	6
Bruikbaarheid	4
Proactiviteit	4
Betrouwbaarheid	3
Zelfonderhoudbaarheid	3
Instelbaarheid	2

Figuur 17

In hoofdstuk drie zijn een aantal risico's van omgevingsbewuste systemen besproken. In ieder van beide casussen zullen de voornaamste risico's in verband worden gebracht met de drie eerdere, in bovenstaande tabel genoemde, eigenschappen.

Risico's van omgevingsbewuste systemen zoals aan bod gekomen in hoofdstuk 3:

- Technische falen
- Gebrek aan kennis
- Communicatie issues
- Privacy / Veiligheid
- Veilig
- Betrouwbaarheid
- Systemintelligentie

Zoals toegelicht in hoofdstuk drie bestaat risico uit enerzijds de kans dat het risico voorkomt en aan de andere kant de impact die een bepaald risico heeft. Daar de casussen fictief zijn er er niet genoeg (technische) informatie voorhanden is om een concrete berekening van kans & impact te maken, worden deze berekeningen dan ook geabstraheerd. Wel zal via toelichting per relatie tussen risico en eigenschap worden aangetoond dat er een verband is. Dit ook op basis van de eerder uitgewerkte theorie in hoofdstuk twee en hoofdstuk 3.

Er is bewust voor deze casussen gekozen om aan te tonen dat een disfunctionerend of niet optimaal functionerend omgevingsbewust systeem, fatale gevolgen zou kunnen hebben.

Onderstaande matrix toont het mogelijke verband tussen de in hoofdstuk drie besproken risico's van omgevingsbewuste systemen en de impact die deze risico's kunnen hebben op één of meerdere van de top drie gestelde eigenschappen van omgevingsbewuste systemen.

Risicos / Eigenschappen	Prestatie	Veiligheid	Aanpasbaarheid
Technisch Falen			
Kennisgebrek			
Communicatieissues			
Veiligheid/Privacy			
Betrouwbaarheid			
Systeem intelligentie			

De uit de in literatuur van hoofdstuk twee naar voren gekomen definities van de drie “belangrijkste” eigenschappen van omgevingsbewuste systemen:

Aanpasbaarheid:

De mate waarin een systeem zich kan aanpassen aan gebruikers, omgeving of andere systemen.

Veiligheid:

De mate van veiligheid van een systeem. Veiligheid, zowel authenticatie alsmede autorisatie zijn bij omgevingsbewuste systemen van groot belang. Een omgevingsbewust systeem neemt zelf bepaalde beslissingen die mogelijk een risico kunnen vormen op het gebied van veiligheid.

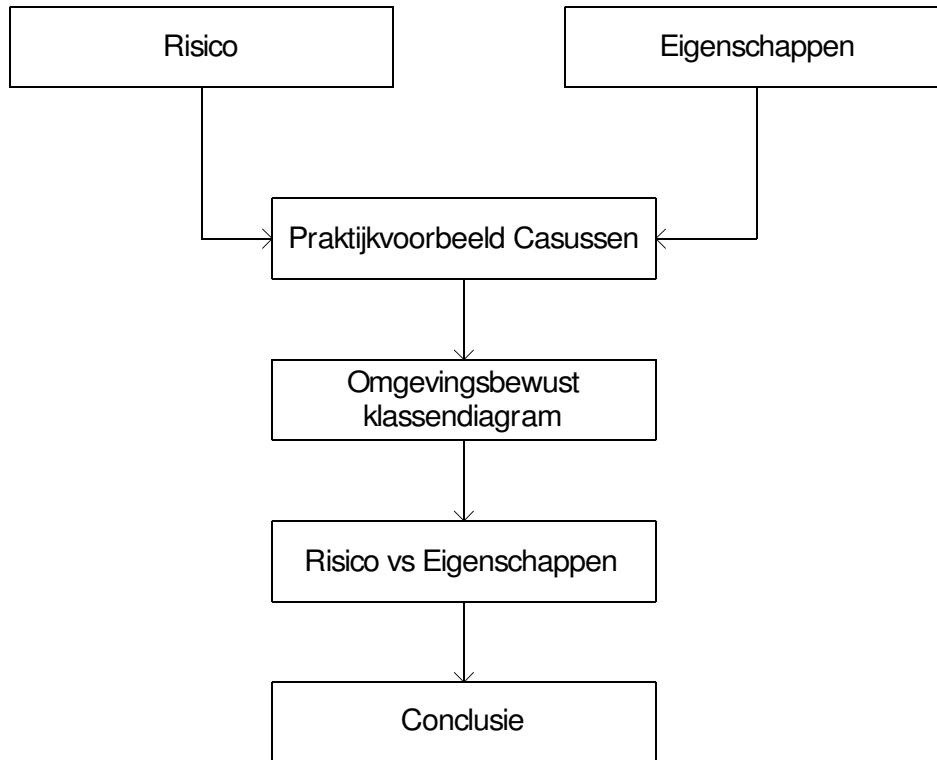
Prestatie:

De mate van presteren van een bepaald systeem. Hierbij kunnen een aantal maatstaven gehanteerd worden. Snelheid en reactietijd zijn twee van deze belangrijke maatstaven.

5.2.1 Methode

In beide uitgewerkte casussen zal per relatie tussen risico en eigenschap gekeken worden aan de hand van bovenstaande matrix, omgevingsbewust klassendiagram en casus of deze risico's aan de orde zijn en in welke mate en op welke manier deze impact hebben op de betreffende systeemeigenschap.

Deze toepassingsmethode kan als volgt worden gemodelleerd (figuur 18):



Figuur 18

Na deze methode te hebben toegepast zal er een conclusie gevormd worden over omgevingsbewuste systemen en risico's waarmee de hoofdvraag van deze thesis beantwoord zal worden:

Wat is een omgevingsbewust systeem, welke risico's brengt een dergelijk systeem met zich mee en hoe zijn deze risico's te beperken?

5.3 Gezondheidszorg

Binnen de moderne gezondheidszorg worden omgevingsbewuste systemen gebruikt. Binnen de gezondheidszorg bijvoorbeeld om zo de vitale lichaamsfuncties van patiënten te kunnen monitoren. In bejaardenhuizen kampt men te kort aan personeel waardoor niet iedereen de zorg krijgt die hij of zij behoeft. Door de woningen uit te rusten met sensoren waarmee hartslag en bloeddruk van een bewoner gemeten kunnen worden, wordt dit uit handen genomen van het reguliere verplegend personeel. Ook staat er zo een betere en constanter beeld van de gezondheid van de patiënt. Er wordt immers op frequentere basis gemeten, dit zou zelfs 24 uur per dag gedaan kunnen worden. Door de uitgebreidere resultaten ontstaat een beter beeld waardoor doktoren en specialisten medicatie beter kunnen afstemmen. Meer zorg op maat is dan het gevolg.

Uiteraard kunnen dergelijke omgevingsbewuste systemen ook een rol van belang spelen bij het waarschuwen van het personeel wanneer bepaalde lichaamsfuncties dermate afwijken dat ingrijpen noodzakelijk is. Denk hierbij aan een te hoge of te lage bloeddruk of in het uiterste geval een hartstilstand. Ook bij dit omgevingsbewuste systeem is het van belang dat de sensoren de informatie uit de omgeving op goede wijze registreren en het achterliggende omgevingsbewuste systeem deze informatie op juiste wijze benut en doorvoert. Een fout tijdens de uitvoering hiervan kan fataal zijn.

5.3.1 Usecases

Volgens [14] is het van belang om bij ieder systeem de actoren vast te stellen. Dit zijn de rollen waarin de gebruikers van het systeem zich voordoen wanneer ze gebruik maken van het systeem. Deze actoren worden weergegeven in zogenaamde use cases om zo naast de actoren ook de functionaliteit van een bepaald systeem vast te leggen. Door middel van de usecases kan zo overzichtelijk worden weergegeven welke actoren welke functionaliteit tot hun beschikking dienen te hebben.

Een use case template ziet er als volgt uit:

Naam	Naam van de usecase
Aannamen	Welke aannamen zijn van kracht alvorens de handeling wordt uitgevoerd
Beschrijving	Beschrijving van de handeling
Uitzonderingen	Uitzonderingen
Resultaat	Resultaat van de handeling

De vraag omtrent een actor in de gezondheidszorg zoals beschreven in 5.1.1 is of het systeem of de gebruiker c.q. patiënt de actor is. Wanneer het omgevingsbewuste systeem de vitale levensfuncties van de gebruiker monitort middels sensoren en hier een afwijking in opmerkt, is dit niet echt een actie van de gebruiker te noemen. Wel ligt, bewust of onbewust, het initiatief bij de patient. Daar gaat immers wat mis bij het een of het ander.

Derhalve kunnen de volgende use cases worden opgesteld:

Patiënt heeft een te hoge bloeddruk:

Naam	Patiënt heeft een te hoge bloeddruk
Aannamen	Systeem is gestart. Patient bevindt zich in gemonitorde ruimte.
Beschrijving	Het systeem is opgestart. De patient bevindt zich in de ruimte waar het omgevingsbewuste systeem een aantal levensfuncties monitort. Wanneer de bloeddruk boven waarde X komt, signaleert het omgevingsbewuste systeem dit en geeft deze informatie door.
Uitzonderingen	De bloeddruk waarde X is hoger afgesteld, bijvoorbeeld vanwege bepaalde medicatie en bijbehorende hogere bloeddruk.
Resultaat	Verantwoordelijke persoon of systeem is op de hoogte van de hogere bloeddruk van de patient.

Patiënt heeft een te lage bloeddruk:

Naam	Patiënt heeft een te lage bloeddruk
Aannamen	Systeem is gestart. Patient bevindt zich in gemonitorde ruimte.
Beschrijving	Het systeem is opgestart. De patient bevindt zich in de ruimte waar het omgevingsbewuste systeem een aantal levensfuncties monitort. Wanneer de bloeddruk onder waarde X komt, signaleert het omgevingsbewuste systeem dit en geeft deze informatie door.
Uitzonderingen	De bloeddruk waarde X is lager afgesteld, bijvoorbeeld vanwege bepaalde medicatie en bijbehorende hogere bloeddruk.
Resultaat	Verantwoordelijke persoon of systeem is op de hoogte van de lagere bloeddruk van de patient.

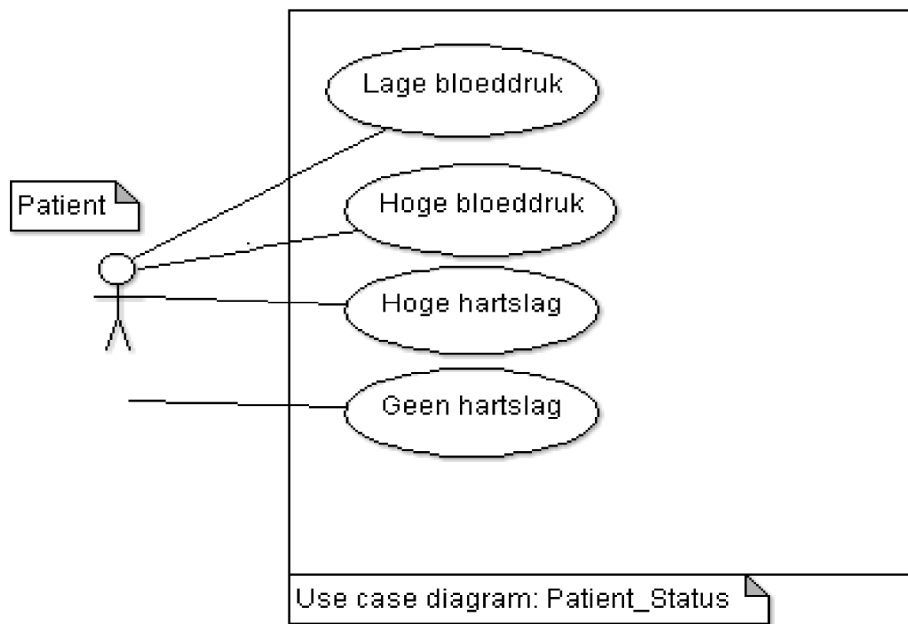
Patiënt heeft een te hoge hartslag:

Naam	Patiënt heeft een te hoge hartslag
Aannamen	Systeem is gestart. Patient bevindt zich in gemonitorde ruimte.
Beschrijving	Het systeem is opgestart. De patient bevindt zich in de ruimte waar het omgevingsbewuste systeem een aantal levensfuncties monitort. Wanneer de hartslag boven waarde X komt, signaleert het omgevingsbewuste systeem dit en geeft deze informatie door.
Uitzonderingen	De hartslag waarde X is hoger afgesteld, bijvoorbeeld vanwege bepaalde medicatie en bijbehorende hogere hartslag.
Resultaat	Verantwoordelijke persoon of systeem is op de hoogte van de hogere hartslag van de patient.

Patiënt heeft geen hartslag:

Naam	Patiënt heeft geen hartslag
Aannamen	Systeem is gestart. Patient bevindt zich in gemonitorde ruimte.
Beschrijving	Het systeem is opgestart. De patient bevindt zich in de ruimte waar het omgevingsbewuste systeem een aantal levensfuncties monitort. Wanneer de hartslag waarde X op 0 komt, signaleert het omgevingsbewuste systeem dit en geeft deze informatie door.
Uitzonderingen	-
Resultaat	Verantwoordelijke persoon of systeem is op de hoogte van de ontbrekende hartslag van de patient.

Dit zijn voorbeelden van situaties die kunnen ontstaan en waar het omgevingsbewuste systeem mee dient om te gaan. Deze use cases kunnen grafisch worden weergegeven in onderstaand usecase diagram (Figuur 19).

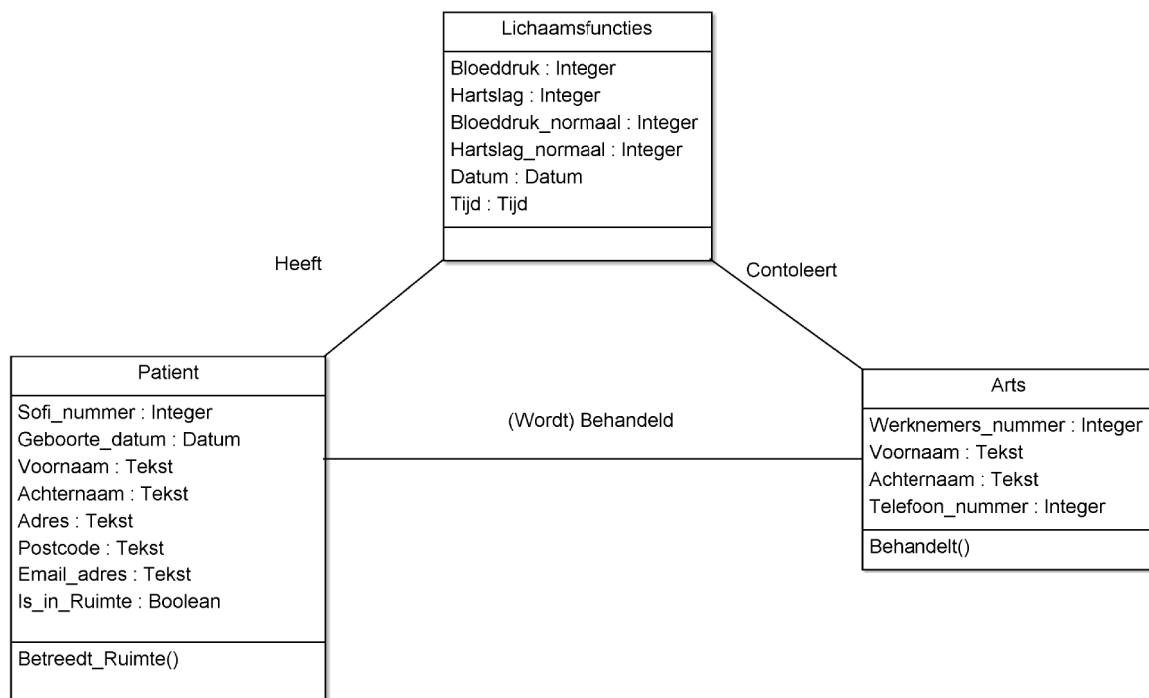


Figuur 19

5.3.2 Klassendiagram / Objectdiagram

In een klassendiagram wordt, zoals beschreven in 4.2.1, vastgelegd uit welke klassen een systeem bestaat en wat de onderlinge relatie is. De klassen bevatten attributen (eigenschappen van de klassen) en mogelijk ook operaties (handelingen) van een bepaalde klasse. Een voorbeeld van een dergelijk model is toegelicht in 4.2.1.

In deze paragraaf zal een klassendiagram worden uitgewerkt dat is gebaseerd op de beschreven usecases en het hieruit voortgekomen usecasediagram.

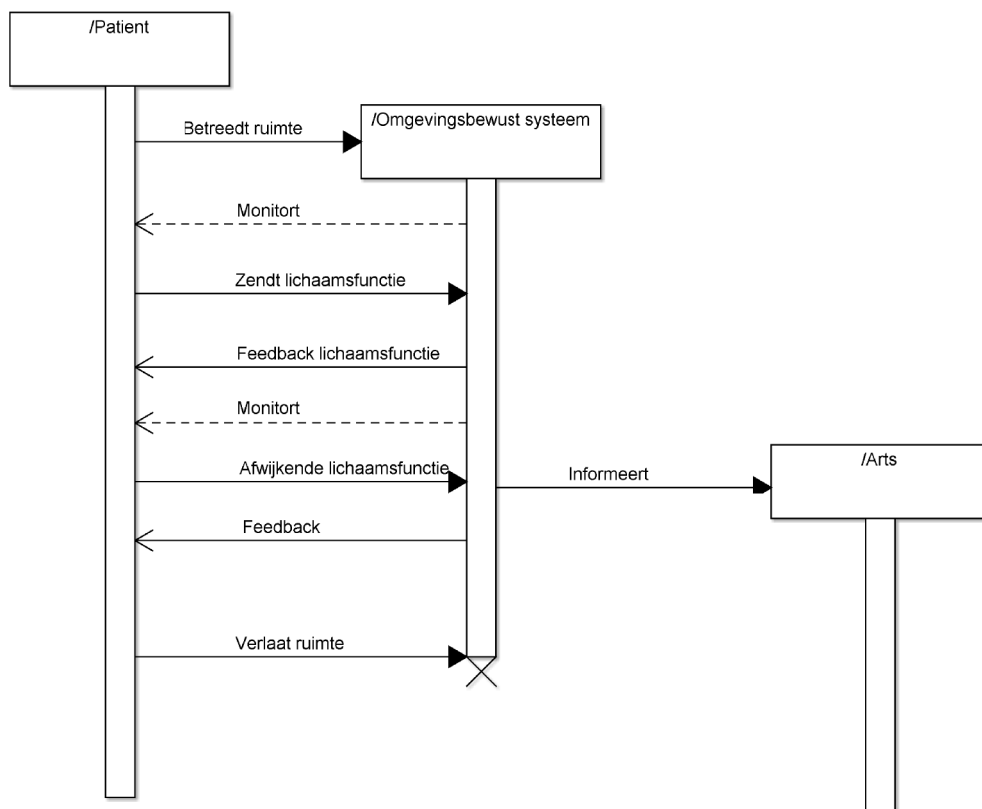


Figuur 20

In bovenstaande afbeelding (figuur 20) worden drie klassen afgebeeld. De klasse patiënt, lichaamsfuncties en de klasse arts. De patiënt heeft een aantal eigenschappen (attributen), hetzelfde geldt voor de arts. De lichaamsfuncties behoren bij de patiënt, welke door de arts weer gecontroleerd worden. Omdat er gedurende langere tijd gemonitord dient te worden wat de lichaamsfuncties van een patiënt zijn en hierbij dus de geschiedenis en een eventuele stijging, daling of andere vorm van trend, waargenomen kan worden, worden deze gegevens in een aparte klasse vermeld.

5.3.3 Sequencediagram

Zoals toegelicht in 4.2.1 geeft een sequence diagram inzicht in met name het tijdsaspect van handelingen en volgorde van handelingen van een bepaald systeem. In dit geval een omgevingsbewust systeem. Logischerwijs kan onderstaand model gebruikt worden voor zowwel de waarde bloedruk alsmede de waarde hartslag.



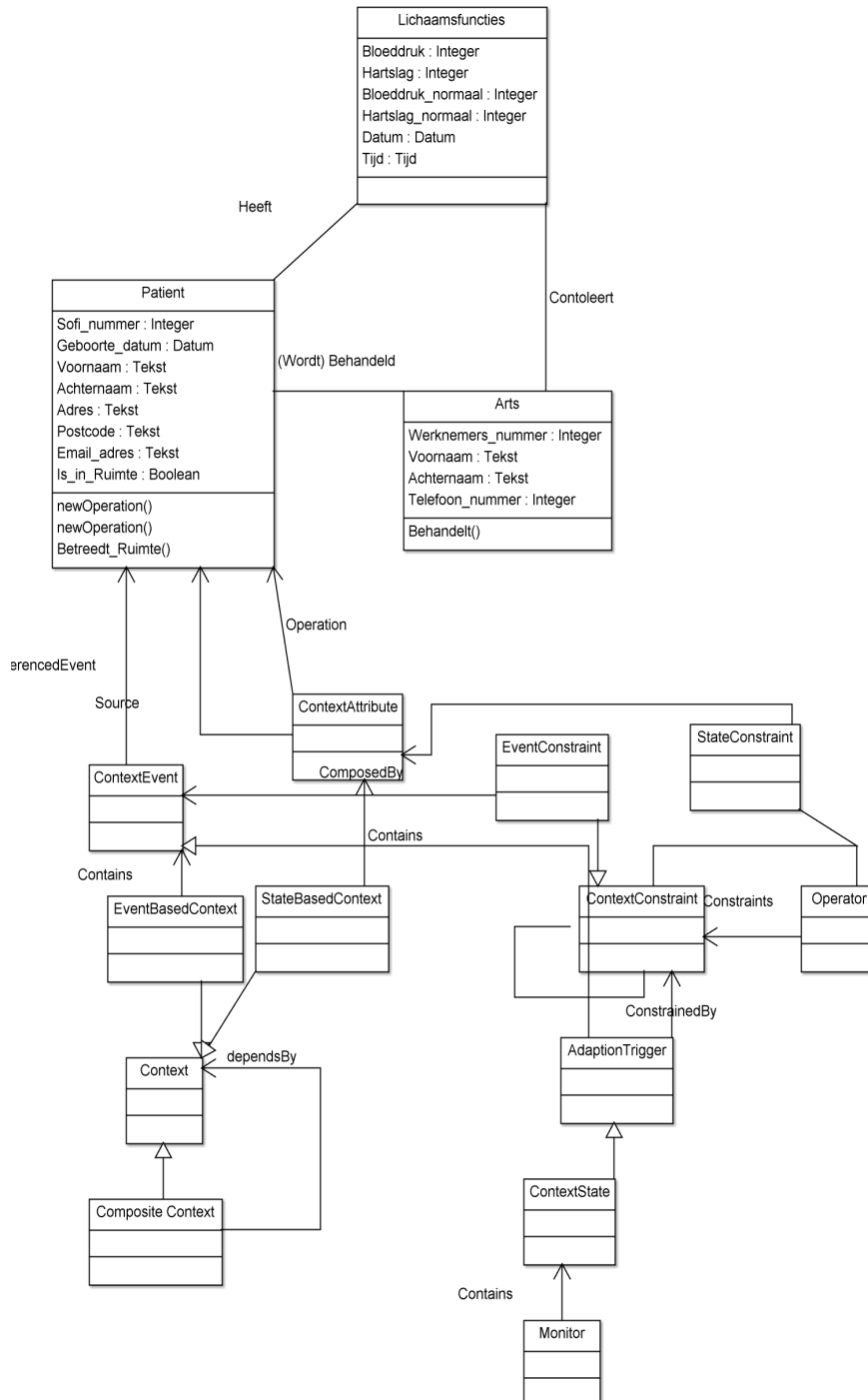
Figuur 21

In bovenstaand sequence diagram (figuur 21) is te zien in welke volgorde de acties worden uitgevoerd. Het betreden van de ruimte door de patiënt initieert het monitoren van de lichaamsfuncties van de patiënt. Deze zendt middels sensoren de lichaamsfuncties naar de ontvangstsensoren van het omgevingsbewuste systeem. Mocht dit gewenst zijn door de patiënt dan kan er door het systeem feedback worden gegeven aan de patiënt over de waarde van de lichaamsfunctie. . Zodra er een afwijkende lichaamswaarde wordt opgemerkt, kan een arts direct geïnformeerd worden. Wanneer de patiënt de ruimte verlaat, stopt het monitoren

5.3.4 Omgevingsbewuste klassendiagrammen

5.3.4.1 Monitoren van de omgeving

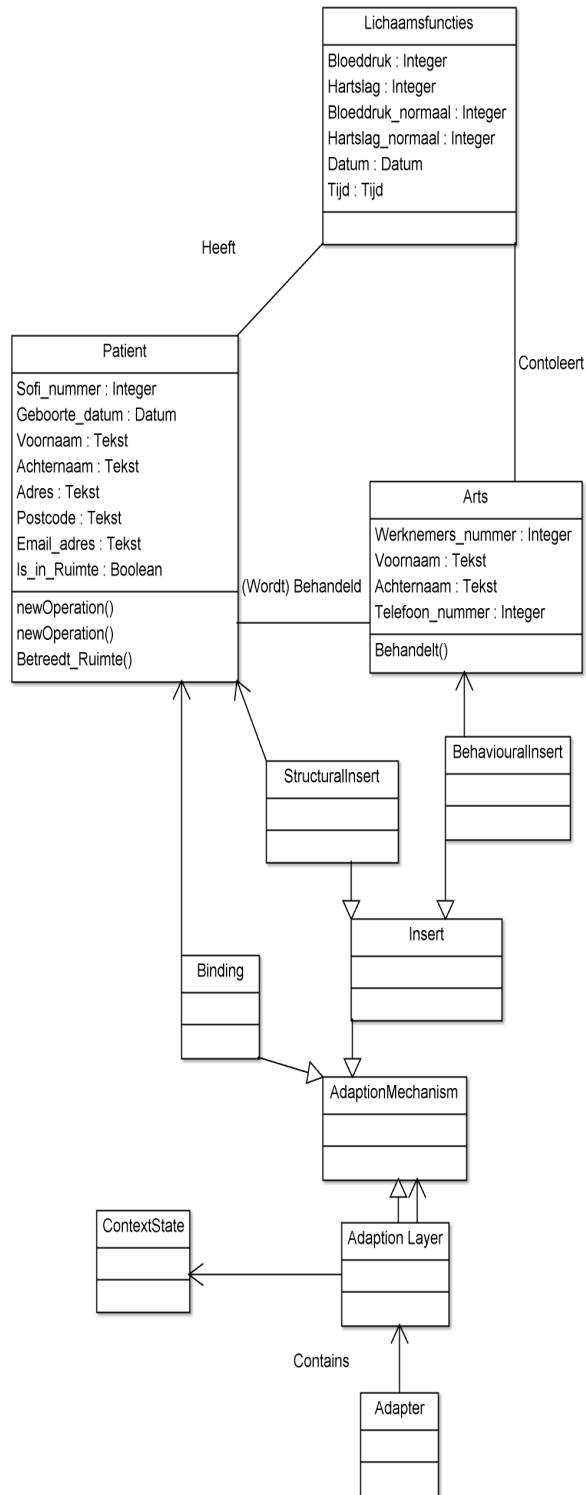
Onderstaand klassendiagram betreft het oorspronkelijke klassendiagram van de moderne gezondheidszorg met hierop toegepast het metamodel “monitoren van de omgeving” van CAMEL.



Figuur 22

5.3.4.2 Reageren op de omgeving

Onderstaand klassendiagram betreft het oorspronkelijke klassendiagram van de moderne gezondheidszorg met hierop toegepast het meta-model “reageren op de omgeving” van CAMEL.



Figuur 23

5.3.5 Risico's versus eigenschappen

De uit de in literatuur van hoofdstuk twee naar voren gekomen definities van de drie “belangrijkste” eigenschappen van omgevingsbewuste systemen:

Aanpasbaarheid:

De mate waarin een systeem zich kan aanpassen aan gebruikers, omgeving of andere systemen.

Veiligheid:

De mate van veiligheid van een systeem. Veiligheid, zowel authenticatie alsmede autorisatie zijn bij omgevingsbewuste systemen van groot belang. Een omgevingsbewust systeem neemt zelf bepaalde beslissingen die mogelijk een risico kunnen vormen op het gebied van veiligheid.

Prestatie:

De mate van presteren van een bepaald systeem. Hierbij kunnen een aantal maatstaven gehanteerd worden. Snelheid en reactietijd zijn twee van deze belangrijke maatstaven.

Per risico zal aangegeven worden wat de relatie is tussen het risico en de bijbehorende eigenschap van het omgevingsbewuste systeem, dit zal gerelateerd worden aan de casus van de moderne gezondheidszorg.

Technisch Falen:

Technisch falen is funest voor ieder systeem, omgevingsbewust of niet. Daar een omgevingsbewust de informatie niet direct van een gebruiker ingevoerd krijgt maar deze verkrijgt uit de omgeving zal een omgevingsbewust systeem hier ook meer techniek voor gebruiken. De kans op een defect is dan logischerwijs ook hoger daar er meer techniek benodigd is. Mocht er een technische defect optreden dan heeft dit impact op de aanpasbaarheid en prestatie van het systeem. Wanneer een aanpassing niet kan worden uitgevoerd heeft dit ook zijn weerslag op de veiligheid van de patiënt. Ook wanneer het omgevingsbewuste systeem onder de maat presteert en de vitale lichaamswaardes niet tijdig worden doorgegeven kan dit de veiligheid van de patiënt in gevaar brengen.

Technisch falen heeft impact op de Aanpasbaarheid, veiligheid en prestatie van een omgevingsbewust systeem.

Kennisgebrek:

Kennisgebrek is in hoofdstuk drie gedefinieerd als kennisgebrek bij de gebruiker van het systeem. Uiteraard dient een omgevingsbewust niet “misbruikt” te worden, dit is bij ieder systeem aan de orde. Een omgevingsbewust systeem wordt niet specifiek gebruikt de omgeving zelf, het systeem haalt in een constante stroom informatie op uit de omgeving. De gebruiker, mits de omgeving als gebruiker is gedefinieerd, heeft wel invloed op de informatie die aan het systeem wordt doorgegeven. Bij deze casus van moderne gezondheidszorg is dit helemaal het geval. Sterker nog, het zijn juist de vitale lichaamswaardes van de gebruiker die centraal staan in deze casus. De patiënt kan bewust of onbewust, of door medisch falen of bijvoorbeeld door inspanning, de gegevens die het omgevingsbewuste systeem vergaart van deze patiënt, beïnvloeden. Het is echter aan het omgevingsbewuste systeem wat er met deze informatie gedaan wordt en welke afwegingen er gemaakt worden op basis van deze informatie.

In CAMEL is hier gedeeltelijk rekening mee gehouden in de vorm van “Behavioural” of “Structural” insert..

Er mag worden aangenomen dat de patiënt tot op zekere hoogte kennis heeft van het doel van het omgevingsbewuste systeem. Wel moet bij het ontwerp van het omgevingsbewuste systeem rekening worden gehouden met het feit dat een patiënt of gebruiker afwijkend gedrag kan vertonen waar het systeem dan toch mee om zal moeten gaan.

Kennisgebrek heeft dan ook in geringe mate impact op de aanpasbaarheid, veiligheid en aanpasbaarheid van een omgevingsbewust systeem. Mits het systeem vooraf ook is ingericht om onverwacht gedrag van de omgeving te kunnen afvangen en verwerken.

Communicatie issues:

Communicatie is cruciaal bij omgevingsbewuste systemen. Communicatie betreft de kunst van het overdragen van informatie. Deze informatie overdracht zal van A tot Z op goede wijze overgebracht dienen te worden. Wanneer een patiënt een hartaanval krijgt, het omgevingsbewuste systeem dit ook als hartaanval markeert maar dit nooit bij de verantwoordelijke arts terecht komt dan is het hele voortraject overbodig geweest. Het omgevingsbewuste systeem is dan ook zo sterk als de zwakste (communicatie) schakel van het systeem.

Wanneer de communicatie niet goed verloopt, het omgevingsbewust systeem niet de juiste of de volledige waardes van lichaamsfuncties van de patiënt doorkrijgt dan kan het systeem niet de juiste afweging maken en bestaat het risico dat er een verkeerde afweging wordt gemaakt.

Dit heeft invloed op de aanpasbaarheid van het systeem. Hierdoor de veiligheid van de ook veiligheid van de patiënt in het geding komen. Het niet tijdig hebben van informatie of het hebben van onvolledige informatie vanwege een communicatie storing zal resulteren in slechtere prestaties van het systeem.

Veiligheid:

Veiligheid als risico versus veiligheid als eigenschap van een omgevingsbewust systeem. Het in het gedrang komen of van of het gebrek aan veiligheid is dan ook een betere definitie van veiligheid als risico. In de gestelde casus van de moderne gezondheidszorg zorgt het niet juist kunnen aanpassen of reageren op de omgeving voor het in gedrang komen van de veiligheid van de patiënt. Dat door gebrekkige prestaties de veiligheid ook in gedrang komt blijkt uit het feit dat prestatie ook weer invloed heeft op de aanpasbaarheid van een systeem. Het niet volledig presteren van een omgevingsbewust systeem is op zich al een risico daar het in de casus een mensenleven betreft.

Privacy:

Met privacy in de context van omgevingsbewuste systemen wordt bedoeld of en in welke mate er zorgvuldig wordt omgegaan met de grote hoeveelheid gegevens die een omgevingsbewust systeem te verwerken krijgt. Blijven zulke gegevens lang bewaard, hoe worden deze bewaard en wie heeft toegang tot deze gegevens? Helemaal bij medische gegevens en patiëntinformatie liggen deze vraagstukken extra gevoelig. Privacy en veiligheid zijn derhalve ook onlosmakelijk verbonden. Gebrek aan veiligheid heeft mogelijk impact op de veiligheid. Geen rekening houden met privacy laat een gebrek aan veiligheid zien.

Het bewaken van de privacy benodigd een bepaalde mate van veiligheid. Dit dient niet ten koste te gaan van de prestaties en aanpasbaarheid van het systeem. Hier zal een balans in gevonden dienen te worden.

Betrouwbaarheid:

Betrouwbaarheid is in het geval van een omgevingsbewust gezondheidszorg systeem van groot belang. Wanneer het systeem niet betrouwbaar is en niet of niet volledig werkt dan heeft dit impact op de veiligheid van de patiënt. Niet alleen of een systeem betrouwbaar is maar ook de mate van betrouwbaarheid is van belang. Een omgevingsbewust systeem dat een klein beetje betrouwbaar is, is niet betrouwbaar. Een systeem dient volledig betrouwbaar te zijn.

Betrouwbaarheid of het gebrek hieraan, heeft dan ook impact om de prestaties en veiligheid van het systeem. Wanneer de betrouwbaarheid te wensen overlaat zal ook het vermogen om aan te passen afnemen.

Systeem intelligentie:

Als risico is dit het gebrek aan systeem intelligentie. Het omgevingsbewuste systeem zal een zekere mate van beslissingsbevoegdheid moeten hebben om tijdig te kunnen reageren of te kunnen handelen. Dit komt de prestaties van een omgevingsbewust systeem ten goede. Anderzijds moet ook de veiligheid van de patiënt gewaarborgd blijven, het is niet gewenst dat het systeem rare beslissingen gaat maken of de verkregen informatie anders gaat weergeven of doorgeven dan gewenst is. Binnen CAMEL kan dit afgevangen worden door duidelijke voorwaarden in de vorm van constraints in te bouwen. Er zal op basis van de constraints bepaald worden of er wel of geen melding wordt gemaakt aan de betreffende arts vanwege afwijkende lichaamsfunctie waardes.

Anderzijds is het gedrag van een patiënt van belang. Het is aan het omgevingsbewust systeem om te bepalen of gedrag incidenteel of structureel is. Wanneer gedrag als structureel wordt gekenmerkt kan hier bij terugkerende structureel gedrag, sneller op gereageerd worden. Het op een vaste dag op een gezette tijd een ruimte verlaten of betreden kan als structureel gedrag worden gekenmerkt. De vraag is echter wie de scheidinglijn bepaald tussen incidenteel en structureel gedrag. Hiermee zal tijdens het ontwikkelen van het systeem rekening gehouden dienen te worden. CAMEL geeft dit weer in het meta model door te werken met Structural en Behavioural inserts.

Wanneer de systeem intelligentie niet op niveau is dan heeft dit impact op de veiligheid van de patiënt. Ook de aanpasbaarheid komt zo in het gedrag. Wanneer een omgevingsbewust systeem te lang doet over het maken van een bepaalde beslissing of het geven van bepaalde feedback dan is dit ook niet bevorderlijk voor de prestaties.

Matrix Risico's versus Eigenschappen omgevingsbewust systeem moderne gezondheidszorg: toont de relatie tussen de betreffende systeem eigenschappen en in hoofdstuk drie besproken risico's.

Risico's / Eigenschappen	Prestatie	Veiligheid	Aanpasbaarheid
Technisch Falen	x	x	x
Kennisgebrek			
Communicatie issues	x	x	x
Veiligheid	x	x	x
Privacy	x	x	x
Betrouwbaarheid	x	x	x
Systeem intelligentie	x	x	x

5.4 Auto industrie

Omgevingsbewuste technieken worden tegenwoordig gebruikt in diverse soorten systemen. Niet alleen in systemen binnen een bedrijfsomgeving maar ook in computers die gebruikt worden in bijvoorbeeld auto's of binnen de moderne gezondheidszorg.

In de moderne auto van tegenwoordig zitten allerlei sensoren verwerkt die in direct contact staan met de boordcomputer van een auto. Denk bijvoorbeeld aan parkeersensoren die de bestuurder informatie geven over objecten rondom een automobiel. Denk ook aan sensoren die aangeven of een veiligheids gordel wel of niet wordt gebruikt, sensoren die automatisch de verlichting in of uitschakelen op basis van (gebrek aan) daglicht of sensoren die ervoor zorgen dat het voertuig afremt zodra een voertuig ervoor afremt. Een andere veel gebruikte sensor in een voertuig is de sensor die waarneemt wanneer er een botsing plaatsvindt om vervolgens de airbags in werking te stellen.

Zonder deze sensoren is een voertuig een stuk minder veilig dan met deze moderne sensoren die van een voertuig een omgevingsbewust systeem maken. Het is uiteraard wel van belang dat de sensoren juist functioneren zodat de achterliggende boordcomputer op basis van de informatie die wordt doorgegeven een goede beslissing kan maken om wel of niet in te grijpen of een bepaalde handeling uit te voeren. Een degelijke ontwikkeling, testfase en implementatie van een dergelijk systeem is dan ook van groot belang.

Zoals aangegeven wordt in de meeste auto's ook gebruik gemaakt van één of meerdere omgevingsbewuste systemen. Middels sensoren wordt de nodige informatie vergaard. Deze informatie wordt beoordeeld door het omgevingsbewuste systeem. Vervolgens wordt feedback gegeven aan de bestuurder of wordt direct actie ondernomen. Een aantal voorbeelden in use case vorm.

5.3.1 Usecases

Bestuurder draagt geen gordel:

Naam	Bestuurder draagt geen gordel
Aannamen	De auto (en dus het systeem) is gestart. De bestuurder bevindt zich op de bestuurderstoel voor een minimale periode van 30 seconden of langer.
Beschrijving	De bestuurder bevindt zich op de bestuurdersstoel en draagt zijn gordel niet. Gordel maakt geen contact met de gordelhouder.
Uitzonderingen	-
Resultaat	Er wordt een oplopende pieptoon gegenereerd.

Passagier draagt geen gordel:

Naam	Passagier draagt geen gordel
Aannamen	De auto (en dus het systeem) is gestart. De passagier bevindt zich op de passagiersstoel voor een minimale periode van 30 seconden of langer.
Beschrijving	De passagier bevindt zich op de passagiersstoel en draagt zijn gordel niet. Gordel maakt geen contact met de gordelhouder.
Uitzonderingen	-
Resultaat	Er wordt een oplopende pieptoon gegenereerd.

Verlichting:

Naam	Verlichting
Aannamen	Er is minder dan X% licht. De auto (en dus het systeem) is gestart.
Beschrijving	Indien de auto gestart is er er minder dan X% licht wordt opgevangen, schakelt het systeem de verlichting in.
Uitzonderingen	Verlichting wordt alsnog handmatig uitgeschakeld.
Resultaat	De verlichting is ingeschakeld.

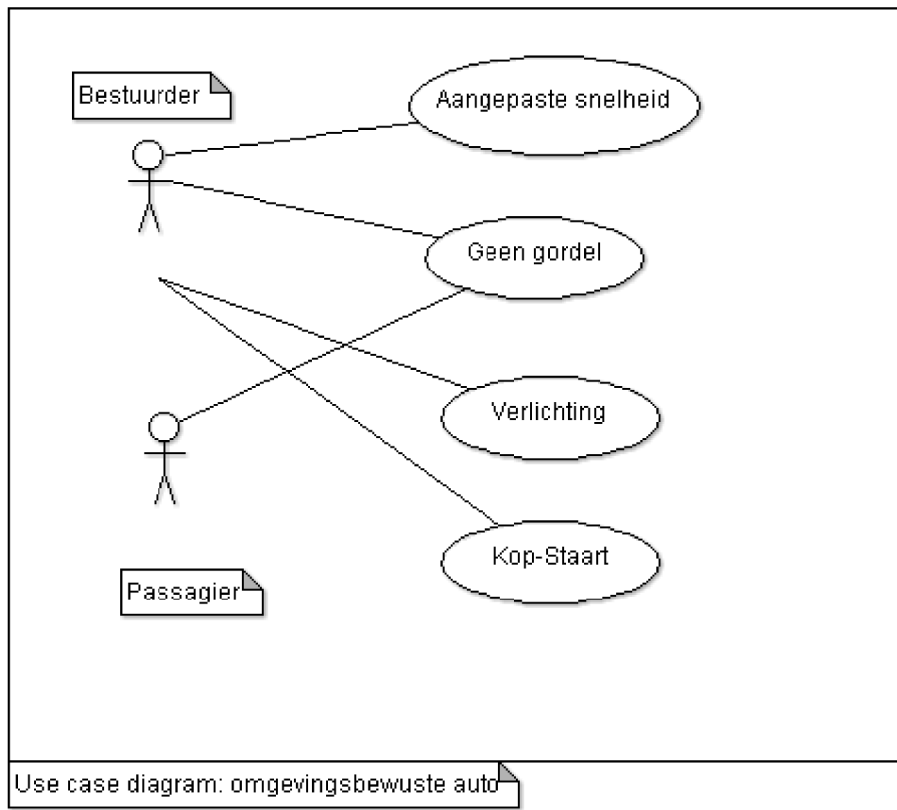
Voorligger past snelheid aan:

Naam	Voorligger past snelheid aan
Aannamen	De auto (en dus het systeem) is gestart. De auto rijdt tussen de X en Y km per uur. De automatische snelheidsaanpas functie is ingeschakeld. De maximale snelheid wordt niet overschreven.
Beschrijving	De voorligger van de auto mindert vaart. De auto mindert evenredig zijn snelheid om zo genoeg afstand te houden. Indien de vaart toeneemt en de maximale snelheid niet overschreven wordt, neemt de vaart van de auto automatisch toe.
Uitzonderingen	De automatische snelheidsaanpassingsfunctie is uitgeschakeld. De auto rijdt niet met een snelheid tussen X en Y.
Resultaat	De snelheid van de auto is aangepast aan de snelheid van de voorliggende auto.

Airbag bij kop-staart botsing:

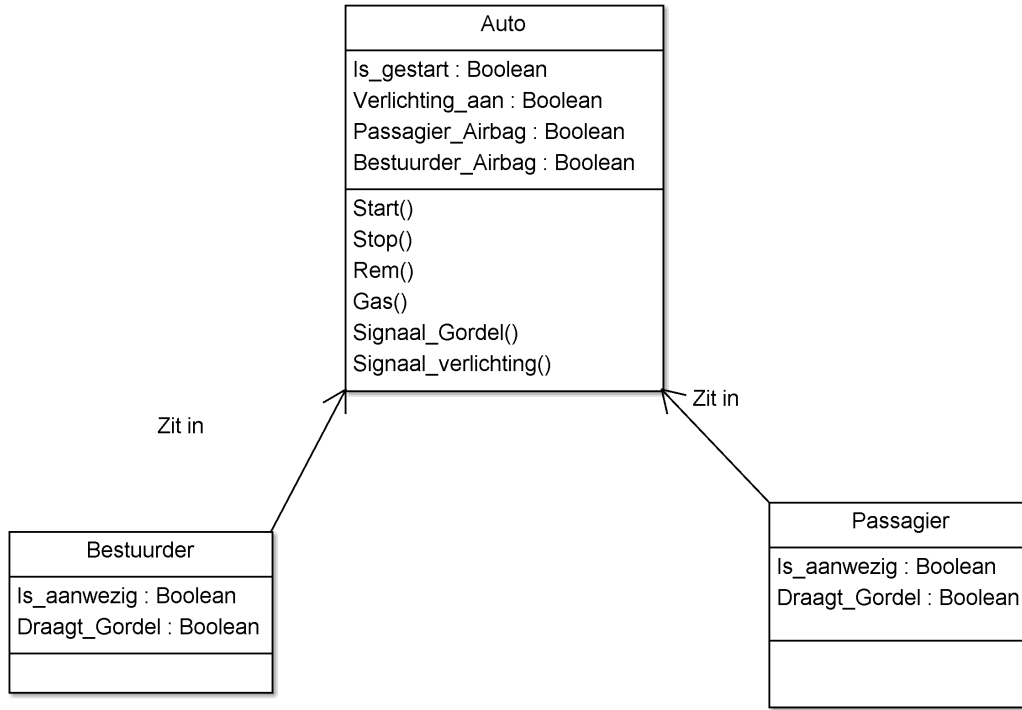
Naam	Airbag bij kop-staart botsing
Aannamen	De bestuurder bevindt zich op de bestuurdersstoel
Beschrijving	Zodra de auto frontaal of van achteren wordt geraakt en de bijbehorende kreukzones niet langer in tact zijn, dienen de airbags aan bestuurderskant opgeblazen te worden.
Uitzonderingen	Airbags zijn van tevoren uitgeschakeld.
Resultaat	Airbags zijn ontplooid.

In een usecase diagram ziet dit er als volgt uit. Er zijn twee actoren, namelijk de bestuurder en de passagier.



Figuur 24

5.4.2 Klassendiagram / Objectdiagram



Figuur 25

Op basis van de usecases zoals gesteld in 5.3.1 kan onderstaand klassendiagram opgesteld worden. Het gaat hier om een drietal klassen, Auto, bestuurder en passagier. Het argument dat de bestuurder van de auto ook passagier is, echter in een andere rol (namelijk de bestuurder van het voertuig) zou aangevoerd kunnen worden. Dit is binnen UML ook te modelleren. In deze specifieke casus is het wel van belang dat de auto onderscheidt maakt tussen de bestuurdersplek en de plek van de passagier. Bij het eventueel ontplooiën van de airbag is het van groot belang dat wel de juiste airbag ontplooid wordt. Er zou eventueel gewerkt kunnen worden met Passagier 1 tm 4 (afhankelijk van de grootte van het voertuig) waarbij dan passagier 1 wordt aangemerkt als de bestuurder. Dan dient wel gespecificeerd te worden welke nummer passagier welke plek inneemt in het voertuig. Deze specificatie wordt hier geabstraheerd, voor het gemak wordt hier gebruik gemaakt van de term bestuurder en passagier.

Voor wat betreft het gedeelte dat gekeken moet een veiligheidsgordel gedragen wordt of niet en of een bestuurder of passagier zich bevindt in de auto kan dit vooralsnog worden afgevangen met bovenstaande klassen en bijbehorende attributen. Hoe dit zal verlopen, zal aan bod komen in de volgende paragraaf waarbij de sequence diagrammen worden uitgewerkt.

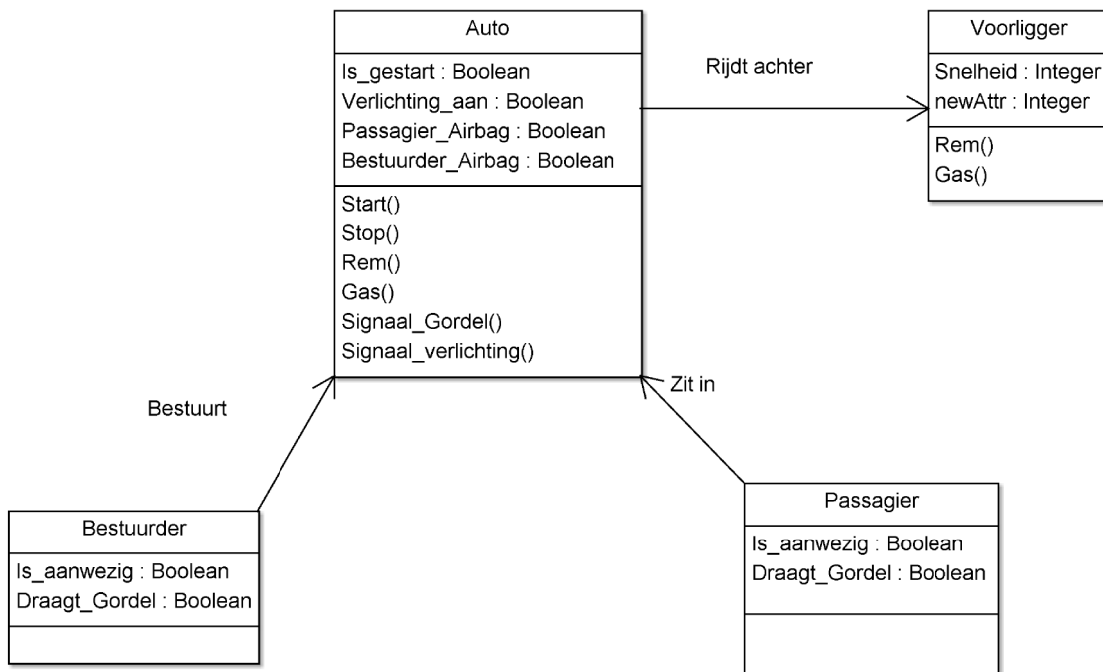
Nu is de vraag op welke manier er met behulp van UML of CAMEL gemodelleerd kan worden dat een dit omgevingsbewuste autosysteem ook rekening dient te houden met een ander voertuig, een voorligger om specifiek te zijn. Op basis van de acties van een voorganger kan het omgevingsbewuste systeem ingrijpen en aanpassingen in de eigen rijstijl uitvoeren door bijvoorbeeld te remmen wanneer de voorligger bovengemiddeld hard remt.

Om te zorgen dat het omgevingsbewuste systeem kan reageren op de handeling van buiten het systeem, zal toch het systeem zich bewust moeten zijn van de mogelijkheid dat een dergelijke situatie zich eventueel zou kunnen voordoen. Het omgevingsbewuste systeem dient dan ook gemodelleerd en ingericht te worden op basis van deze mogelijke situatie.

Wanneer dan ook de auto en het omgevingsbewuste systeem van de auto als een geheel wordt gezien, kan gesteld worden dat een bestuurder en passagier van de auto ook tot de omgeving behoren. Omdat zij als het ware, deel uit maken van hetzelfde voertuig ligt deze manier van benaderen wellicht niet voor de hand.

Wanneer we deze wijze van redeneren doorzetten kunnen we ook het voorliggende voertuig meenemen tijdens het modelleren van de klassen. Hoewel fysiek gezien deze voorligger tot de omgeving behoort, nemen wij voor het belang van het goed functioneren van het systeem, en uiteindelijk de veiligheid van passagier, bestuurder en andere weggebruikers de voorliggende auto op in het klassendiagram.

Met deze aanpassing komt het klassendiagram er als volgt uit te zien:



Figuur 26

Er kan getwist worden over het feit om meer gegevens van de voorlegger vast te stellen. Om bijvoorbeeld te kijken naar het merk en type auto en of er op basis van eerdere ervaringen met

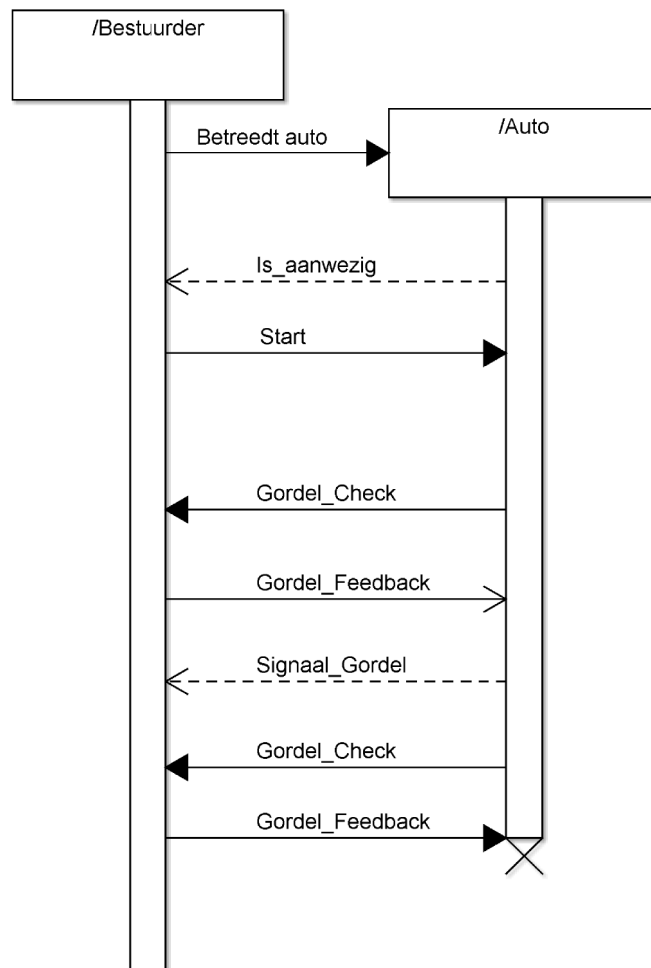
voorliggers van dit merk auto gesteld kan worden dat zij vaak onverwachte reacties uitvoeren. Dit is echter ook weer afhankelijk van de bestuurder van de voorliggende auto en is niet zo zeer merk gebonden.

Wel kan het verstandig zijn om (bijvoorbeeld middels GPS tracking) vast te stellen of een auto zich binnen of buiten de bebouwde kom bevindt of eventueel op een snelweg rijdt om zo een verband te leggen met bepaalde verwachte snelheidswijzigingen. Dit wordt in deze thesis echter niet meegenomen en zal worden geabstraheerd daar het niet te plaatsen valt in het risico of omgevingsbewuste domein.

5.4.3 Sequencediagram

Om het overzicht te handhaven zullen er in deze casus aparte sequence diagrammen worden gemaakt per use-case.

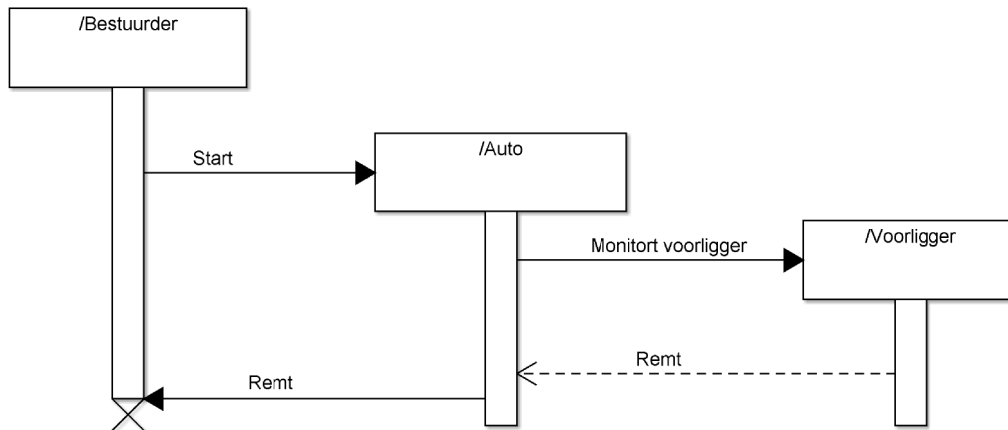
Gordel Check (Bestuurder / Passagier):



Figuur 27

In bovenstaand diagram is te zien dat de bestuurder de auto betreedt, de auto start. Vervolgens controleert de auto of de bestuurder een gordel draagt. Wanneer dit niet het geval is, geeft hij een signaal terug. Dit blijft herhaald worden totdat het omgevingsbewuste systeem vaststelt dat de gordel wordt gedragen. Hetzelfde geldt uiteraard voor de passagier.

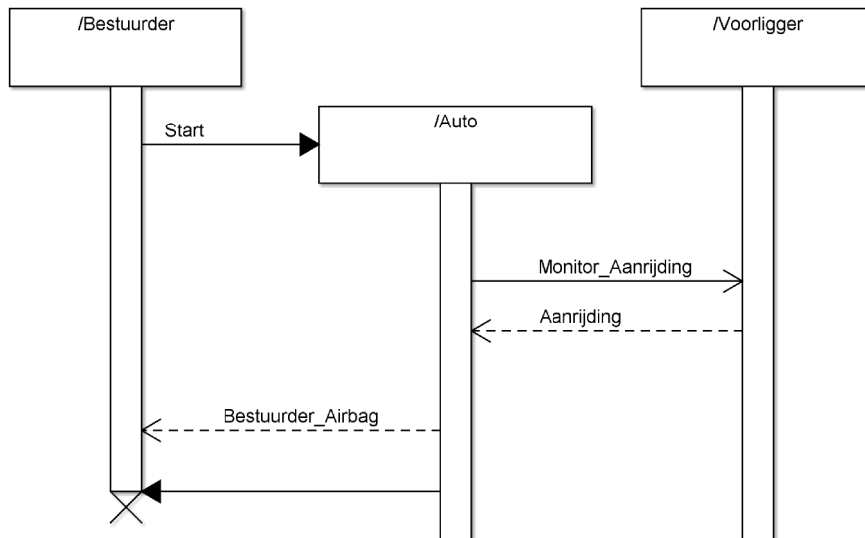
Voorligger remt:



Figuur 28

In dit sequencediagram is te zien hoe de bestuurder de auto start. De auto monitort de voorligger. Wanneer de voorligger remt zal de auto ook remmen. Deze volgorde kan herhaald worden. Het feit dat het uitvoeren van deze acties nog aan allerlei voorwaarden dienen te voldoen (functie moet door bestuurder worden aangezet, voorligger moet een bepaalde snelheid hebben en met een percentage X remmen etc) worden niet weergegeven in het sequence diagram.

Aanrijding / Airbag:



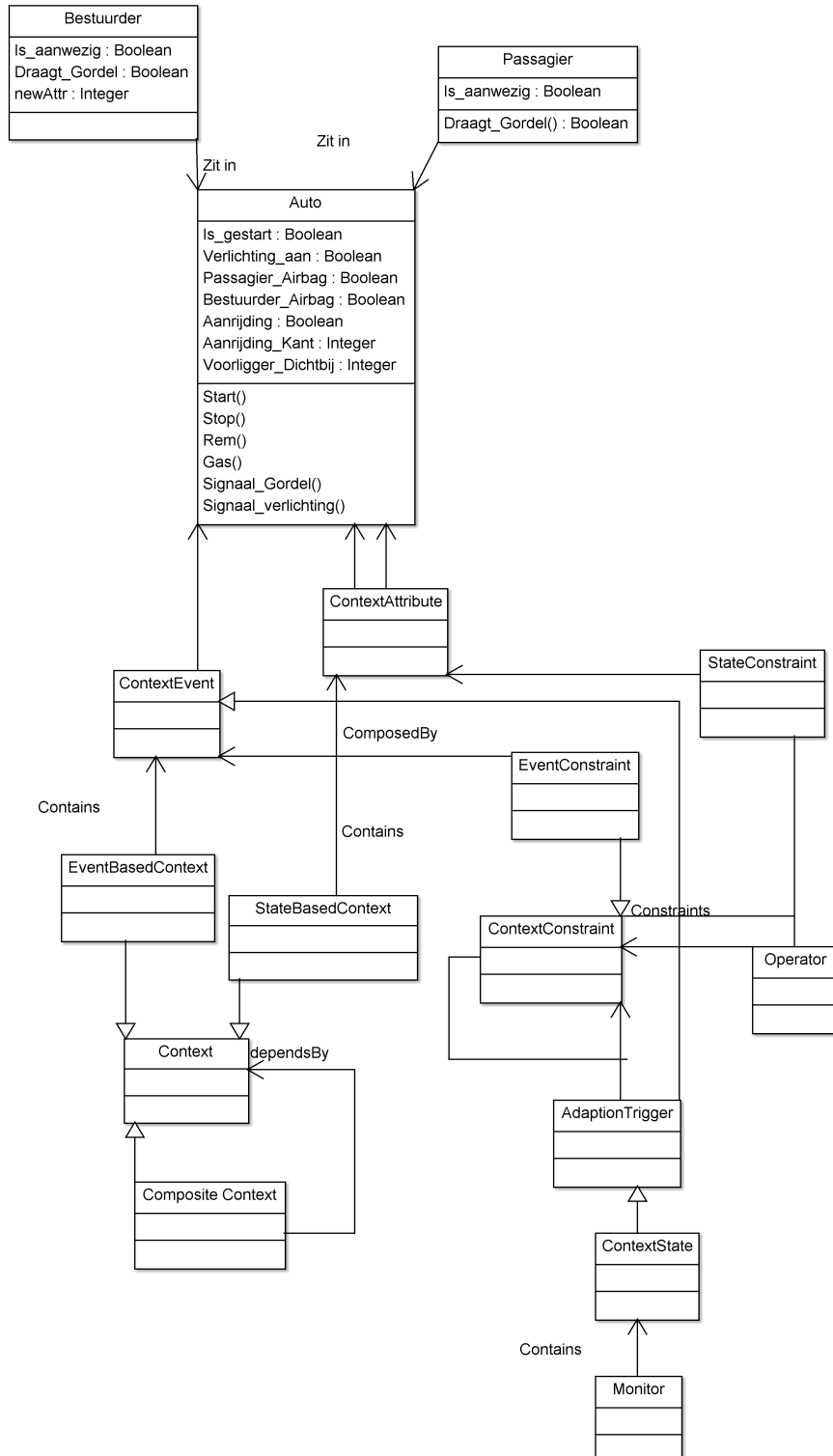
Figuur 29

In dit sequencediagram is te zien hoe de bestuurder de auto start. Het omgevingsbewuste systeem van de auto monitor of er een aanrijding met de voorligger heeft plaatsgevonden. Indien dit het geval is dan zal de airbag van de bestuurder ontplooid worden.

5.4.4 Omgevingsbewuste klassendiagrammen

5.4.4.1 Monitoren van de omgeving

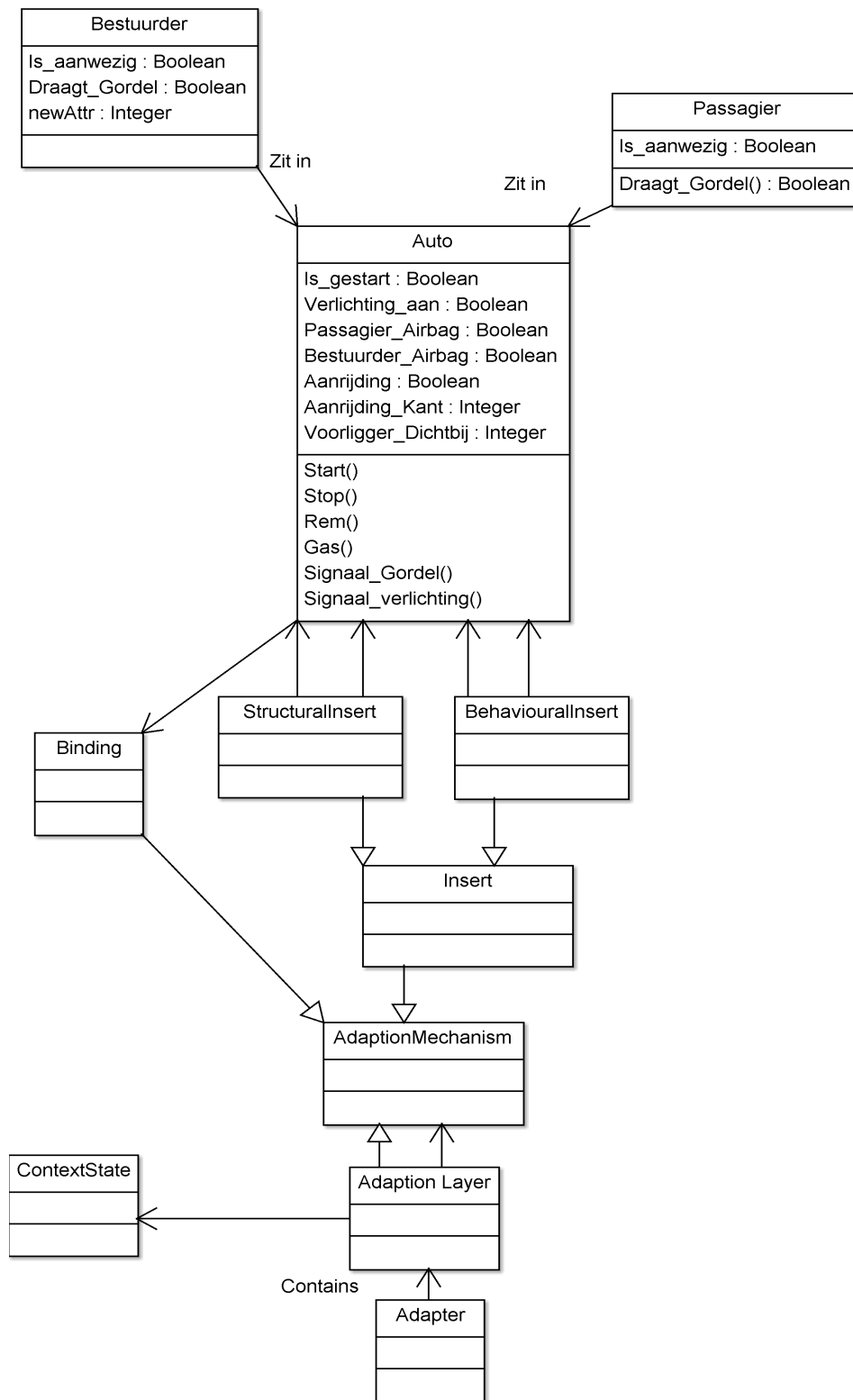
Onderstaand klassendiagram betreft het oorspronkelijke klassendiagram van een omgevingsbewust autosysteem met hierop toegepast het metamodel “monitoren van de omgeving” van CAMEL.



Figuur 30

5.4.4.2 Reageren op de omgeving

Onderstaand klassendiagramm betreft het oorspronkelijke klassendiagramm van een omgevingsbewust autosysteem met hierop toegepast het metamodel “reageren op de omgeving” van CAMEL.



Figuur 31

5.4.5 Risico's versus eigenschappen

De uit de in literatuur van hoofdstuk twee naar voren gekomen definities van de drie “belangrijkste” eigenschappen van omgevingsbewuste systemen:

Aanpasbaarheid:

De mate waarin een systeem zich kan aanpassen aan gebruikers, omgeving of andere systemen.

Veiligheid:

De mate van veiligheid van een systeem. Veiligheid, zowel authenticatie alsmede autorisatie zijn bij omgevingsbewuste systemen van groot belang. Een omgevingsbewust systeem neemt zelf bepaalde beslissingen die mogelijk een risico kunnen vormen op het gebied van veiligheid.

Prestatie:

De mate van presteren van een bepaald systeem. Hierbij kunnen een aantal maatstaven gehanteerd worden. Snelheid en reactietijd zijn twee van deze belangrijke maatstaven.

Per risico zal aangegeven worden wat de relatie is tussen het risico en de bijbehorende eigenschap van het omgevingsbewuste systeem, dit zal gerelateerd worden aan de casus het omgevingsbewuste autosysteem.

Technisch Falen:

Technisch falen is funest voor ieder systeem, omgevingsbewust of niet. Daar een omgevingsbewust de informatie niet direct van een gebruiker ingevoerd krijgt maar deze verkrijgt uit de omgeving zal een omgevingsbewust systeem hier ook meer techniek voor gebruiken. De kans op een defect is dan logischerwijs ook hoger daar er meer techniek benodigd is. Mocht er een technische defect optreden dan heeft dit impact op de aanpasbaarheid en prestatie van het systeem. Wanneer een aanpassing niet kan worden uitgevoerd heeft dit ook zijn weerslag op de veiligheid van de bestuurders of passagiers. Ook wanneer het omgevingsbewuste systeem onder de maat presteert en informatie uit de omgeving niet tijdig worden doorgegeven kan dit de veiligheid bestuurder in gevaar brengen.

Technisch falen heeft impact op de Aanpasbaarheid, veiligheid en prestatie van een omgevingsbewust systeem.

Kennisgebrek:

Kennisgebrek is in hoofdstuk drie gedefinieerd als kennisgebrek bij de gebruiker van het systeem. Uiteraard dient een omgevingsbewust niet “misbruikt” te worden, dit is bij ieder systeem aan de orde. In het geval van een omgevingsbewust autosysteem dient de gebruiker wel op de hoogte te zijn van het doel en de mogelijkheden van het omgevingsbewust systeem. In het bijzonder van de mogelijke aanpassingen en handelingen die dit omgevingsbewuste systeem tot uitvoer kan brengen. Indien dit niet het geval is dan kan de bestuurder een schrikreactie vertonen, iets wat de veiligheid niet ten goede komt. Kennis gebrek heeft in deze specifieke casus dan ook impact op veiligheid. Niet op prestaties of aanpasbaarheid.

Communicatie issues:

Communicatie is cruciaal bij omgevingsbewuste systemen. Communicatie is de kunst van het overdragen van informatie. Deze informatie overdracht zal van A tot Z op goede wijze overgebracht dienen te worden. Wanneer het omgevingsbewuste autosysteem niet de remactie van zijn voorligger waarneemt dan kan dit verstrekende gevolgen hebben. Wanneer dit wel wordt waargenomen maar wanneer er niet juist gereageerd of aangepast wordt dan is het effect hetzelfde. Het omgevingsbewuste systeem is dan ook zo sterk als de zwakste (communicatie) schakel van het systeem. Er bestaat dan ook het risico dat er een verkeerde afweging wordt gemaakt.

Dit heeft invloed op de aanpasbaarheid van het systeem. Hierdoor de veiligheid van de ook veiligheid van de bestuurder en omgeving in het gedrang komen. Het niet tijdig hebben van informatie of het hebben van onvolledige informatie vanwege een communicatie storing zal resulteren in slechtere prestaties van het systeem.

Veiligheid:

Veiligheid in het verkeer is van levensbelang. Het systeem dient veilig te zijn en geen onvoorspelbaar gedrag te vertonen. Er dient duidelijk te zijn in welke situaties er wel en in welke situaties er niet ingegrepen wordt door het omgevingsbewuste autosysteem en in welke mate er ingegrepen kan worden. Indien dit niet op orde is, presteert het omgevingsbewuste autosysteem onder de maat. Dit vanwege de hoge veiligheidseisen die ook vanuit de overheid gesteld zullen worden. Gebrekkige veiligheid heeft dan ook impact op alle drie de systeemeigenschappen.

Privacy:

Met privacy in de context van omgevingsbewuste systemen wordt bedoeld of en in welke mate er zorgvuldig wordt omgegaan met de grote hoeveelheid gegevens die een omgevingsbewust systeem te verwerken krijgt. Blijven zulke gegevens lang bewaard, hoe worden deze bewaard en wie heeft toegang tot deze gegevens? Bij een omgevingsbewust autosysteem dient de informatie die een dergelijk systeem vergaart uiteraard niet op straat te belanden, van een groot privacy risico is echter geen sprake. Derhalve heeft privacy dan ook geen invloed op één van de gestelde eigenschappen.

Betrouwbaarheid:

Betrouwbaarheid is in het geval van een omgevingsbewust autosysteem van groot belang. Wanneer het systeem niet betrouwbaar is en niet of niet volledig werkt dan heeft dit impact op de veiligheid van de bestuurder en van andere weggebruikers. Niet alleen of een systeem betrouwbaar is maar ook de mate van betrouwbaarheid is van belang. Een omgevingsbewust systeem dat een klein beetje betrouwbaar is, is niet betrouwbaar. Een systeem dient volledig betrouwbaar te zijn.

Betrouwbaarheid of het gebrek hieraan, heeft dan ook impact om de prestaties en veiligheid van het systeem. Wanneer de betrouwbaarheid te wensen overlaat zal ook het vermogen om aan te passen afnemen.

Systeem intelligentie:

Als risico is dit het gebrek aan systeem intelligentie. Het omgevingsbewuste systeem zal een zekere mate van beslissingsbevoegdheid moeten hebben om tijdig te kunnen reageren of te kunnen handelen. Dit komt de prestaties van een omgevingsbewust systeem ten goede. Anderzijds moet ook de veiligheid van de bestuurders en de omgeving gewaarborgd blijven, het is niet gewenst dat het systeem rare beslissingen gaat maken of de verkregen informatie anders gaat weergeven of doorgeven dan gewenst is. Binnen CAMEL kan dit afgevangen worden door duidelijke voorwaarden in de vorm van constraints in te bouwen. Er zal op basis van de constraints bepaald worden of er wel of geen melding wordt gemaakt aan de betreffende arts vanwege afwijkende lichaamsfunctie waarden.

Wanneer gedrag van de omgeving als structureel wordt gekenmerkt kan hier bij terugkerende structureel gedrag, sneller op gereageerd worden. Het op een vaste dag op een gezette tijd een ruimte

verlaten of betreden kan als structureel gedrag worden gekenmerkt. Een aantal vraagtekens zijn hierover reeds aan bod gekomen in de casus van de moderne gezondheidszorg.

Wanneer de systeem intelligentie niet op niveau is dan heeft dit impact op de veiligheid van de bestuurder, passagier en omgeving. Ook de aanpasbaarheid komt zo in het gedrag. Wanneer een omgevingsbewust systeem te lang doet over het maken van een bepaalde beslissing of het geven van bepaalde feedback dan is dit ook niet bevorderlijk voor de prestaties.

Matrix Risico's versus Eigenschappen omgevingsbewust systeem omgevingsbewust autosysteem: toont de relatie tussen de betreffende systeem eigenschappen en in hoofdstuk drie besproken risico's.

Risico's / Eigenschappen	Prestatie	Veiligheid	Aanpasbaarheid
Technisch Falen	x	x	x
Kennisgebrek		x	
Communicatie issues	x	x	x
Veiligheid	x	x	x
Privacy			
Betrouwbaarheid	x	x	x
Systeem intelligentie	x	x	x

Hoofdstuk 6 – Conclusie

De onderzoeksvraag van deze thesis luidt als volgt:

Wat is een omgevingsbewust systeem, welke risico's brengt een dergelijk systeem met zich mee en hoe zijn deze risico's te beperken?

Een omgevingsbewust systeem is een systeem dat (vaak middels sensoren) informatie uit de omgeving opvangt om deze informatie vervolgens te filteren, te verwerken en vervolgens te reageren op deze wijzigingen in de omgeving. De opkomst van omgevingsbewuste systemen is sterk en geschiedt op allerlei terreinen. Denk hierbij aan hifi thuisinstallaties, sfeerverlichting maar ook in moderne autosystemen en zelfs in de moderne gezondheidszorg, dit laatste weliswaar nog in een teststadium.

Zoals de naam al aangeeft zal een omgevingsbewust systeem zich bewust moeten zijn van de omgeving. Wanneer een systeem niet af weet van het bestaan van de omgeving, kan er geen informatie verkregen worden uit de omgeving en derhalve ook niet gereageerd worden op de omgeving. Het omgevingsbewuste systeem monitort de omgeving in afwachting van eventuele wijzigingen.

Een omgevingsbewust systeem is op een aantal fronten geavanceerder dan een traditioneel systeem waarbij de gebruiker op directe wijze zorgt draagt voor de input.

Het verkrijgen van informatie uit de omgeving, het verwerken van deze informatie en het vervolgens op juiste wijze reageren op de omgeving zorgt voor een aantal bottlenecks die bij traditionele systemen niet aan de orde zijn. Ten eerste zal een systeem ingericht moeten zijn de omgeving te kunnen waarnemen. Vervolgens zal de grote stroom informatie die een omgevingsbewust systeem te verwerken krijgt, gefilterd moeten worden op basis van vooraf aangegeven waardes. Op basis van de combinatie van de verkregen informatie, vooraf aangegeven waardes en eerder gemaakte keuzes in soortgelijke situaties zal het systeem een beslissing maken hoe te reageren om de omgeving.

Een omgevingsbewust systeem vereist een groot aantal eigenschappen overeenkomstig met de eigenschappen van een traditioneel systeem met daarbij een aantal extra facetten in de vorm van een aantal extra eigenschappen. Naast deze extra functionele eisen worden moet er in het geval van omgevingsbewuste systemen ook op het gebied van de niet functionele eisen (de uitvoering van de functionele eisen) in grote mate aandacht worden besteed aan de kwaliteit van ontwikkeling, inrichting en uitvoering van deze niet functionele eisen. Zo is bij omgevingsbewuste systemen in grotere mate dan bij een traditionele systemen, behoefte aan een optimale prestatie, veiligheid, betrouwbaarheid en aanpasbaarheid aan de omgeving. Deze laatste eigenschap betreft het kunnen reageren op de omgeving of het kunnen reageren van wijzigingen in de omgeving.

Wanneer er tijdens de ontwikkeling van een omgevingsbewust systeem niet doelbewust rekening wordt gehouden met het grote belang van extra aandacht in de vorm van deze en de overige in hoofdstuk 2 genoemde eigenschappen dan zal het systeem en de omgeving van het systeem bloot komen te liggen aan een aantal risico's.

Één van de voornaamste risico's is het feit dat het omgevingsbewust systeem een niet bekend is met bepaald gedrag van de omgeving, dit gedrag niet kan verklaren of verwerken en hierdoor een verkeerde keuze maakt in de vorm van een ongewenste reactie. Een dergelijk scenario is te wijten aan een gebrek betrouwbaarheid, gebrek aan veiligheid, communicatie gebrek en een gebrek aan systeemintelligentie. Ieder van de hier genoemde oorzaken is weer een risico's op zich. Samen kunnen deze risico's resulteren in fatale scenario's.

Met de twee uitgewerkte casussen is aangetoond dat de risico's een flinke impact kunnen hebben. De berekende kans dat een dergelijk risico kan voorkomen is in deze thesis niet aan bod gekomen. Wel kan geconcludeerd worden dat bij zware medische afhankelijkheid en omgevingsbewuste systemen in auto's het disfunctioneren van een omgevingsbewust systeem een gigantische impact kan hebben met eventueel de dood tot gevolg.

Uit beide matrices is af te leiden dat de systeemeigenschappen van omgevingsbewuste systemen verband houden met de aanwezigheid van bepaalde risico's en andersom. Het ontbreken van één of meer van de drie eigenschappen kan leiden tot het optreden van de genoemde risico's.

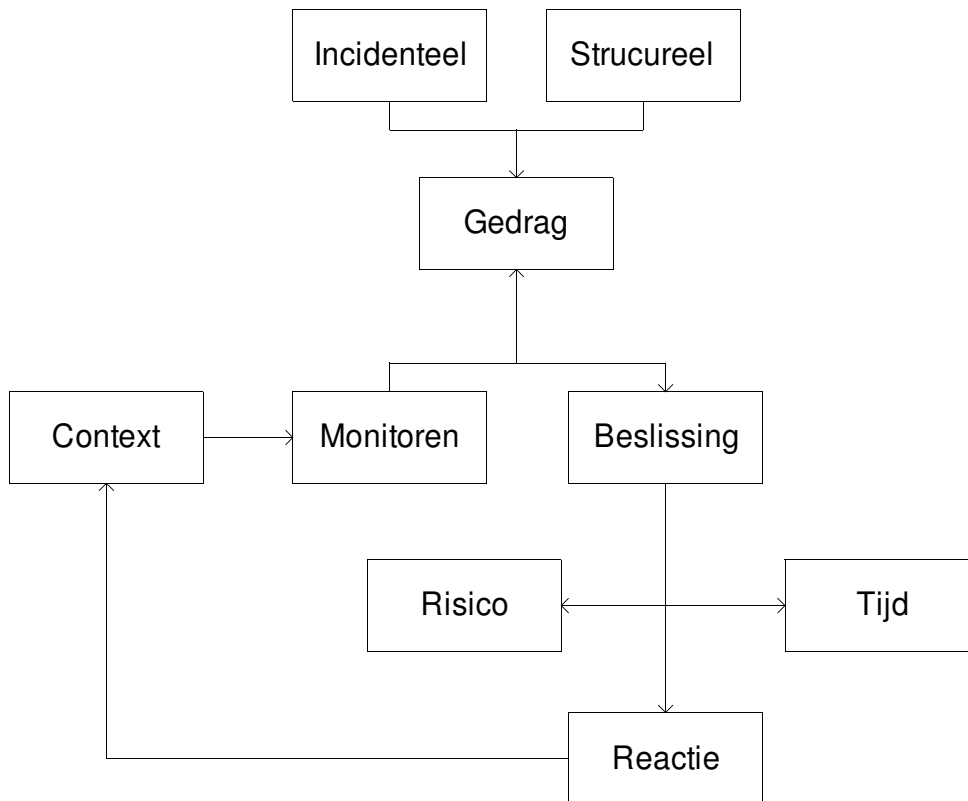
Om te zorgen dat een omgevingsbewust systeem van meet af aan op goede wijze tot stand komt en er met elke benodigde eigenschap rekening wordt gehouden is het zaak om vooraf goed na te denken over de eisen aan het tot stand te komen omgevingsbewuste systeem. Het systeem moet zich bewust zijn van de omgeving. Om dit te realiseren zal de omgeving mee gemodelleerd dienen te worden of zal het model van een bestaand systeem dienen te worden uitgebreid om het systeem of het model van het systeem omgevingsbewust te doen maken. Hiervoor biedt CAMEL twee meta modellen.

Uit deze meta modellen komt naar voren dat de vooraf opgegeven waardes (constraints) en de systeemintelligentie één van de belangrijkste aspecten van een omgevingsbewust systeem zijn. Dit is het gedeelte waar een beslissing wordt gemaakt over hoe te reageren op de omgeving. Deze beslissing staat centraal in het omgevingsbewuste systeem en is cruciaal in het feit of en in welke mate er risico's zullen optreden. Andersom kan het ontbreken van de de systeemeigenschappen resulteren in een verkeerde beslissing van het systeem wat logischerwijs ook risico's tot gevolg kan hebben.

Het omgevingsbewust zal onderscheid moeten kunnen maken tussen incidenteel en structureel gedrag van de omgeving. Indien bepaald gedrag als structureel is gekenmerkt kan er namelijk sneller gereageerd worden door het systeem. Het tijd element is van groot belang in de twee casussen, in zijn algemeenheid is bij omgevingsbewuste systemen de tijd waarin wordt gereageerd van belang. Men is niet gediend van een op het eerste oog geavanceerd omgevingsbewust systeem dat (te) lang doet over het verwerken van een reactie. Dit lange wachten zou ten koste gaan van de prestaties van het systeem. Tijd speelt dan ook een belangrijke rol in relatie tot omgevingsbewuste systemen.

Door rekening te houden met de in deze thesis aan bod gekomen elementen van omgevingsbewuste systemen kan de kans op het optreden van de genoemde risico's van een omgevingsbewust systeem worden verkleind. Ook een omgevingsbewust systeem zal nooit volledig risico vrij zijn.

Met behulp van het volgende meta model heb ik getracht om alle cruciale elementen van omgevingsbewuste systemen en hun onderlinge relatie in kaart te brengen. In het kort gesteld is dit meta model een grafische weergave van de door bij vastgestelde bevindingen tijdens het onderzoeksverloop van de thesis.



Figuur 32

De componenten van bovenstaand meta model zijn gedeeltelijk bestaande elementen en onder andere afgeleid uit de CAMEL taal. Denk hierbij aan gedrag (structureel / incidenteel), context, monitoren en reactie. De elementen beslissing, risico en tijd zijn hieraan toegevoegd. Samen met de bestaande elementen vormen de nieuwe elementen het meta model van risico's van omgevingsbewuste systemen. De relaties en pijlen tussen de elementen tonen de samenhang en het de werking van een omgevingsbewust systeem in algemene vorm.

6.1 Vervolgonderzoek

In een eventueel vervolgonderzoek zou er gekeken kunnen worden naar de gewenste mate van systeemintelligentie van een omgevingsbewust systeem. Er zal een balans gevonden moeten worden tussen de mate van systeemintelligentie van het systeem en de veiligheid en betrouwbaarheid van de te nemen beslissingen. Hierbij zouden patroonherkenning, indien op grote schaal datamining en mogelijk ook kunstmatige intelligentie een rol kunnen spelen.

Ook het vraagstuk over hoe het beste gedrag geïdentificeerd kan worden om het vervolgens tot of incidenteel of structureel gedragen te scharen kan een aanleiding zijn tot vervolgonderzoek. Ook hierbij zullen de eigenschappen van een omgevingsbewust systeem centraal dienen te staan en met name de juist balans tussen deze eigenschappen.

Literatuur

1. K. Raatikainen, H. Bærbak Christensen, T. Nakajima. Application Requirements for Middleware for Mobile and Pervasive Systems. *Mobile Computing and Communications Review*, Volume 6, Number 4. Oktober 2002.
2. K. Raatikainen, H. Bærbak Christensen, T. Nakajima. Application Requirements for Middleware for Mobile and Pervasive Systems. *Mobile Computing and Communications Review*, Volume 6, Number 4. Oktober 2002.
3. A. Ranganathan, R.H. Campbell. A Middleware for Context-Aware Agents in Ubiquitous Computing Environments. *Middleware 2003*. 2003.
4. A. Coronato, G. De Pietro, JH. Park, HC. Chao. A Framework for Engineering Pervasive Applications Applied to Intra-vehicular Sensor Network Applications. Springer Science + Business Media. April 2009.
5. M. Maia, L.S. Rocha, R.M.C. Andrade. Requirements and Challenges for Building Service-Oriented Pervasive Middleware. *ICPS'09*. Juli 2009.
6. E. Niemelä, J. Latvakoski. Survey of Requirements and Solutions for Ubiquitous Software. *Mobile Ubiquitous Computing Conference'04*. Oktober 2004.
7. N. Bicocchi, G. Castelli, M. Mamei, A. Rosi, F. Zambonelli, M. Baumgarten, M. Mulvenna. Knowledge Networks for Pervasive Services. *CPS'09*. Juli 2009.
8. U. Varshney. Pervasive Healthcare and Wireless Health Monitoring. *Mobile Netw Appl (2007)* / Springer Science + Business. Juli 2007.
9. K. Henricksen, J. Indulska. A Software Engineering Framework for Context-Aware Pervasive Computing. Proceedings of the Second IEEE International Conference on Pervasive Computing and Communications (PerCom'04). 2004
10. Y.C. Lee, E.S. Khorasani, S. Rahimi. , B. Gupta. A Generic Mobile Agent Framework for Ambient Intelligence. Proceedings of the 2008 ACM symposium on Applied computing . *Maart 2008*.

11. W.K. Edwards, R.E. Grinter. At home with Ubiquitous computing: Seven Challenges. Ubicomp 2001 / Springer-Verlag Berlin Heidelberg 2001. 2001.
12. G.D. Abowd, E.D. Mynatt. Charting Past, present and future research in ubiquitous Computing. ACM Transactions on Computer-Human interaction. 2000.
13. A.K. Dey. Understanding and Using Context to appear in Personal and Ubiquitous Computing, 2001.
14. J. Warmer, A. Kleppe. Praktisch UML 3de editie. 2004
15. B.A. Aubert, M. Patry, S. Rivard. A framework for IT outsourcing risk management. ACM SIGMIS Database Volume 36 Issue 4, Fall 2005. 2005.
16. C. Forst, D. Allen, J. Porter, and P. Bloodworth. Operational risk and resilience: understanding and minimizing operational risk to secure shareholder value. 2000.
17. J.G.M. Frijns, Rapport Risicomanagement De praktijk in Nederland, Amsterdam januari 2006.