

Eindverslag

Ontwikkeling Webwinkel

Module:

Afstuderen

Coördinator:

Dhr. A.M. Schipper

Begeleidend docent:

Dhr. E.M. Van Doorn

Examinator:

Dhr. E.H. Janssen

Opdrachtgever:

ASolutions - Advanced Software Solutions

Student:

Alexander Bouwens 99008510

Datum & Plaats:

8-10-2004, Den Haag

Versie:

1.0

Document versies

Versie	Datum	Auteur	Commentaar
0.1	20/09/04	A.J.W. Bouwens	Beoordeling concept
1.0	06/10/04	A.J.W. Bouwens	Final

Document distributie

Versie	Datum	Persoon
0.1	20/09/04	E.M. Van Doorn
0.1	20/09/04	E.H. Janssen
1.0	08/10/04	E.M. Van Doorn
1.0	08/10/04	E.H. Janssen
1.0	08/10/04	Gecommitteerde
1.0	08/10/04	S. Borman

Referaat

Bouwens, A.J.W., EINDVERSLAG Ontwikkeling webwinkel, Krimpen aan den IJssel, 2004

Dit verslag beschrijft het project zoals deze door de auteur is uitgevoerd ter afronding van de opleiding Informatica en Informatiekunde aan de Haagse Hogeschool. Dit project is uitgevoerd in naam van ASolutions - Advanced Software Solutions BV in de periode van 17/05/04 t/m 08/10/04.

Het doel van het project is het ontwikkelen van een generieke webwinkel.

Descriptoren:

- Webwinkel
- Internet
- Apache Jakarta Struts
- Java
- RAD (Rapid Application Development)
- UML (Unified Modelling Language)

Voorwoord

Ik wil hierbij graag gebruik maken van de gelegenheid om een aantal mensen te bedanken voor hun bijdrage. Natuurlijk wil ik iedereen bedanken die een bijdrage hebben geleverd aan het eindverslag. In het bijzonder wil ik de volgende personen bedanken:

Op de eerste plaats wil ik dhr. S. Borman bedanken voor zijn steun gedurende de gehele afstudeerstage, evenals het beschikbaar stellen van de mogelijkheid voor het afstuderen bij ASolutions te krimpen aan den IJssel.

Ten tweede wil ik graag dhr. R. Borman bedanken voor zijn constructieve bijdrage aan het eindverslag.

Als laatste wil ik dhr. E.M. Van Doorn en dhr. E.H. Janssen bedanken voor hun bijdrage aan het realiseren van het eindverslag, evenals het beoordelen hiervan.

Alexander Bouwens, Krimpen aan den IJssel, 7-10-2004

Inhoudsopgave

1. Inleiding	1
Deel 1 – De Opdracht	2
2. Organisatie	3
2.1 ASolutions	3
2.2 organisatorische inrichting	3
2.3 Plaats van de Afstudeerder binnen de organisatie	3
3. Opdrachtschrijving	4
3.1 Probleemstelling	4
3.2 Doelstelling	4
3.3 Cruciale succesfactoren	4
3.4 Uitgangssituatie	4
3.5 Werkzaamheden	5
3.6 Methoden en technieken	5
3.7 Resultaten	5
4.0 Plan van Aanpak	6
4.1 Inleiding	6
4.2 Aanpak	6
4.2.1 Afbakening	6
4.2.2 Projectbeheersing	6
4.2.3 Risicoanalyse	6
4.3 Planning	7
5. Methoden en technieken	9
5.1 Inleiding	9
5.2 keuzes en motivatie	9
5.1.1 keuze project beheersingsmethode	9
5.1.2 keuze modelleringstechniek	9
5.2 Rapid Application Development (RAD)	10
5.3 Unified Modelling Language (UML)	11
5.4 Apache Jakarta Struts Framework	12
5.4.1 Inleiding	12
5.4.2 Wat is een framework?	12
5.4.3 Het Jakarta Struts Framework	12
Deel 2 – Proces en Product	14
6. Requirementsplanning	15
6.1 Inleiding	15
6.2 Joint Requirements Planning sessie (JRP-sessie)	15
6.3 Scope	16
6.3.1 Afbakening	16
6.3.2 Functionaliteiten	16
6.3.3 Prioriteiten	17
6.3.4 Doelgroep van de webwinkel	17
7. User design	18
7.1 Inleiding	18
7.2 Joint Application Design Sessie (JAD-sessie)	18
7.3 Functionele eisen	19
7.3.1 Use case diagram	19
7.4 Niet-functionele eisen	21
7.5 Initieel Klassendiagram	22
7.6 Sequence diagrammen	24
7.7 Toestandsdiagrammen	25
7.8 Compleet klassediagram	26
7.9 Compleet klassediagram met struts	27
7.9.1 Configuratielaag	27
7.9.2 Databaselaag	28

7.9.3 controllerlaag.....	29
7.9.4 Modellaag.....	30
7.9.5 Viewlaag.....	31
7.10 ERD diagram.....	32
7.11 Schermontwerp.....	33
Deel 3 – Implementatie	34
8.0 Construction	35
8.1 Inleiding	35
8.2 Ontwikkeling van de webwinkel applicatie	35
8.3 Problemen.....	35
8.4 Testen.....	36
9. Cut Over.....	37
Deel 4 – Evaluatie	38
10. Evaluatie	39
10.1 Procesevaluatie.....	39
10.1.1 Gebruikte methode	39
10.1.2 Requirementsplanning	39
10.1.3 User Design	39
10.1.4 Construction	40
10.1.5 Cut Over	40
10.1.6 Gebruikte techniek.....	40
10.1.7 Communicatie binnen het project.....	41
10.1.8 Theorie en praktijk.....	41
10.2 Product Evaluatie	42
10.2.1 Scope	42
10.2.2 Functioneel ontwerp	42
10.2.3 Beheerderhandleiding	42
10.2.4 Webwinkel applicatie	42
Referenties	43
Literatuur	43
Digitale informatie.....	43
 Bijlage A Plan van Aanpak	
 Bijlage B Scope	
 Bijlage C Functioneel Ontwerp	
 Bijlage D Beheerderhandleiding	
 Bijlage E Testset en Testresultaten	

Lijsten van Tabellen en figuren

<i>Tabel 1, risico analyse</i>	<i>6</i>
<i>Tabel 2, Timebox webwinkel project.....</i>	<i>7</i>
<i>Tabel 3, Timebox Requirementsplanning fase</i>	<i>7</i>
<i>Tabel 4, Timebox User Design fase</i>	<i>8</i>
<i>Tabel 5, Timebox Construction fase</i>	<i>8</i>
<i>Tabel 6, Timebox Cut Over fase</i>	<i>8</i>
<i>Tabel 7, Timebox Rapportage fase</i>	<i>8</i>
<i>Tabel 8, functionaliteiten en prioriteiten van de webwinkel</i>	<i>17</i>
<i>Tabel 9, use case beschrijving in tekst volgens Warmer en Kleppe, Praktisch UML tweede editie</i>	<i>20</i>
<i>Tabel 10, Overzicht van gevonden klassen en attributen voor het initieel klassen diagram</i>	<i>22</i>
 <i>Figuur 1, Samenhang van de UML diagrammen.....</i>	 <i>11</i>
<i>Figuur 2, Wisselwerking tussen de verschillende struts componenten.....</i>	<i>13</i>
<i>Figuur 3, Use case diagram volgens UML</i>	<i>19</i>
<i>Figuur 4, Initieel klassendiagram</i>	<i>23</i>
<i>Figuur 5, Sequence diagram van bestellen</i>	<i>24</i>
<i>Figuur 6, Toestandsdiagram klasse winkelwagen</i>	<i>25</i>
<i>Figuur 7, Compleet klassendiagram</i>	<i>26</i>
<i>Figuur 8, Configuratielaag</i>	<i>27</i>
<i>Figuur 9, Databaselaag</i>	<i>28</i>
<i>Figuur 10, Controllerlaag</i>	<i>29</i>
<i>Figuur 11, Modellaag</i>	<i>30</i>
<i>Figuur 12, Viewlaag</i>	<i>31</i>
<i>Figuur 13, ERD-diagram</i>	<i>32</i>
<i>Figuur 14, schermontwerp webwinkel applicatie</i>	<i>33</i>

1. Inleiding

Ter afronding van mijn opleiding aan de Haagse Hogeschool heb ik (de afstudeerder) een afstudeeropdracht uitgevoerd voor ASolutions. De opdracht is het ontwikkelen van een generieke webwinkel.

Het project (de afstudeeropdracht) is zelfstandig uitgevoerd. De beschreven procesbeslissingen zijn dan ook door de afstudeerder genomen. Beslissingen qua inhoud en koers van het project zijn veelal in samenwerking met de opdrachtgever en ketenpartners (zie Hoofdstuk 2.3) genomen.

Het doel van dit rapport is een duidelijk beeld creëren van de uitgevoerde werkzaamheden, de werkwijze en opgeleverde producten tijdens het uitvoeren van het project.

Om een zo duidelijk mogelijk overzicht te geven van de werkzaamheden, werkwijze en opgeleverde producten is dit rapport op de volgende wijze ingedeeld:
In “deel 1 – de opdracht” wordt de opdrachtschrijving en het plan van aanpak gegeven en worden de gekozen methoden en technieken beschreven.
In “deel 2 – product en proces” worden de bedrijfsanalyse (requirementsplanning) en het functioneel,- en technisch ontwerp (user design) gegeven en beschreven.
In “deel 3 – implementatie” wordt de implementatie van de webwinkel beschreven.
In “deel 4 – evaluatie” worden tenslotte de opgeleverde producten en de processen die uitgevoerd zijn om tot de opgeleverde producten te komen beoordeeld.
In de afzonderlijke delen worden de faseringen die RAD (Rapid Application Development) methode voorschrijft gebruikt voor de hoofdstuk indeling.
Dit zorgt ervoor dat de lezer een chronologisch overzicht krijgt van de uitgevoerde werkzaamheden.

De volgende producten zijn opgenomen als externe bijlage:

- **Plan van aanpak**
Beschrijft de gehanteerde aanpak en planning voor het project
- **Scope**
Omvat een onderzoek naar de mogelijkheden op het gebied van de in de opdrachtschrijving genoemd probleem en de afbakening van het project.
- **Functioneel Ontwerp**
Het functioneel ontwerp is een document dat bij ASolutions gebruikt wordt als contract met de klant. Om die reden bevat het zowel functionele als enige technische eisen.
- **Beheerderhandleiding**
De beheerderhandleiding beschrijft hoe het systeem gemodelleerd is en beschrijft enkele functionaliteiten van het systeem.
- **Testset en Testresultaten**
Omvat de bij het testen gebruikte testsets en de resultaten van de test.

Deel 1

—

De Opdracht

2. Organisatie

De afstudeeropdracht is uitgevoerd in opdracht van ASolutions. Dit hoofdstuk zal deze organisatie beschrijven, evenals de plaats van de afstudeerder binnen de organisatie.

2.1 ASolutions

ASolutions is een ICT bedrijf gespecialiseerd in het aanbieden van diensten voor de ontwikkeling van software. Kort samengevat levert ASolutions maatwerksoftware, softwareonderhoud en consultancy. ASolutions is een familiebedrijf en bestaat uit een klein team van 10 medewerkers. Het bedrijf heeft vanwege de geringe omvang een zeer platte organisatie waar onderling vooral informeel wordt gecommuniceerd. Dankzij deze korte onderlinge lijnen zijn de werknemers in staat snel en flexibel te reageren.

2.2 organisatorische inrichting

ASolutions is te typeren als een netwerkorganisatie* waar ieders expertise wordt benut door hem in een bijpassend project te plaatsen. Elke werknemer voert dan ook een aanzienlijk deel van zijn taken uit bij de opdrachtgever.

2.3 Plaats van de Afstudeerder binnen de organisatie

Bij ASolutions wordt in projectteams gewerkt, dat had tot gevolg, dat aangezien de afstudeerder individueel aan het webwinkel project heeft gewerkt, de afstudeerder buiten de overige teams werkte. Vandaar dat de afstudeerder binnen ASolutions alleen met de stagebegeleider dhr. S. Borman heeft samengewerkt.

**Netwerkorganisatie: Een virtuele netwerk organisatie is een combinatie van meerdere –geografisch verspreide- partijen (personen en/of organisaties), die door bundeling van elkaar aanvullende kernactiviteiten en middelen een gezamenlijk doel trachten te bereiken. Deze virtuele organisatie kent een gelijkwaardige machtsverdeling van deelnemers en is voor de coördinatie van deze activiteiten afhankelijk van elektronische verbindingen (ICT infrastructuur).*

3. Opdrachtomschrijving

Dit hoofdstuk bevat de beschrijving van de opdracht. Dit hoofdstuk heeft als doel het verschaffen van een duidelijk inhoudelijk beeld van de afstudeeropdracht.

3.1 Probleemstelling

Om aan de vraag uit het bedrijfsleven te blijven voldoen wil ASolutions een generieke webwinkel ontwikkelen. Er is een potentiële klant bereid gevonden om de functionele eisen van de te bouwen webwinkel mede te bepalen. Op deze manier hoopt ASolutions haar klanten beter van dienst te zijn en het klantenbestand uit te breiden.

3.2 Doelstelling

Het doel van de afstudeeropdracht is het ontwikkelen van een generieke (herbruikbare) webwinkel applicatie, waarbij de interne werking (functionaliteit) van de webwinkel van groter belang is dan het uiterlijk. De applicatie moet aan de eisen voldoen die samen met een potentiële klant worden opgesteld.

3.3 Cruciale succesfactoren

Bij aanvang van het project werd duidelijk dat de omgeving van het project werd gevormd door vier partijen. De partijen zijn ASolutions, Boodschappen uit Holland (de potentiële klant), de grafisch ontwerper (Miranda Burger) en het bedrijf dat de creditcard betalingen afhandelt (Buckaroo). Het slagen van het project is grotendeels afhankelijk van de medewerking en beschikbaarheid van deze partijen. In de onderstaande lijst zijn de cruciale succesfactoren beschreven.

- Alle producten en mijlpalen zullen door de afstudeerder volgens planning ingeleverd worden. Producten dienen goedgekeurd te worden door de opdrachtgever.
- De medewerking en beschikbaarheid van zowel ASolutions als de omliggende partijen en de betreffende examinatoren.

In eerste instantie werden de factoren als een projectrisico beschouwd. Het werd echter duidelijk dat het meer dan alleen een risico was. Een risico is immers in de meeste gevallen te beheersen. Aangezien weinig invloed uitgeoefend kan worden op de medewerking en de beschikbaarheid van de ketenpartners (zie bovenstaand kader) werd besloten om dit als cruciale succesfactor te laten gelden.

3.4 Uitgangssituatie

Er is een businessplan opgesteld, waarin globaal staat hoe de applicatie eruit moet komen te zien, het bevat een kleine hoeveelheid functionele eisen.

De volgende software zal gebruikt worden:

- Windows 2000/2003.
- Office 2003.
- Eclipse.
- Java 2.
- MySQL.
- Tomcat server.
- Jakarta Struts Framework

3.5 Werkzaamheden

In het kader van de afstudeeropdracht zullen de volgende activiteiten verricht worden:

- Plan van aanpak opstellen.
- Uitvoeren requirementsplanning fase:
 - Timebox plannen.
 - Scope Samenstellen.
 - Uitvoeren joint requirementsplanning sessie.
- Uitvoeren User design fase:
 - Timebox plannen.
 - Uitvoeren Joint Application design sessie.
 - Prototype bouwen.
- Uitvoeren Construction fase:
 - Timebox plannen.
 - Applicatie bouwen.
 - Uitvoeren van de user reviewboard.
- Uitvoeren Cutover fase:
 - Timebox plannen.
 - Implementeren van de Webwinkel.

3.6 Methoden en technieken

RAD zal worden gebruikt als projectbeheersingsmethode. Deze methode maakt gebruik van timeboxing en incrementeel ontwikkelen. UML zal worden gebruikt als modelleringstechniek. De volgende diagrammen zullen in ieder geval worden toegepast:

- Use case diagram.
- Klassendiagram.
- Sequence diagram.

In hoofdstuk 4 zullen de methoden en technieken nader worden toegelicht.

3.7 Resultaten

De volgende producten zullen opgeleverd worden:

- Plan van aanpak.
- Webwinkel applicatie.
- Beheerderhandleiding webwinkel
- Scope
- Testset
- Testresultaten

4.0 Plan van Aanpak

4.1 Inleiding

Op basis van de opdrachtomschrijving is het plan van aanpak opgesteld. Het plan van aanpak is aan het begin van het project samengesteld en bevat daarom planningen die soms niet gehaald zijn en aannames die achteraf niet kloppen. In de evaluatie zal besproken worden welke planningen en aannames niet kloppen. De in de volgende paragrafen beschreven aanpak en planning is ondermeer totstandgekomen met het door RAD voorgeschreven timeboxing principe.

4.2 Aanpak

4.2.1 Afbakening

Voordat er daadwerkelijk aan het project begonnen werd, is eerst een globale afbakening opgesteld. Afsproken werd dat het project in twee delen zou worden gesplitst, een functioneel deel en een grafisch deel. Het functionele deel zou worden ontwikkeld door de afstudeerder en het grafische deel door grafisch ontwerper mevr. M. Burger. Voor beide delen werd onafhankelijk van elkaar een planning gemaakt.

4.2.2 Projectbeheersing

Alvorens men begint aan een project is het verstandig verschillende methoden, technieken en software omgevingen met elkaar te vergelijken en daaruit diegene te kiezen die het meest geschikt lijken voor het project. Voor het webwinkel project is er gekozen om gebruik te maken van de RAD (Rapid Application Development) methode. Voor het ontwerp is er gekozen voor de technieken van UML (Unified Modelling Language) en voor de realisatie van de webwinkel applicatie is er gekozen voor de programmeertalen Java en MySQL binnen het Struts framework. In hoofdstuk 5 Methoden en Technieken zullen de keuzes voor de methoden en technieken worden toegelicht, tevens worden de methoden en technieken uitgebreid beschreven.

4.2.3 Risicoanalyse

Om de kans op mislukking van het project zo klein mogelijk te houden is een risicoanalyse opgesteld.

Tabel 1, risico analyse

Risico	Soort	Kans	Maatregel	Mogelijk gevolg
Ziekte	Menselijk	Middel	In de planning is rekening gehouden met ziekte door extra tijd te plannen per fase	Er zal als gevolg van langdurig ziek zijn uitloop plaatsvinden.
Geen overeenstemming potentiële klant	Project	Gering	Door middel van het opstellen van een functioneel ontwerp in samenwerking met de klant, dat dient als contract, zal de kans op overeenstemming groter zijn.	Wanneer het functioneel ontwerp niet tijdig word gekeurd kan het project wellicht niet tijdig worden afgerond
Hardware/ Software	Technisch	Groot	Er wordt lokaal en op de netwerk server een back-up gemaakt van het werk.	Wanneer beide systemen uitvallen is uitloop onvermijdbaar.

De bovenstaande figuur illustreert de drie belangrijkste risico's. Deze zijn belangrijk omdat deze drie risico's uitloop van het project kunnen inhouden.

Het risico "ziekte" is opgenomen omdat dit veelal zonder waarschuwing kan voor komen. Om hierop te anticiperen is de planning voorzien een uitloop marge in iedere afzonderlijke fase.

Om te voorkomen dat de opdrachtgever na afronding van het project ontevreden is over het resultaat is het risico "Geen overeenstemming potentiële klant" opgenomen. Hierbij is als maatregel gekozen voor het opstellen van een functioneel ontwerp in samenwerking met de potentiële klant, dat als contract zal dienen.

Als derde risico is "Hardware/ Software" opgenomen. Om de continuïteit van het project te waarborgen is gekozen om gemaakt werk dubbel op te slaan. Het werk is dus op de lokale computer als op de server van het netwerk van ASolutions opgeslagen.

4.3 Planning

De keuze om RAD als projectbeheersingsmethode te gebruiken, heeft een grote invloed op de planning. Dit komt omdat RAD een specifieke manier van plannen voorschrijft genaamd Timeboxing. Timeboxing is een planningstechniek waarbij de opleverdatum van het te bouwen product een onwrikbare deadline is. Hierdoor dient de rek om eventuele uitloop op te vangen gevonden te worden in de scope, de scope is de functionele inhoud en kwaliteit van het op te leveren product. Men stelt aan het begin van het project de scope vast en gaat daarna op basis van de scope de overige fases van het project inplannen. Het inplannen is gebeurd met behulp van de MoSCoW regels (Must have, Should have, Could have, Won't have). Deze regels zijn erop gericht dat iedere fase wordt opgebouwd uit werkzaamheden met hoge en lage prioriteit. Het samen onderbrengen van zowel hoge,- als lage prioriteitswerkzaamheden in één timebox heeft tot gevolg dat bij uitloop van de hoge prioriteitswerkzaamheden er lage prioriteitswerkzaamheden in de timebox geschrapt kunnen worden zodat uitloop wordt voorkomen en de deadline dus toch gehaald wordt.

Bij het webwinkel project is er begonnen met opstellen een globale projectplanning en een planning voor de requirementsplanning fase van RAD. De globale projectplanning ingedeeld in de verschillende fasen die bij RAD doorlopen dienen te worden, met daarbij een extra fase rapporteren voor het schrijven van het eindverslag. Deze planning wordt in tabel 2 weergegeven.

Tabel 2, Timebox webwinkel project

ID	Task Name	Start	End	Duration	2004				
					Jun	Jul	Aug	Sep	
1	Requirementsplanning	17-5-2004	14-6-2004	4.20w					
2	User design	14-6-2004	21-7-2004	5.60w					
3	Construction	21-7-2004	16-9-2004	8.40w					
4	Cut over	16-9-2004	24-9-2004	1.40w					
5	Rapportage	24-9-2004	13-10-2004	2.80w					

De requirementsplanning fase van RAD is al van te voren ingepland omdat in deze fase de scope verkregen wordt waarop de timeboxes voor de overige fases gemaakt zullen worden. De planning van de requirementsplanning staat in tabel 3.

Tabel 3, Timebox Requirementsplanning fase

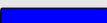
ID	Task Name	Start	End	Duration	May 2004	
					5-16	5-23
1	Joint Requirements Planning sessie	17-5-2004	21-5-2004	1w		
2	Scope	24-5-2004	14-6-2004	3.20w		

De timeboxes voor de overige fases van het project staan in tabel 4 t/m 7 weergegeven.

Tabel 4, Timebox User Design fase

ID	Task Name	Start	End	Duration	Jun 2004		Jul 2004	
							7-4	
1	Joint Application Design sessie	14-6-2004	18-6-2004	1w				
2	initieel klassediagram	18-6-2004	21-6-2004	0.40w				
3	Use Cases	21-6-2004	22-6-2004	0.40w				
4	Niet-functionele eisen	22-6-2004	22-6-2004	0.20w				
5	Sequence diagrammen	22-6-2004	24-6-2004	0.60w				
6	Toestandsdiagrammen	24-6-2004	30-6-2004	1w				
7	Compleet klassediagram	30-6-2004	6-7-2004	1w				
8	Schermontwerp	6-7-2004	8-7-2004	0.60w				
9	Prototype	8-7-2004	14-7-2004	1w				
10	Joint Application Design sessie	14-7-2004	21-7-2004	1.20w				

Tabel 5, Timebox Construction fase

ID	Task Name	Start	End	Duration	Jul 2004	Aug 2004					Sep 2004	
					7-25	8-1	8-8	8-15	8-22	8-29	9-5	
1	database laag	21-7-2004	23-7-2004	0.60w								
2	controller laag	23-7-2004	12-8-2004	3w								
3	model laag	12-8-2004	1-9-2004	3w								
4	view laag	1-9-2004	7-9-2004	1w								
5	creditcard betalingen	7-9-2004	8-9-2004	0.40w								
6	testen	8-9-2004	14-9-2004	1w								
7	user reviewboard	14-9-2004	16-9-2004	0.60w								

Tabel 6, Timebox Cut Over fase

ID	Task Name	Start	End	Duration	Sep 2004	
					9-12	9-19
1	Implementatie	16-9-2004	16-9-2004	0.20w		
2	Beheerdershandleiding samenstellen	16-9-2004	24-9-2004	1.40w		

Tabel 7, Timebox Rapportage fase

ID	Task Name	Start	End	Duration	Sep 2004	
					9-19	9-26
1	Verslag afronden	24-9-2004	7-10-2004	2w		

5. Methoden en technieken

5.1 Inleiding

Dit hoofdstuk dient ter verduidelijking van de gekozen methoden en technieken. Eerst worden de keuzes van de methoden en technieken toegelicht en vervolgens worden de methoden en technieken beknopt beschreven.

5.2 keuzes en motivatie

In deze paragraaf worden de keuzes toegelicht die gemaakt zijn met betrekking tot de project beheersingsmethode, modelleringstechniek en programmeertalen.

Per methode of techniek wordt gemotiveerd waarom juist voor deze methode of techniek gekozen is.

5.1.1 keuze project beheersingsmethode

Bij de keuze van de te gebruiken projectbeheersingsmethode is rekening gehouden met de volgende alternatieven:

- SDM (System Development Methodology)
- RAD (Rapid Application Development)
- DSDM (Dynamic System Development Methodology)

De methode waarmee ik (de afstudeerder) het meest bekend ben is SDM, omdat deze methode uitgebreid behandeld en gebruikt is op school. Met RAD en DSDM heb ik kennis gemaakt tijdens de module SO-07 (Methoden en technieken van systeemontwikkeling). Beide methoden heb ik al eens eerder gebruikt bij verschillende projecten.

Ondanks dat ik vaker met SDM heb gewerkt, is er niet voor deze methode gekozen, dit omdat deze methode vooral bedoeld is voor de beheersing van grote projecten met een looptijd van een halfjaar of langer. Dit zelfde geldt voor DSDM. Om die reden is de uiteindelijke keuze gevallen op RAD. RAD is uitermate geschikt voor de beheersing van kleine(re) projecten, zoals de huidige webwinkel opdracht. Daarnaast heb ik goede ervaring opgedaan met het gebruik van RAD en UML tezamen, bij bijvoorbeeld het ip-03 project. Het feit dat UML goed in te passen is in deze methode is nog een extra pluspunt aangezien ik met een object georiënteerde programmeertaal ga werken.

5.1.2 keuze modelleringstechniek

Bij de keuze van modelleringstechniek was vrij beperkt aangezien er vanuit school maar met twee technieken gewerkt is, namelijk Yourdon en UML. Een nieuwe techniek bestuderen zou in verband met de tijdsduur van het project wellicht niet haalbaar of onwenselijk zijn.

Aangezien de webwinkel applicatie object georiënteerd geprogrammeerd zal worden, was de keuze voor de modelleringstechniek zo gemaakt, want alleen UML is voor objectoriëntatie geschikt.

5.1.3 Keuze programmeertalen en framework

De keuze van programmeertaal is genomen bij het bespreken van een mogelijke opdracht bij ASolutions. Tijdens de bespreking gaf de afstudeerder aan dat hij graag een project met Java wilde uitvoeren, omdat hij al enige ervaring met Java had en zich er graag meer in wilde verdiepen. ASolutions bood vervolgens de webwinkel applicatie opdracht aan als een mogelijk project. Aan het webwinkel project werd vervolgens het struts framework gekoppeld aangezien ASolutions gehoord had dat dit framework goed werkt en zij er ervaring in wilde opdoen.

Er is gekozen voor MySQL als database, dit met name omdat het eenvoudig te gebruiken, platform onafhankelijk en kosteloos is.

5.2 Rapid Application Development (RAD)

Het webwinkel project is uitgevoerd aan de hand van de RAD-methode.

RAD staat voor Rapid Application Development en is een methode die een iteratief werkproces beschrijft. Dit houdt in dat, in tegenstelling tot een "waterval" methode als SDM, een applicatie ontwikkeld wordt in losse onderdelen (of modules).

Deze losse modules worden van ontwerp tot aan implementatie toe door ontwikkeld zonder het ontwikkelproces van die module te onderbreken en een bepaalde stap uit het proces opnieuw uit te voeren. Het voordeel hiervan is dat een module snel volledig geïmplementeerd en getest kan worden door de opdrachtgever, of de betrokken eindgebruikers.

Zo kan er door de eindgebruikers, die uiteindelijk met de applicatie moeten gaan werken, direct feedback gegeven worden over de applicatie.

Wensen voor nieuwe functionaliteiten of veranderingen in bestaande functies kunnen direct ontworpen en geïmplementeerd worden waar vervolgens weer snel feedback op gegeven kan worden.

Bij RAD is de snelheid van ontwikkelen het belangrijkste uitgangspunt. Dit wordt bereikt door deze iteratieve manier van werken. De klant krijgt snel resultaten te zien en kan daardoor veel bij het project betrokken worden wat leidt tot een applicatie die beter aansluit op de wensen van de eindgebruikers.

RAD schrijft een viertal fasen voor die één voor één uitgevoerd worden tijdens het ontwikkeltraject. Deze vier fasen zijn:

- Requirements Planning
- User Design
- Construction
- Cut Over

De *requirementsplanning* is een beschrijving van de inhoud van het te ontwikkelen systeem.

Hierin worden zaken behandeld als een globale project beschrijving, bepalen van de systeemgrenzen, bepalen van de betrokken groepen en de prioriteiten binnen het project.

Dit alles wordt opgenomen in de scope. De requirementsplanning wordt eenmalig opgesteld voor het gehele project, maar kan ook bij grotere projecten per module apart opgesteld worden. De *User Design* is waar het ontwerp voor het systeem (onderdeel) gemaakt wordt.

De user design omvat zowel het functionele en het technische aspect van het ontwerp en is dus de bron voor de implementatie. RAD beschrijft geen techniek om een systeem te ontwerpen en laat de ontwikkelaar vrij om zelf een techniek te kiezen. Voor de webwinkel applicatie is er gekozen om UML te gebruiken voor het maken van de User Design. In de *Construction* fase wordt de module voor de applicatie geprogrammeerd.

Het is aan te bevelen een hogere generatie programmeertaal te gebruiken omdat die vaak een veel snellere ontwikkelsnelheid hebben dan oudere programmeertalen.

Bij de *Cut Over* wordt de ontwikkelde module geïmplementeerd in het systeem en kan dan getest worden op technisch en functioneel niveau. In deze fase kunnen de betrokken eindgebruikers ook feedback geven over eventuele wensen die ze nog hebben.

Wanneer er nog veranderingen nodig zijn, kan men terug naar de User Design gaan om deze aan te passen om te voldoen aan de wensen van de eindgebruikers. De cyclus gaat dan weer

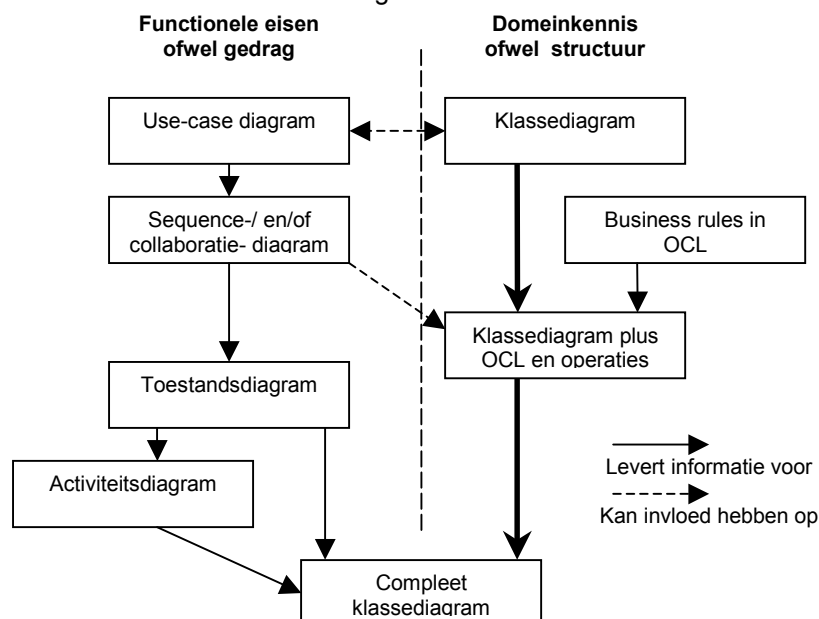
door naar de Construction fase en zal uiteindelijk weer in de Cut Over terecht komen. Deze cyclus tussen User Design en Cut Over kan herhaald worden totdat er een module staat waar de opdrachtgever tevreden mee is.

5.3 Unified Modelling Language (UML)

Zoals vermeld staat is er gekozen voor UML als modelleringstechniek. In tegenstelling tot wat veel ontwikkelaars denken, is UML geen methode, het is een visuele standaard taal (techniek), dat wil zeggen het standaardiseert geen werkwijze, maar begrippen en diagrammen waarin die begrippen gebruikt kunnen worden. UML wordt dus gebruikt om de software te modelleren in een project dat beheerst wordt door een ontwikkelmethode zoals bijvoorbeeld SDM. UML biedt een aantal diagrammen die gezamenlijk het model van het software systeem vormen. Niet al deze diagrammen hoeven perse in een project gemaakt te worden, de ontwikkelaar kiest zelf welke diagrammen hij wel of niet wenst te gebruiken.

- Het usecase diagram toont hoe het systeem gebruikt kan worden door externe entiteiten zoals menselijke gebruikers.
- Het klassendiagram toont de statische structuur van het softwaresysteem weergegeven als klassen en hun relaties.
- Het objectdiagram toont de statische structuur van het softwaresysteem weergegeven als objecten en hun relaties.
- Het sequencediagram toont de volgorde in tijd van de boodschappen die in het systeem verstuurd en ontvangen worden.
- Het collaboratiediagram toont de toestanden waarin een object zich kan bevinden gedurende zijn levensloop.
- Het activiteitsdiagram toont de activiteiten die door een deel van het systeem worden uitgevoerd, inclusief parallelisme.
- Het component diagram toont de verdeling van het gehele systeem in componenten en de relaties tussen die componenten.
- Het deployment diagram toont hoe de softwarecomponenten in een bepaalde systeemconfiguratie worden gebruikt.

Naast de diagrammen biedt UML ook een standaard taal waarin beperkingen, condities en regels die gelden in de diagrammen weergegeven kunnen worden. Deze taal heet de Object Constraint Language (OCL). Door middel van deze diagrammen en OCL komt men uiteindelijk tot de modellering van de te ontwikkelen software, figuur 1 geeft de samenhang tussen de verschillende diagrammen weer.



figuur 1, samenhang van de UML diagrammen. Bron: Warmer en Kleppe, Praktisch Uml 2^e editie

5.4 Apache Jakarta Struts Framework

5.4.1 Inleiding

Dit hoofdstuk zal het apache Jakarta struts framework beschrijven.

Voordat echter dieper ingegaan zal worden op struts het struts framework, zal eerst een korte uitleg gegeven worden over wat een framework eigenlijk inhoud.

5.4.2 Wat is een framework?

Er zijn veel verschillende interpretaties van wat een framework precies inhoud.

Een framework is te vergelijken met een software library Net als een library bevat een framework methoden en routines die te gebruiken zijn binnen de applicatie die ontwikkeld wordt. Het grote verschil tussen een software library en een framework is dat een framework een set generieke componenten levert waarop de applicatie gebouwd wordt. Dus in plaats van een aantal classes erbij halen die gebruikt worden om de functionaliteit van de applicatie uit te breiden zoals bij een library, levert een framework een set generieke basis functionaliteiten die uitgebreid worden door de applicatie. Een logisch gevolg hiervan is dus dat het framework de programmeur voorschrijft hoe hij de classes van zijn applicatie dient vorm te geven.

5.4.3 Het Jakarta Struts Framework

Het struts framework is een framework dat gecreëerd is door Craig McClanahan.

In 2000 heeft hij het project gedoneerd aan de apache software foundation.

Vanaf dat moment is het een opensource framework geworden dat verder ontwikkeld wordt door vrijwilligers. Ondertussen bestaat het framework al uit meer dan 300 classes en is versie 1.6 net uitgegeven.

Het struts framework is speciaal ontworpen voor het bouwen van Java webapplicaties. Het doel van het struts framework is programmeurs te dwingen hun webapplicaties op een dusdanige wijze te bouwen dat deze zo onderhoudbaar en herbruikbaar mogelijk zijn.

Struts poogt onderhoudbaarheid en herbruikbaarheid te bereiken door gebruik te maken van het Model-View-Controller (MVC) model. Dit model splitst de applicatie op in drie lagen, te weten de model,- de view,- en controllerlaag. Om deze reden wordt dit model ook wel het drie-lagen model genoemd.

De viewlaag is de bovenste laag in het model en bevat alle schermen. Het doel van de viewlaag is het bieden van een platform waarop de gebruiker opdrachten (requests) aan de applicatie doorgeeft en om applicatie reacties (responses) op weer te geven.

De modellaag is de middelste laag en bevat de functionaliteiten van de applicatie.

De controllerlaag is de onderste laag van het model. Deze laag heeft tot doel de opdrachten (requests) van de viewlaag te beoordelen en door te geven aan de modellaag die de opdracht uitvoert. Als vervolgens de modellaag klaar is met het uitvoeren van de opdracht bepaald de controllerlaag wat de reactie (response) van de applicatie zal zijn en stuurt de response terug naar de viewlaag die de response weergeeft.

De afhankelijkheid tussen de onderlinge lagen loopt van boven naar beneden, dit heeft tot gevolg dat de viewlaag applicatie specifiek is, dat de modellaag volledig herbruikbaar is door een soortgelijke applicatie en dat de controllerlaag bij alle soorten applicaties herbruikbaar is. Struts levert grotendeels zelf de controllerlaag voor alle applicaties die met struts ontwikkeld worden.

Een andere wijze waarop struts de afhankelijkheid tussen de onderlinge lagen zo klein mogelijk houdt en dus de herbruikbaarheid van de code zo groot mogelijk houdt, is de wijze waarop de lagen met elkaar communiceren. In struts verloopt de communicatie tussen de lagen door middel van het sturen van opdrachten (requests) en reacties van de applicatie (responses). Als een laag echter ook informatie in de vorm van data wil meegeven aan de

volgende laag dan koppelt de zendende laag een data transfer object (DTO's) aan de opdracht (request) of reactie (response). Een data transfer object is een object dat tussen de lagen staat en alleen gebruikt wordt om data tijdelijk in op te slaan.

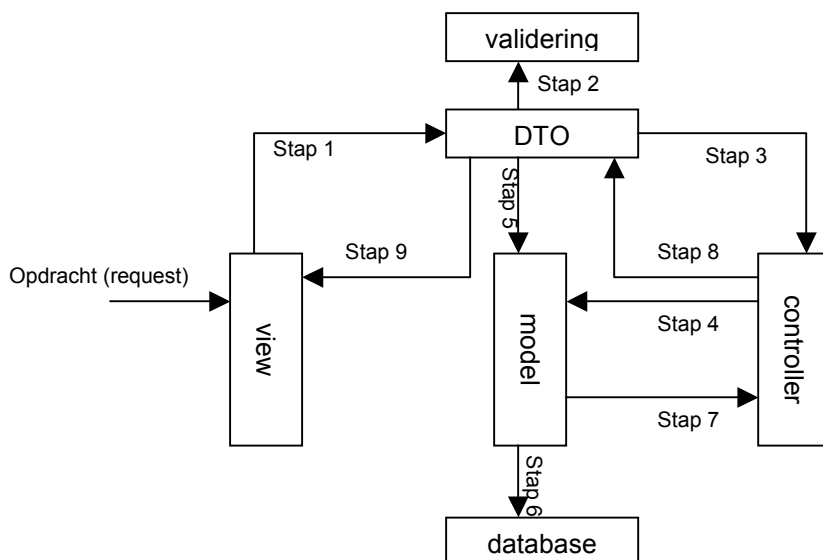
Naast gebruik te maken van het MVC-model maakt struts ook gebruik van tag-libraries om de onderhoudbaarheid te vergroten. Door gebruik te maken van tag-libraries probeert struts de viewlaag toegankelijker te maken voor webdesigners en grafisch ontwerpers.

Om dit te bereiken hebben de ontwikkelaars van struts besloten de viewlaag zo vorm te geven dat het lijkt alsof er zich op deze laag alleen html en javascript bevindt.

In werkelijkheid zijn de "html" tags uit de tag-libraries aanroepen van Java code die zeer uiteenlopende functionaliteiten in zich herbergen, van het maken van een tabel tot het doorlopen van een for-loop of het maken van een frame. Op deze manier is de viewlaag, waaraan vaak aanpassingen worden verricht, ook door "leken" te onderhouden. In de praktijk worden voor de viewlaag vaak de tag-libraries in jsp's gebruikt, maar struts ondersteunt vele webdesign talen om de viewlaag vorm te geven, van html, xml tot javascript of wml.

Het laatste onderdeel van struts is de validatie module. Deze module is opgenomen om ervoor te zorgen dat er geen ongeldige data de applicatie inkomt. Aan elk veld in een "html" form worden eisen opgesteld hoe de data die in dat veld wordt ingevoerd eruit moet zien. Dit gebeurt in een xml bestand. In dit bestand staat bijvoorbeeld of een waarde een getal moet zijn of niet. Als de waarde van het veld vervolgens gesubmit wordt en in een DTO gestopt wordt dan wordt de validatie module aangeroepen die de waarden controleert. Voldoet de data niet aan de eisen dan krijgt de gebruiker een melding dat hij de waarde moet aanpassen.

Om de wisselwerking tussen al deze componenten duidelijk te maken is in figuur 2 een sterk vereenvoudigd overzicht gegeven van de stappen die uitgevoerd worden bij het verwerken van één opdracht (request).



Figuur 2, wisselwerking tussen de verschillende struts componenten.

Er zijn nog drie eigenschappen van het struts framework die nog niet behandeld zijn, namelijk de gevolgen van de opensource opzet, de jonge leeftijd van struts en dat de modules vaak door verschillende ontwikkelaars gebouwd zijn.

Het opensource karakter van het struts framework heeft tot gevolg dat er nogal wat meningsverschillen onder programmeurs en de ontwikkelaars van struts zijn wat de juiste invulling is van de MVC-model voor struts. De jonge leeftijd van struts heeft tot gevolg dat de documentatie over struts nogal beperkt en gelijksoortig is.

Aangezien de modules vaak door verschillende ontwikkelaars gebouwd zijn verloopt de communicatie tussen de verschillende modules soms nogal moeizaam en onnodig ingewikkeld.

Deel 2

—

Proces en Product

6. Requirementsplanning

6.1 Inleiding

Dit hoofdstuk beschrijft de requirementsplanning fase van RAD.

Volgens RAD is het doel van deze fase het vaststellen van de scope (reikwijdte) van het project. De scope positioneert de te ontwikkelen applicatie binnen de informatiearchitectuur van de organisatie waar het gaat functioneren.

In de scope wordt vastgelegd wat de systeemgrenzen zijn, welke koppelingen er zijn met de omgeving en welke functies de applicatie zou moeten krijgen.

Het vaststellen van de scope gebeurt door middel van gesprekken met de opdrachtgever en eventueel de toekomstige gebruikers in een zogenaamde Joint Requirements Planning Sessie (JRP-sessie).

6.2 Joint Requirements Planning sessie (JRP-sessie)

De Joint Requirements planning sessie (JRP-sessie) is een bijeenkomst van de projectleider (de afstudeerder), IT-specialist (dhr. S. Borman), grafisch ontwerper (mevr. M. Burger) en de opdrachtgevers annex potentiële klant (dhr. S. Borman en mevr. A. Aldershof).

De reden dat dhr. S. Borman zowel op de stoel van de IT-specialist als op de stoel van de opdrachtgever zit, is omdat hij de eis van een generieke applicatie stelt aan de te ontwikkelen webwinkel applicatie.

Tijdens de Joint Requirements Planning sessie (JRP-sessie) werd de te ontwikkelen webwinkel applicatie in hoofdlijnen besproken. Tijdens de gesprekken kwamen onder andere aan de orde:

- Afbakening
- Functionaliteiten
- Prioriteiten
- Doelgroep

Het doel van de Joint Requirements Planning sessie (JRP-sessie) was het bereiken van consensus over de inhoud van de te ontwikkelen webwinkel applicatie en de prioriteiten hierin. Belangrijk hierbij was het vaststellen welke functionaliteiten essentieel zijn voor het functioneren van de webwinkel applicatie en dus een hoge prioriteit krijgen en welke functionaliteiten van minder groot belang zijn en dus een lage prioriteit krijgen. De bevindingen uit de Joint Requirements Planning sessie (JRP-sessie) zijn opgenomen in de scope en in de bijlage functioneel ontwerp.

6.3 Scope

6.3.1 Afbakening

Bij de afbakening van het webwinkel project was het belangrijk om in de gaten te houden dat de grens van het project niet op een dusdanige wijze getrokken werd dat de probleemstelling niet meer opgelost zou kunnen worden.

Met dit in het achterhoofd is het project begrensd bij het ontwikkelen van één webwinkel applicatie. Deze applicatie is alleen gericht op het mogelijk maken van de verkoop van producten op het internet. Functionaliteiten als het eventueel inpassen in een bestaand informatiesysteem vallen buiten de reikwijdte van het project.

Tevens is besloten het project in twee delen op te splitsen, een functioneel deel en een grafisch deel. Het functionele deel zal worden ontwikkeld door de afstudeerder en het grafische deel door grafisch ontwerper mevr. M. Burger.

6.3.2 Functionaliteiten

Tijdens het bespreken van de functionaliteiten ging het erom dat vast kwam te staan wat precies de wensen van de opdrachtgevers waren en of deze wensen haalbaar waren. Een belangrijk aspect hierbij was dat de wensen en eisen van de opdrachtgevers de probleemstelling oplossen.

Alvorens dieper in te gaan op de functies van de webwinkel zijn eerst globale eisen opgesteld. De globale eisen zijn:

- Generieke applicatie, dus de code is vrijwel volledig herbruikbaar bij een andere webwinkel.
- Bezoekers kunnen producten bekijken en zoeken.
- Bezoekers kunnen producten aan hun winkelwagen toevoegen
- Bezoekers kunnen favoriete producten aan een persoonlijke boodschappenlijst toevoegen voor hergebruik, zodat zij bij een tweede bezoek niet opnieuw naar die producten hoeven te zoeken.
- Bezoekers kunnen producten bestellen.

Vervolgens is er dieper ingegaan op de globale eisen wat leidde tot de volgende functionaliteiten:

- Bestelfunctie
 - De bezoeker kan een product selecteren en het te bestellen aantal instellen.
 - Als de bezoeker op product toevoegen klikt dan wordt de winkelwagen gevuld met het product en het gewenste aantal.
 - Als het product al in de winkelwagen aanwezig is dan wordt het te bestellen aantal opgehoogd met het ingestelde aantal.
 - Wordt voor het eerst een product toegevoegd aan de winkelwagen dan moet de gebruiker de regio selecteren (Eu, Europa buiten EU, overige landen), vervolgens de verzendwijze (standard/priority default is standard).
Na het selecteren berekent de webwinkel de verzendkosten en het totaalbedrag.
 - Als de bezoeker op bestellen klikt dan moet hij zijn persoonsgegevens invoeren en betaalt hij.
 - Na betaling ontvangt de bezoeker een orderbevestiging per e-mail.
 - De eigenaar van de webwinkel krijgt ook een email met de bestelling.
Het emailadres van de eigenaar van de webwinkel is instelbaar.
- Boodschappenlijstfunctie
 - De bezoeker kan een product toevoegen aan zijn persoonlijke boodschappenlijst
 - De bezoeker kan een product verwijderen uit de boodschappenlijst
 - De bezoeker kan een product toevoegen aan winkelwagen vanuit de boodschappenlijst
- Zoekfunctie
 - De bezoeker kan zoeken op productnaam
 - Zijn er geen resultaten dan wordt de zoekterm opgeslagen in de database en krijgt de bezoeker een optie te zien voor het aanvragen van een offerte om het niet bezoeker product alsnog te bestellen
- Offerte aanvraag functie
 - De bezoeker kan een offerte aanvragen door op "offerte aanvragen" te klikken.
 - Na versturing ontvangt de gebruiker direct een ontvangst bevestiging

- Product onderhoudsfunctie
 - De administrator kan inloggen op de onderhoudsmodule
 - De administrator kan de producten op de site wijzigen door een csv-bestand met productinformatie te uploaden naar de server.
 - De administrator kan de niet gevonden zoektermen opvragen en verwijderen uit de database.
 - De administrator kan afbeeldingen uploaden naar de webwinkel
- Betaalwijze
 - De bezoeker kan met creditcard betalen, de creditcard betalingsmodule zal geleverd worden door Buckaroo BV.
- Taalfunctie
 - De webwinkel moet in de volgende talen beschikbaar zijn:
 - Nederlands
 - Engels
 - De gebruiker kan zelf de taal van zijn voorkeur kiezen
- Webpollfunctie
 - De bezoeker kan middels een webpoll aangeven welke producten hij nog mist in de product voorraad.
- Informatieschermen
 - Er komen informatie schermen over de privacy policy en de verzendkosten.
- Generiek
 - De webwinkel moet generiek zijn, dat wil zeggen bruikbaar bij verschillende afnemers zonder al te veel aanpassingen.

6.3.3 Prioriteiten

Toen eenmaal de functionaliteiten bekend waren, is er overgegaan tot het stellen van de prioriteiten. Dit is gedaan met gebruikmaking van de MoSCoW (Must have, Should have, Could have, Won't have) regels. Tabel 8 geeft een overzicht van de functionaliteiten en hun prioriteiten.

Tabel 8, functionaliteiten en prioriteiten van de webwinkel

Functionaliteit	Prioriteit
Bestelfunctie	Hoog
Product onderhoudsfunctie	Hoog
Betaalwijze	Hoog
Generiek	Hoog
Boodschappenlijstfunctie	Middel
Zoek functie	Middel
Offerte aanvraagfunctie	Middel
Taalfunctie	Middel
Webpollfunctie	Laag
Informatieschermen	Laag

Gezien de beperkte tijdsduur van de afstudeerperiode is er besloten de onderdelen met de laagste prioriteit te laten vervallen.

6.3.4 Doelgroep van de webwinkel

Het is van belang de doelgroep goed te definiëren, dit omdat het opgeleverde product op de doelgroep moet aansluiten qua taal, lay-out etc. zodat het eindproduct ook daadwerkelijk bruikbaar voor de doelgroep is.

De potentiële klant wilde graag een webwinkel hebben voor Nederlanders in het buitenland die producten willen kopen die alleen in Nederland verkrijgbaar zijn. Aangenomen is dat het hier zal gaan om Nederlanders die van middelbare leeftijd zijn en zowel Engels als Nederlands als voertaal hebben.

Daarnaast moet de webwinkel generiek worden. Het is dus zaak dat als er een afnemer komt die een webwinkel wilt, die een andere doelgroep voor ogen heeft, dat de niet generieke code van webwinkel dan eenvoudig aangepast kan worden aan de juiste doelgroep.

7. User design

7.1 Inleiding

Het doel van de fase User design is om te komen tot een specificatie van het te ontwikkelen systeem, dat wil zeggen de functionele en technische eisen moeten worden vastgelegd. Dit gebeurt met name in de Joint application design sessie (JAD-sessie). In deze fase worden de bij andere methoden bekende functioneel,- en technisch ontwerp gemaakt.

7.2 Joint Application Design Sessie (JAD-sessie)

De Joint Application Design is een overleg tussen de IT-specialisten die aan het project werken en waarvan minimaal één persoon ook aanwezig is geweest op de JRP-sessie. De Joint Application Design sessie kent twee bijeenkomsten. In de eerste bijeenkomst wordt op basis van de in de requirementsplanning opgestelde scope, de specificaties gedetailleerd uitgewerkt. In de tweede bijeenkomst worden aan de hand van een prototype de specificaties van (delen) van het systeem verbeterd. Dit prototype is gebaseerd op de in de eerste bijeenkomst opgestelde specificaties.

De eerste bijeenkomst leverde de specificaties op die gemodelleerd zijn in de UML diagrammen die verderop in dit hoofdstuk aanbod komen.

Bij de tweede bijeenkomst werd het prototype getest. Het prototype omvatte een productoverzicht waarbij producten in de winkelwagen geplaatst konden worden. Dit prototype werd goedgekeurd en is in de construction fase gebruikt als voorbeeld waarnaar de rest van het programma gebouwd ging worden.

7.3 Functionele eisen

De functionele eisen zijn vastgelegd door middel van de use case techniek van UML. Deze functionele eisen zijn grotendeels gebaseerd op de scope die verkregen is na het uitvoeren van de requirementsplanning.

Er is ervoor gekozen om use cases te maken omdat deze techniek het mogelijk maakt om functionele eisen op een formele en makkelijk te begrijpen manier vast te leggen. Use cases bestaan uit een model en een beschrijving van het model in tekst.

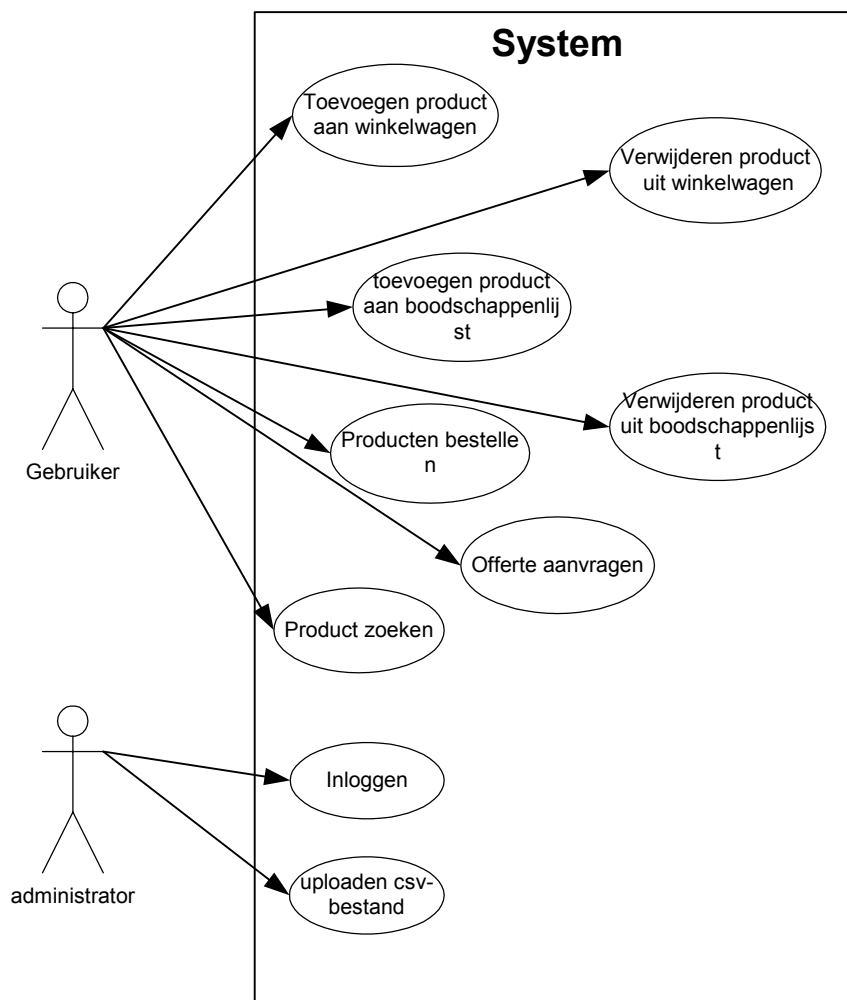
7.3.1 Use case diagram

Een use case is een beschrijving van een bepaalde wijze waarop het systeem gebruikt kan worden en wordt gebaseerd op de functionele eisen.

In een use case speelt de actor een grote rol, een actor is een gebruiker van het systeem, dit kan een persoon zijn of een extern systeem, bij de webwinkel applicatie dus de gebruiker(s) en de administrator.

Uiteindelijk zijn de hier onderstaande use cases samengesteld.

Figuur 3, use case diagram volgens UML



Bij ieder use case diagram hoort ook een beschrijving in tekst in een standaard tabel, tabel 9 is een voorbeeld hiervan.

Tabel 9, use case beschrijving in tekst volgens Warmer en Kleppe, Praktisch UML tweede editie

Naam	Toevoegen product aan winkelwagen
Samenvatting	De gebruiker voegt een product toe aan de winkelwagen
Actoren	Gebruiker
Aannamen	Het product is leverbaar
Beschrijving	1. De gebruiker selecteert een product 2. De gebruiker stelt het te bestellen aantal in 3. De gebruiker klikt op product toevoegen 4. De webwinkel controleert of het product al in de winkelwagen zit, is dit het geval dan treed er een uitzondering op 5. De webwinkel voegt het product en het te bestellen aantal toe aan de winkelwagen, is dit het eerste product dan treed er een uitzondering op 6. De webwinkel berekent de verzendkosten 7. De webwinkel berekent het totaal gewicht, als dit groter is dan 30kg dan treed er een uitzondering op. 8. De webwinkel berekent de totale kosten
Uitzonderingen	1. [product al toegevoegd] Als het product zich al in de winkelwagen bevindt dan wordt het aantal van het product met het ingestelde aantal opgehoogd. 2. [eerste product] Als dit het eerste product is in de winkelwagen dan moet de gebruiker aangeven in welke regio hij zich bevindt en welke wijze van verzenden hij wilt gebruiken. 3. [Maximum gewicht overschreden] Als het totaal gewicht groter is dan 30kg dan krijgt de gebruiker een melding dat het maximum gewicht is overschreden en dat hij producten kan verwijderen of een offerte kan aanmaken voor grote bestellingen.
Resultaat	Het product en het product aantal zijn toegevoegd aan de winkelwagen en de verzendkosten, totaal gewicht en de totale kosten zijn berekend.

Op deze wijze is een duidelijk overzicht verkregen van de functionele eisen, de volgende stap in de UML modelleringstechniek is de vertaling naar een sequence diagram.

7.4 Niet-functionele eisen

Niet functionele eisen zijn eisen die gesteld worden aan het product zonder specifiek in te gaan op de functionaliteit van het systeem zelf. Deze eisen vinden hun oorsprong meestal bij het gebruikersgemak, de vormgeving en het platform waarop de applicatie komt te draaien. Na de Joint Application Design sessie zijn de volgende niet-functionele eisen opgesteld:

- Platform
 - Platform onafhankelijk
- Server
 - Tomcat Server
- Programmeertalen
 - Java
 - MySQL
 - HTML
 - Code moet herbruikbaar, onderhoudbaar en leesbaar zijn
 - Struts framework

Veel van de niet-functionele eisen zijn afgeleid van de functionele eis dat de webwinkel generiek moet zijn. Vandaar dat er programmeertalen en platformen zijn gekozen die dit mogelijk maken. Tevens is er gekozen om de applicatie platform onafhankelijk te maken om de afnemers van de webwinkel de vrijheid te geven zelf te bepalen in welke omgeving de webwinkel moet draaien, zodat een hogere klanttevredenheid bereikt wordt. Belangrijk is tevens dat de code herbruikbaar, onderhoudbaar en leesbaar moet zijn, dit omdat er in de toekomst andere programmeurs onderhoud zullen plegen en verder bouwen aan de webwinkel applicatie.

7.5 Initieel Klassendiagram

Officieel wordt het initieel klassendiagram gemaakt op basis van gevonden klassen, attributen en operaties uit de probleemstelling of opdrachtschrijving. Er is echter voor gekozen om aan de hand van de scope, de use cases en mijn ervaring op het gebied van systeemontwikkeling het klassendiagram samen te stellen. Dit is gedaan omdat de ervaring uitwijst dat een initieel klassendiagram gemaakt volgens de officiële wijze vrijwel altijd weinig nuttig is. Dat komt omdat het erg afhankelijk is van termen die toevallig in de probleemstelling of opdrachtschrijving genoemd worden. Het klassendiagram dat op basis van de scope samengesteld is bevat daarentegen wel al de functionele en niet-functionele eisen en is daarom heel wat bruikbaar. Na inventarisatie zijn de volgende klassen, attributen en operaties geïdentificeerd die weergegeven zijn in tabel 10.

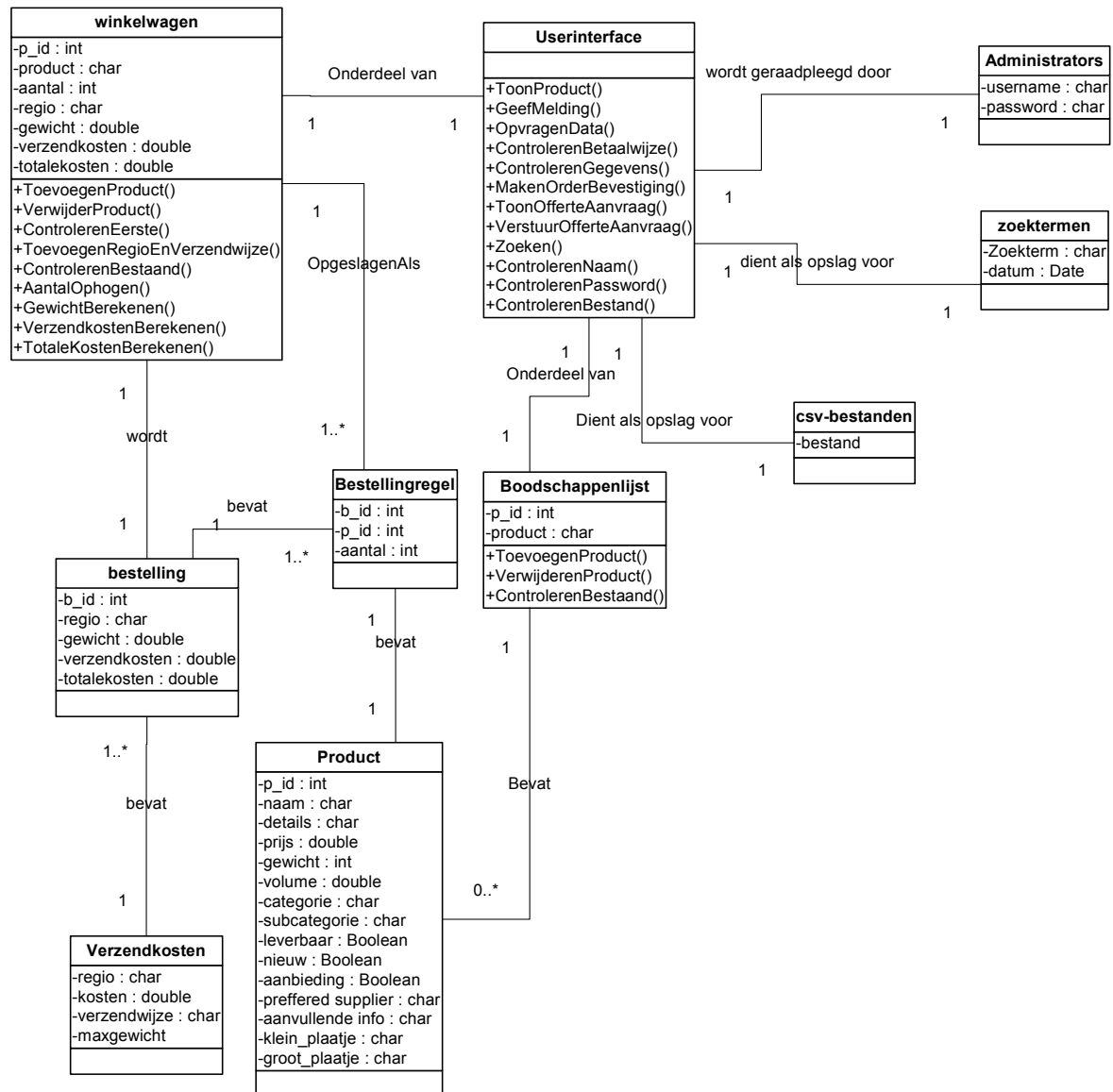
Tabel 10, Overzicht van gevonden klassen en attributen voor het initieel klassen diagram

Klasse	Beschrijving	Attributen	Operaties
Administrators	Een tabel die de inlog informatie bevat van de administrators	Username Password	-
Bestelling	Een tabel die bestellingen bevat van de klanten (gebruikers) van de webwinkel	B_id Regio Gewicht Verzendkosten Totale kosten	-
Bestellingregel	Een tabel die de producten bevat die besteld zijn.	B_id P_id Aantal	-
Boodschappenlijst	Een lijst van favoriete producten van een gebruiker	P_id Product	Toevoegen product Verwijderen bestaand Controleren bestaand
Csv-bestanden	Een locatie waar csv-bestanden naartoe geupload worden	Bestand	-
Product	Een tabel die alle productinformatie bevat	P_id Naam Details Prijs Gewicht Volume Categorie Subcategorie Leverbaar Nieuw Aanbieding Preferred_supplier Aanvullende info Klein plaatje Groot plaatje	-
Userinterface	Een scherm waarop informatie getoond kan worden	-	Toon product Geef melding Opvragen data Controleren betaalwijze Controleren gegevens Maken_orderbevestiging Toon offerte aanvraag Verstuur_offerte_aanvraag Zoeken Controleren naam Controleren password Controleren bestand
Winkelwagen	Een scherm dat alle producten bevat die door de gebruiker geselecteerd zijn om te bestellen	P_id Product Aantal Regio Gewicht Verzendkosten Totale kosten	Toevoegen product Verwijder product Controleren eerste Toevoegen regio en verzendwijze Controleren bestaand Aantal ophogen Gewicht berekenen Verzendkosten berekenen Totale kosten berekenen
Zoektermen	Een tabel die zoektermen bevat waarop geen resultaat kwam	Zoekterm Datum	-

De klassen winkelwagen, boodschappenlijst en userinterface uit de tabel, representeren de functionele werking van de webwinkel applicatie, de overige klassen representeren database tabellen.

Uiteindelijk is het klassendiagram samengesteld dat weergegeven is in figuur 4.

Figuur 4 initieel klassendiagram

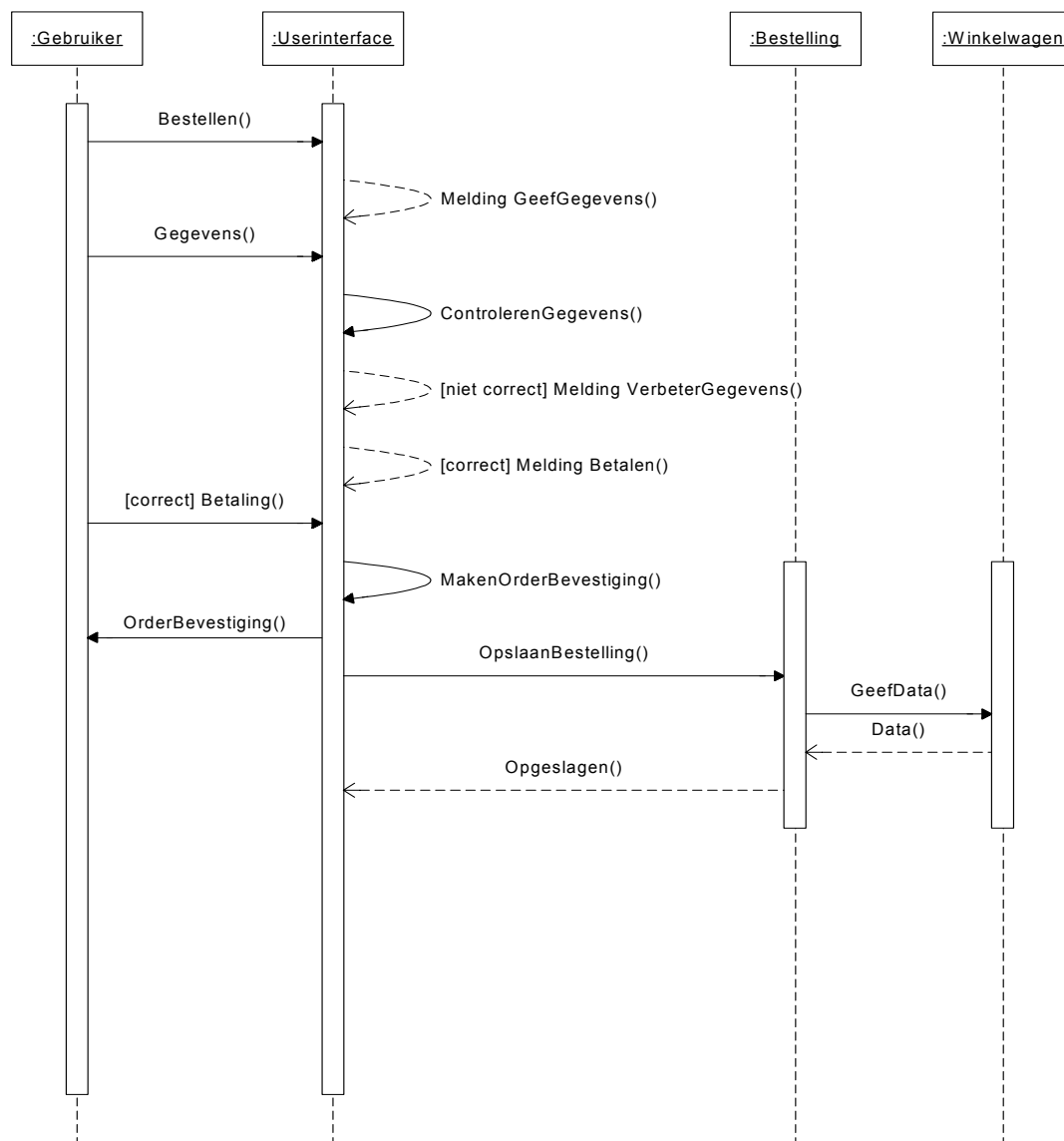


7.6 Sequence diagrammen

Na het maken van het initieel klassendiagram is verder gegaan met het maken van de sequence diagrammen. Sequence diagrammen nemen de use cases als uitgangspunt en leggen informatie vast over het interne gedrag van het systeem.

Besloten is de sequence diagrammen te maken om eventueel over het hoofd geziene functionaliteiten en gedragingen van het systeem op te sporen en op te nemen in het uiteindelijke complete klassendiagram waarop de webwinkel applicatie gebaseerd wordt. In figuur 5 wordt als voorbeeld het sequence diagram van bestellen getoond.

Figuur 5, Sequence diagram van bestellen

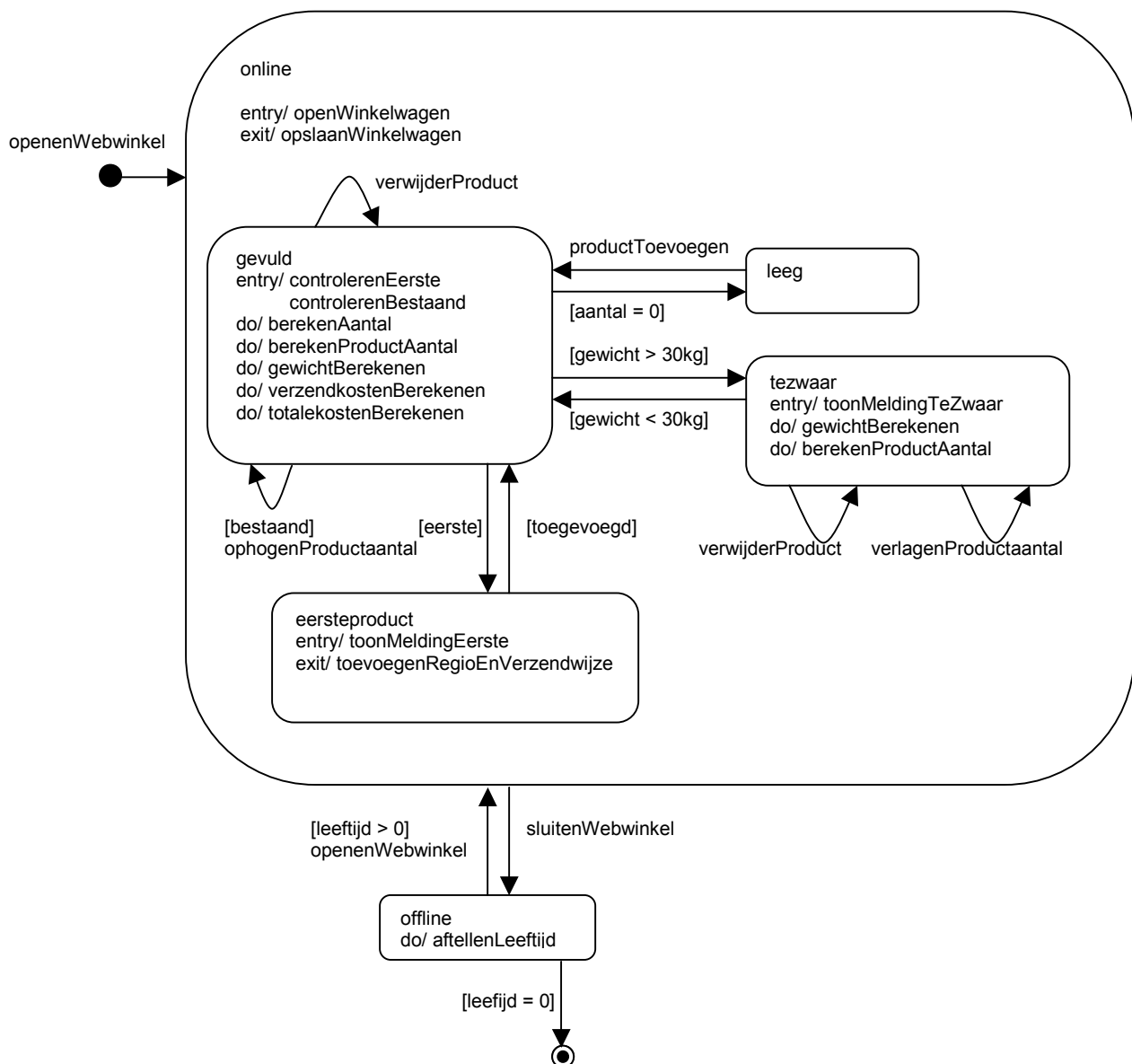


Dankzij de sequence diagrammen werd duidelijk dat verschillende associaties in het initieel klassendiagram verkeerd benoemd waren en anders hoorden te lopen dan voorheen gedacht. De volgende stap in UML is het modelleren van de toestandsdiagrammen.

7.7 Toestandsdiagrammen

De toestandsdiagrammen nemen de sequence diagrammen als basis en tonen in welke toestanden een object zich kan bevinden.
De reden om een toestandsdiagram te maken is om een duidelijk beeld van de werking van het te bouwen systeem te krijgen.

Figuur 6 Toestandsdiagram klasse winkelwagen

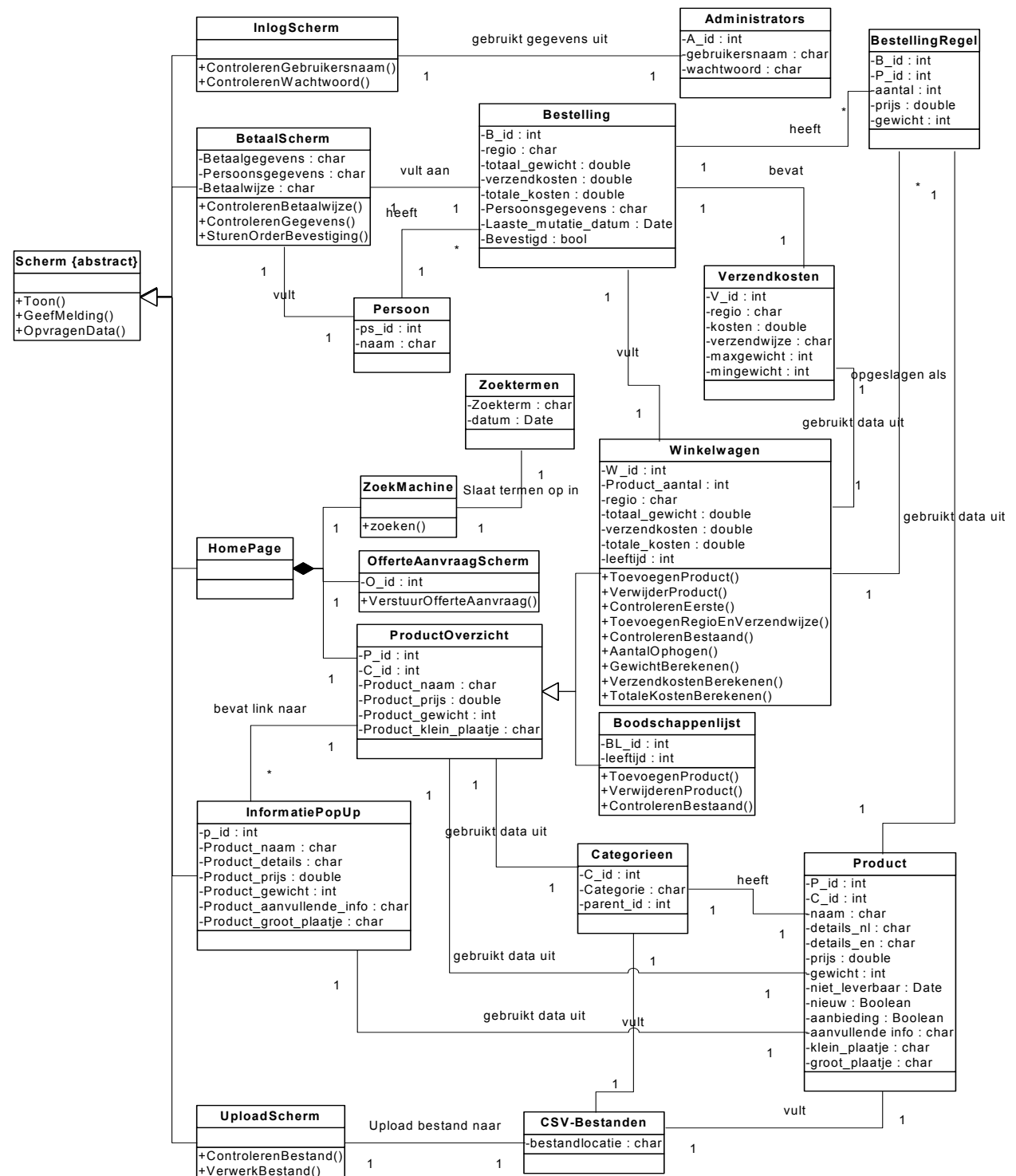


Uit de toestandsdiagrammen werd duidelijk dat het gebruik van slechts één userinterface in het initieel klassendiagram onwerkbaar is. Deze conclusie is getrokken omdat één userinterface nooit goed in staat zal zijn elke toestand van de webwinkel juist weer te geven, aangezien de webwinkel zich geregeld in meer dan één (sub)toestand bevindt.

7.8 Compleet klassendiagram

De laatste stap in het UML traject is het samenstellen van het complete klassendiagram. Het complete klassendiagram wordt gebaseerd op het initieel klassendiagram en de informatie die gevonden is met behulp van de in de sequence,- en toestandsdiagrammen. De webwinkel applicatie wordt gebaseerd op het compleet klassendiagram, in figuur 7.

Figuur 7, compleet klassendiagram



7.9 Compleet klassendiagram met struts.

Na het afronden van het compleet klassendiagram, kwam het programmeren aanbod, voorafgegaan door een literatuurstudie naar de werking van het struts framework. In de literatuur over struts werd vermeld dat het ontworpen model van een applicatie zonder al veel problemen overgezet kan worden in de modellaag van het MVC-model. In de praktijk bleek dit echter niet het geval. Dat komt met name omdat het systeemgedrag en samenstelling zoals die in het compleet klassendiagram zijn gemodelleerd niet terug komen in de modellaag. Het enige wat wel terugkomt zijn de functionaliteiten. Tevens bood de literatuur over het struts framework over het algemeen een zeer eenzijdige kijk op ontwikkelen in het struts framework. De literatuur was vrijwel altijd geschreven voor een publiek dat bestaat uit ervaren programmeurs die al vaker met het MVC-model hebben gewerkt. Daardoor beperkte de uitleg in de literatuur zich vrijwel altijd tot het direct programmeren in het struts framework en liet het vaak een goed overzicht van het framework achterwege.

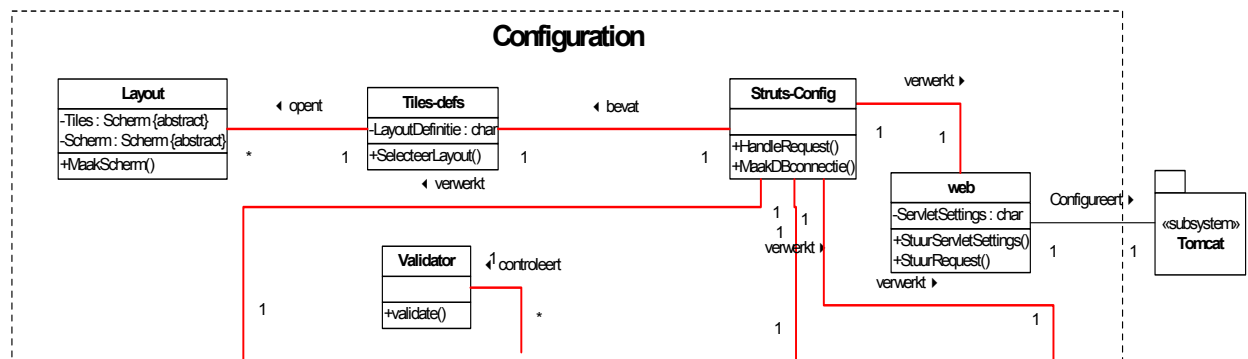
Vanwege de boven genoemde redenen is besloten een nieuw klassendiagram samen te stellen dat niet alleen het model van de te ontwikkelen webwinkel applicatie bevat maar ook de klassen van het struts framework waarin het model zich bevindt.

Het was helaas niet eenvoudig om dit nieuwe klassendiagram samen te stellen. Dat kwam met name omdat de literatuur te wensen overliet en omdat het MVC-model waar struts gebruik van maakt een theorie is die in de praktijk nogal verschillende invullingen krijgt. Het uiteindelijke klassendiagram met struts is daarom een weergave van de webwinkel applicatie binnen het struts framework naar mijn eigen inzicht in de werking van het MVC-model. Aangezien dit klassendiagram nogal omvangrijk is geworden is het per laag opgesplitst. In de volgende paragrafen zal ik de lagen behandelen. Aangezien het struts framework een applicatie van beneden naar boven bestuurd zal eerst onderste laag behandeld worden en zal de bovenste laag als laatste aan de orde komen. Dit alles om het klassendiagram en de werking van de webwinkel applicatie zo duidelijk mogelijk te maken. Voor het gehele klassendiagram verwijs ik u naar de beheerderhandleiding in de bijlage.

7.9.1 Configuratielaag

De configuratielaag is geen officiële laag uit het MVC-model, maar is wel een wezenlijk onderdeel van struts. De laag bestaat uit naar XML geconverteerde klassen die het struts framework aansturen en horen officieel in de controller laag thuis. De reden dat ze toch in een aparte laag zijn gezet is om de cruciale rol die klassen spelen in het framework te benadrukken. In figuur 8 staat de configuratielaag weergegeven.

Figuur 8, configuratielaag



De belangrijkste klasse in deze laag is de struts-config klasse.

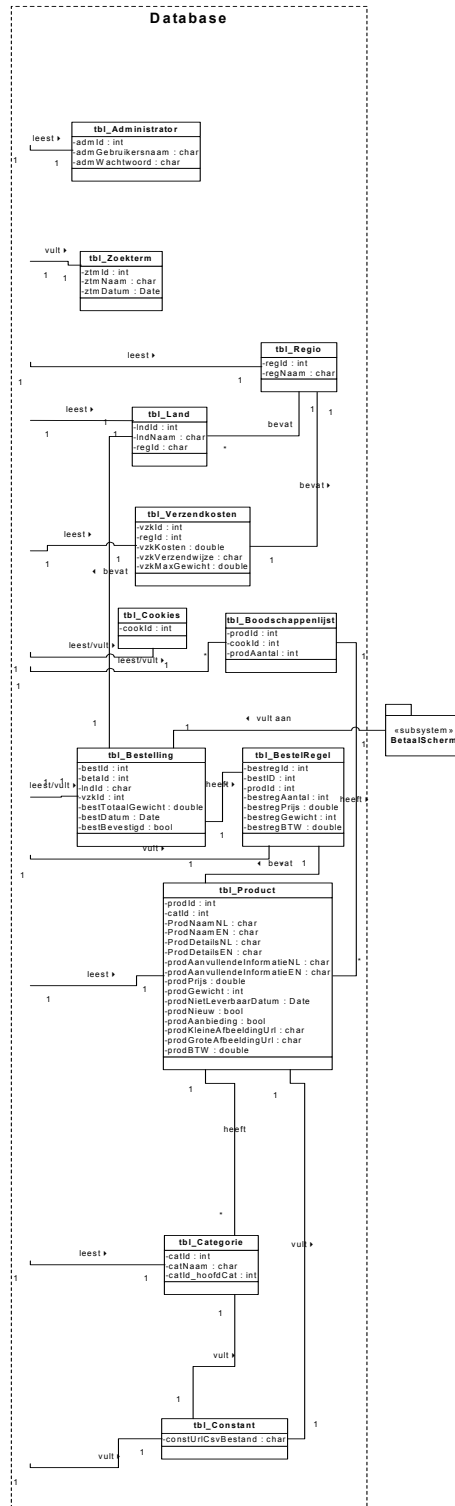
Deze klasse stuurt alle opdrachten (request) en applicatie reacties (responses) naar de juiste klasse en zorgt er zo voor dat de juiste functionaliteit wordt uitgevoerd bij een opdracht.

De web klasse dient voor de configuratie van de gekozen server. De lay-out en tiles-defs klassen zorgen voor de juiste scherm opbouw. De validator klassen zorgt tenslotte voor de validering van ingevoerde data. Gezien de sleutel rol die deze klassen spelen zijn dit de eerste klassen die men implementeert.

7.9.2 Databaselaag

De databaselaag bevat alle klassen die database tabellen representeren. Deze laag staat alleen in contact met de modellaag. De modellaag is de enige laag van de applicatie waar met database informatie gewerkt mag worden. Deze laag vormt de basis voor het ERD-diagram. In figuur 9 is de databaselaag weergegeven.

Figuur 9, databaselaag

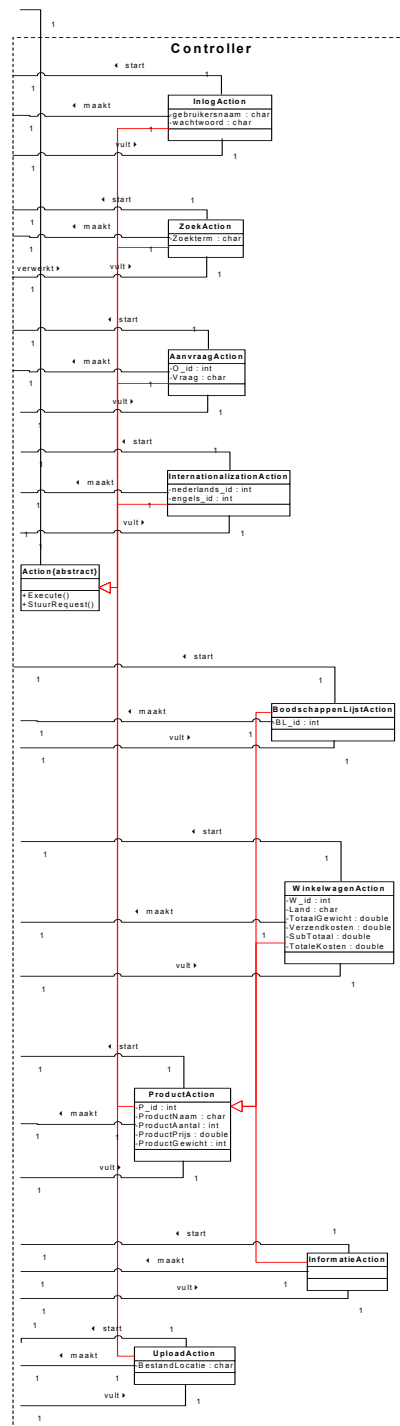


7.9.3 controllerlaag

De controllerlaag is verantwoordelijk voor het aansturen van de webwinkel applicatie. De klassen in deze laag ontvangen de opdrachten (request) en applicatie reacties (responses) van de struts-config klasse. De klassen bepalen vervolgens wat de functie in de modellaag moet uitvoeren.

Als de functie in de modellaag klaar is met zijn taak dan genereert de klasse in de controllerlaag een applicatie reactie (response) en koppelt daar eventueel mee te zenden data aan. Figuur 10 geeft een overzicht van de controllerlaag.

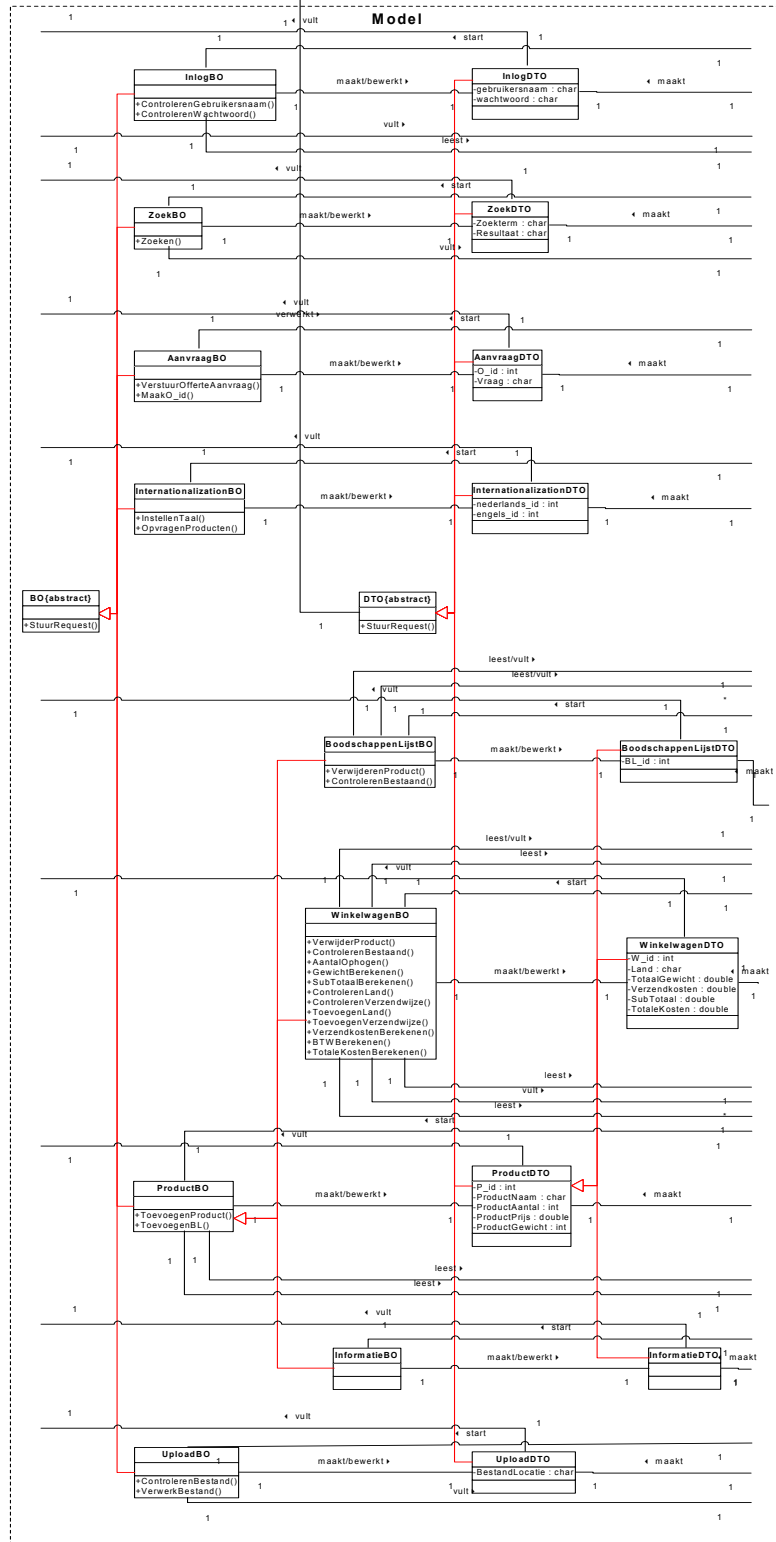
Figuur 10, de controllerlaag



7.9.4 Modellaag

De modellaag bevat de daadwerkelijke functionaliteiten van de applicatie. De laag wordt volledig aangestuurd door de controllerlaag en is de enige laag die contact maakt met de database. De klassen in deze laag voeren hun taak uit en vullen een data transfer object (dto) na afloop, dat doorgestuurd wordt naar de controllerlaag.

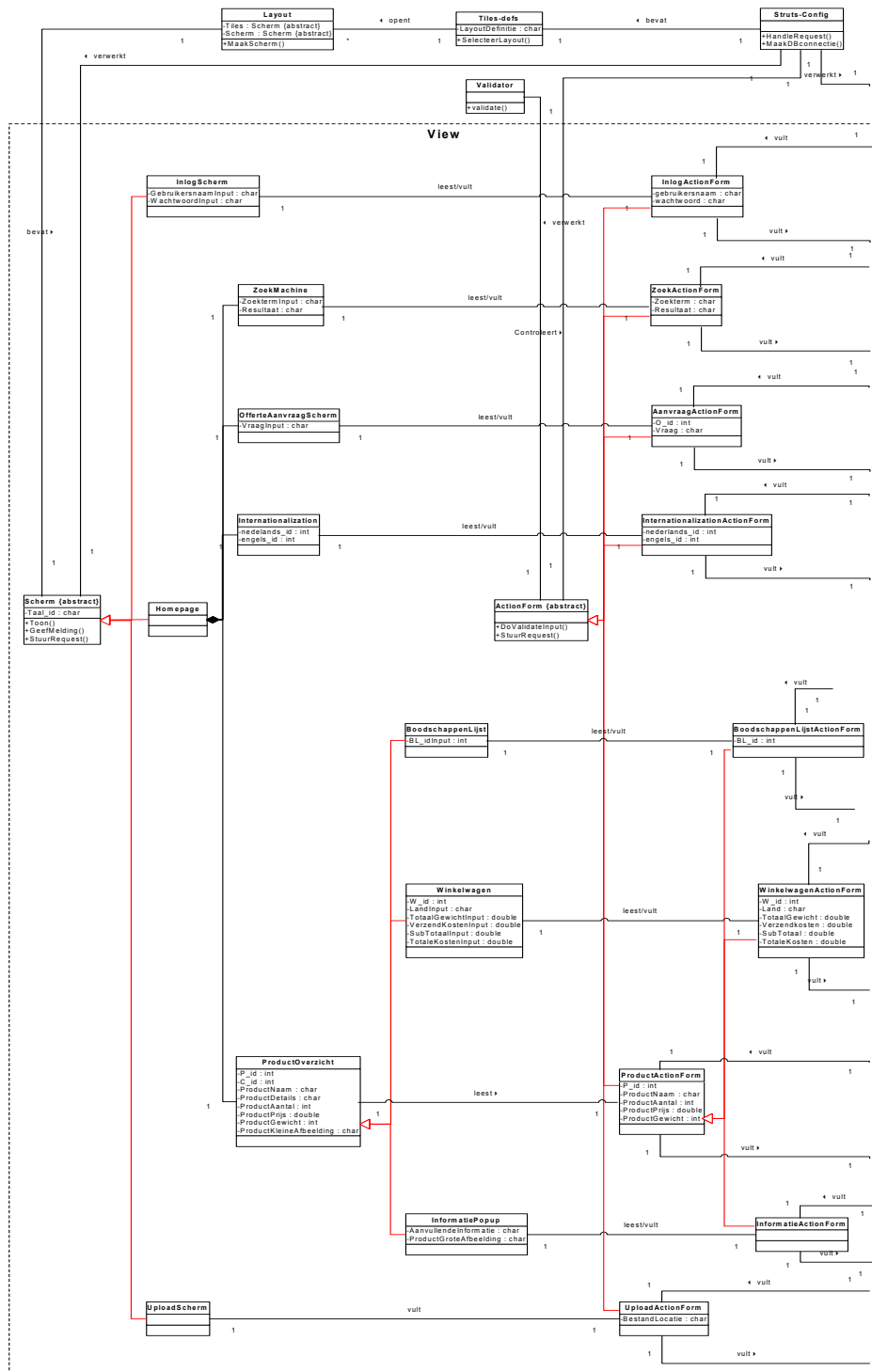
Figuur 11, de modellaag



7.9.5 Viewlaag

De viewlaag bevat naast alle schermen ook een bijzonder type data transfer object (dto) namelijk een actionform. Het actionform is een klasse die alle waarden van een bepaald htmlform bevat. Als een htmlform gesubmit wordt dan worden de waardes uit het htmlform gekopieerd naar het actionform. Vervolgens worden de waarden in het actionform gecontroleerd op ongeldige invoer. Als de waarden geldig zijn dan wordt het actionform doorgestuurd naar de controllerlaag, zo niet dan verschijnt er een melding op het scherm dat er geldige waarden ingevuld moeten worden.

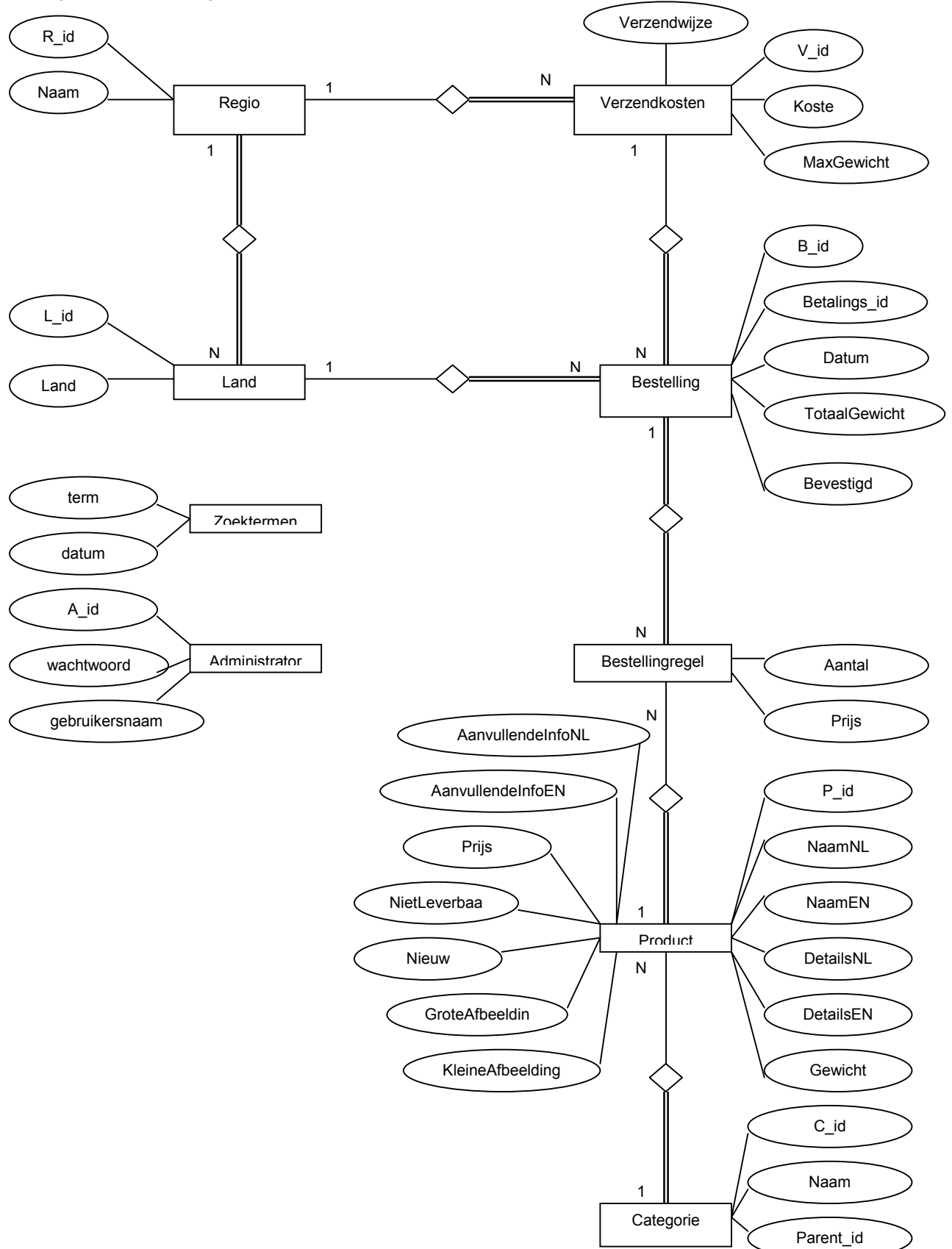
Figuur 12, de viewlaag



7.10 ERD diagram

MySQL is gekozen om de database in te bouwen. Aangezien MySQL een relationele database is en de ontwerpdocumentatie tot nog toe alleen van object oriëntatie uitging is er een ERD samengesteld om de vertaalslag van compleet klassendiagram naar de database eenvoudiger te maken. Het ERD is in figuur 13 weergegeven.

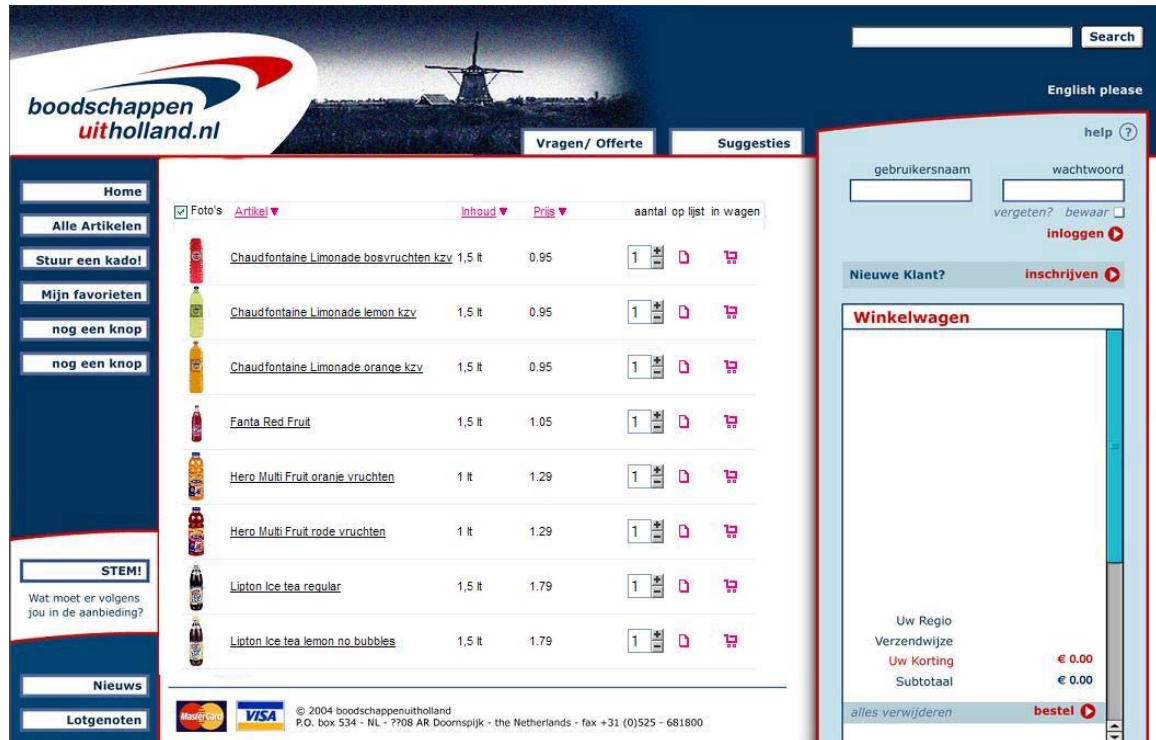
Figuur 13, Het ERD diagram



7.11 Schermontwerp

Om alle betrokken partijen een duidelijk beeld te geven van de schermen te ontwikkelen webwinkel applicatie is er een scherm gemaakt door grafisch ontwerper. Het scherm staat in figuur 14 weergegeven, voor overige schermen wordt u verwezen naar de bijlage functioneel ontwerp.

Figuur 14, schermontwerp webwinkel applicatie



Deel 3

—

Implementatie

8.0 Construction

8.1 Inleiding

Het doel van de fase Construction is het ontwikkelen van de applicatie. In deze fase wordt het systeem gebouwd op basis van het model uit de User Model fase. Vervolgens wordt de applicatie getest en goed,- of afgekeurd.

8.2 Ontwikkeling van de webwinkel applicatie

De RAD-methode schrijft voor dat in deze fase begonnen wordt met de bouw van de kernfunctionaliteiten. Dit zijn de functionaliteiten met de hoogste prioriteit. Een overzicht van de functionaliteiten en hun prioriteiten is gegeven in tabel 8. Als eenmaal de kernfunctionaliteiten zijn gebouwd wordt er overgegaan tot functionaliteiten met een lagere prioriteit.

Deze werkwijze hield voor het webwinkel applicatie project in dat eerst de database gebouwd werd. Tijdens de bouw van de database is er rekening gehouden met de naamgeving die bij ASolutions wordt aangehouden. Om die reden beginnen alle tabellen met de prefix tbl_ en hebben alle tabel attributen een prefix die begint met een afkorting van de tabelnaam. Na het bouwen van de database is er verder gegaan met het programmeren van de configuratielaag van het struts framework.

Deze laag bepaald in grote lijnen het gedrag van het systeem, om die reden is deze laag iedere keer aangepast toen in de andere lagen duidelijk werd dat een functionaliteit toch net iets anders verloopt of weergegeven moest worden dan voorheen aangenomen was.

Toen eenmaal de configuratielaag zo ver gevorderd was dat er een applicatie op kon draaien is er overgegaan tot het bouwen van de controller,- en modellaag.

Deze lagen bevatten de functionaliteiten van de applicatie.

Nadat de controller,- en modellaag af waren, werd er overgegaan tot de bouw van de viewlaag. De view laag is de laag die alle schermen bevat. Verrassend genoeg gaf juist deze laag de meeste problemen. Dit kwam omdat tweetaligheid, validatie ,dubbel submit voorkoming en cookie,- en sessieafhandeling allemaal in deze laag plaats vinden wat de opbouw van een view niet eenvoudiger maakte.

8.3 Problemen

Het voornaamste probleem dat in deze fase tegengekomen is, was dat duidelijk merkbaar was dat verschillende onderdelen van struts gebouwd zijn door verschillende programmeurs met elk een andere kijk op het MVC-model. Dit kwam vooral tot uiting in koppelingen maken tussen de verschillende modules. De ene module wilde de informatie in een vorm ontvangen of leveren die een andere module net niet accepteerde. Dit is wellicht een gevolg van de relatief jonge leeftijd van het struts framework.

Een ander probleem dat zich voordeed was dat de grafische interfaces die geleverd werden door de grafisch ontwerper niet volledig voldeden en aanpassingen vereisten. Daardoor is de nodige uitloop ontstaan. Er is geprobeerd de grafisch ontwerper te bereiken per e-mail en telefoon, helaas liet de grafisch ontwerper pas na enkele maanden weer iets van zich horen. Daarom is een deel van de lay-out toch door de afstudeerder gemaakt terwijl dit niet zo was afgesproken.

8.4 Testen

Tijdens de bouw van de webwinkel applicatie is er doorlopend getest.

Nadat er eenmaal een applicatie gebouwd was die leek te voldoen aan alle systeemeisen is er overgegaan tot een algemene systeemtest waarbij alle functionaliteiten tegen het licht gehouden zijn.

Dit is gebeurd met behulp van een testset, voor een overzicht van deze testset word u verwijzen naar de bijlage testsets en testresultaten.

Uit de test kwam naar voren dat enkele views nog niet klopten, deze zijn daarna aangepast en nogmaals getest totdat ze het gewenste resultaat opleverden.

9. Cut Over

In de Cut over fase van RAD vindt de overdracht van de applicatie plaats.

Daarnaast wordt de applicatie gedocumenteerd.

Voor het webwinkel applicatie project is de documentatie in de vorm van een beheerderhandleiding uitgevoerd. Hierin wordt het model van de applicatie beschreven en wordt de product uploadfunctie nader beschreven aangezien daar om gevraagd was door de opdrachtgever. De beheerderhandleiding is opgenomen in de bijlagen.

De daadwerkelijke overdracht van de applicatie heeft nog niet plaats gevonden omdat de opdrachtgever nog geen provider heeft gekozen, waar de applicatie komt te draaien.

Deel 4

—

Evaluatie

10. Evaluatie

Het afstuderen heeft niet alleen het behalen van een diploma als doel, de afstudeerder is van mening dat je hier ook ervaring van op doet en dus er ook iets van leert. Gedurende het leerproces zijn het maken van fouten natuurlijk. De uitdaging ligt hem dan ook in het leren van die fouten en het goed ermee omgaan.

Dit hoofdstuk onderkend de fouten die zijn gemaakt en beschrijft een toekomstige alternatieve werkwijze om deze fouten te voorkomen. In dit hoofdstuk worden de producten en de procesgang zo objectiefmogelijk beschreven om ze een goed beeld te creëren van de problemen.

10.1 Procesevaluatie

Voor het evalueren van procesgang is een verdeling gemaakt in een aantal paragrafen. Als eerste wordt de gebruikte methode per fase geëvalueerd gevolgd door de gebruikte techniek. Ten slotte wordt de communicatie met de opdrachtgever geëvalueerd en wordt de theorie tegen de praktijk afgezet.

10.1.1 Gebruikte methode

Zoals eerder is beschreven is de methode RAD gebruikt.

Ik ben zeer te spreken over RAD. Deze methode heeft voor een duidelijke leidraad gezorgd gedurende het project.

Ik ben van mening dat deze methode zeer geschikt is voor relatief kleine projecten van een halfjaar of minder.

Dankzij de opzet van RAD ben ik instaat geweest de benodigde informatie om de webwinkel applicatie te ontwikkelen, boven water te krijgen. Daarnaast is RAD heel geschikt om snel in te springen op veranderende wensen en eisen van de opdrachtgevers. Ik zal in de volgende paragrafen per RAD fase evalueren wat goed en minder goed is gegaan.

10.1.2 Requirementsplanning

De requirementsplanning fase is goed doorlopen.

De timebox planning zoals die gemaakt was in het plan van aanpak (hoofdstuk 4) is ruim gehaald. Ik ben er in deze fase succesvol in geslaagd alle functionele eisen boven water te krijgen.

10.1.3 User Design

De user design fase is vlot verlopen, mede dankzij mijn keuze voor UML als modelleringstechniek. Het UML-traject past namelijk naadloos in deze fase.

Ondanks het vlotte verloop is deze fase toch uitgelopen, dit kwam omdat het compleet klassendiagram niet geheel bruikbaar was in het struts framework.

In het vervolg zal ik me eerder verdiepen in de mogelijkheden van de programmeertalen en programmeeromgevingen voordat ik ga modelleren, zodat de opgeleverde modellen beter zullen aansluiten.

Vanwege de uitloop is de timebox planning niet gehaald en liep ik een achterstand van ruim twee weken op.

10.1.4 Construction

De Construction fase is redelijk verlopen.

De beperkte documentatie van het struts framework was de grootste hindernis tijdens het doorlopen van deze fase.

Ik zal een voorbeeld geven van een probleem waar ik tegenaan liep door de beperkte documentatie van het struts framework.

Het struts framework bevat een heleboel meegeleverde klassen bevat die heel “eenvoudig” functionaliteiten zoals bijvoorbeeld validering uitvoeren. In de praktijk betekende dat, dat validering via één regel code gebeurde. De documentatie bevatte vaak wel een uitleg hoe die ingepast moest worden in de rest van de code, maar als die wijze van inpassen niet werkte was nergens terug te vinden waaraan dat kon liggen of hoe het ook anders kan.

Om die reden zal ik bij een volgend project eerder kiezen voor een softwareomgeving waar meer documentatie voor beschikbaar is.

Een ander probleem dat zich voordeed in deze fase was de gebrekkige communicatie met de grafisch ontwerper. De grafisch ontwerper leverde de afbeeldingen voor de webwinkel aan in een ai-formaat (Adobe Illustrator bestand). Een ai-formaat bestand bestaat uit een raamwerk van verschillende afbeeldingen die nog losgeknippt moeten worden.

Er was echter afgesproken dat de afbeeldingen in .jpg-formaat zouden worden aangeleverd. Daarnaast waren de afbeeldingen alleen in het Nederlands gemaakt terwijl afgesproken was dat ook Engelse afbeeldingen gemaakt zouden worden.

Ik heb veel e-mails gestuurd naar de grafisch ontwerper en enkele telefoontjes gepleegd om te vragen of de afbeeldingen in orde gemaakt konden worden.

Ik kreeg de grafisch ontwerper echter niet telefonisch te pakken en pas na enkele maanden werd er gereageerd op de e-mails. Dit leidde ertoe dat ik wat de lay-out betrof de moed enigszins had opgegeven dat die nog op tijd af zou zijn.

Ik had de moed echter niet zo snel moeten opgeven, want toen na enkele maanden toch een reactie kwam bleek dat de uitloop toch minder was dan ik verwacht had waardoor de gestelde deadline weer reëler werd. Dat de gestelde deadline toch niet gehaald werd wijt ik daarom aan het te vroeg opgeven van de moed uit onervarenheid en het te veel zelf willen doen waardoor ik mijn stagebegeleider niet tijdig op de hoogte stelde van dit probleem.

Ik zal in het vervolg beter rekening moeten houden met de neiging tot te veel zelf willen doen en ik zal moeten proberen via meerdere kanalen te communiceren.

Door beide problemen is de timebox planning voor deze fase wederom niet gehaald.

Ik ben mij daarom tijdens deze fase zorgen gaan maken of het project wel afgerond kon worden binnen de gestelde tijd. Daarom is er besloten de aandacht vooral te richten op de kernfunctionaliteiten van het programma.

10.1.5 Cut Over

Zoals vermeld in hoofdstuk 11 is deze fase slechts deel uitgevoerd, omdat de opdrachtgever nog geen provider heeft. Daarom zal ik in de toekomst er voor zorgen dat ik eerder duidelijk maak aan de opdrachtgever wat er van hem/haar verlangd wordt bij de verschillende fases van het project.

10.1.6 Gebruikte techniek

Als Modellingstechniek heb ik voor UML gekozen.

Dit is prima bevallen, met name omdat het goed aansluit bij de user design fase in RAD.

Ik heb de UML standaardmodellen met name gebruikt om de functionele eisen te bepalen en een basis te vormen van waaruit geprogrammeerd kon worden.

UML biedt een uitgebreide keur aan modellen, waarvan ik lang niet alle modellen gemaakt heb. De kunst was dan ook het vinden van de juiste modellen bij die gemaakt dienden te worden ten einde een goed ontwerp te kunnen maken.

Ik ben van mening dat dit goed gelukt is.

Bij het modelleren van de diagrammen is gebruikt gemaakt van Microsoft Visio. Dit programma sluit goed aan bij de techniek en zullen we in de toekomst zeker nog eens gebruiken.

10.1.7 Communicatie binnen het project

De communicatie met de opdrachtgever verliep goed. Dit is grotendeels te danken aan het feit dat er zowel via bijeenkomsten, de telefoon en via e-mail werd gecommuniceerd. Alleen is de communicatie met de grafisch ontwerper minder geslaagd geweest, zoals is uit gelegd in paragraaf 10.1.3.

Hoewel communicatie over het algemeen goed verliep zijn er wel enkele punten welke ik in de toekomst anders zou aanpakken. Zo zal ik in een eerder stadium in een project proberen te anticiperen op met wie en wanneer afspraken nodig zijn. Dit heeft twee voordelen: Je raakt eerder bekend met de omgeving van het project en mogelijke knelpunten worden eerder achterhaald en voorkomen.

10.1.8 Theorie en praktijk

Een van de leerelementen is het verschil tussen theorie en praktijk op communicatief niveau. In de praktijk ben ik veelal geconfronteerd met het feit dat een product het resultaat is van afstemmingsmomenten met meerdere partijen. In de theorie wordt op school weinig aandacht besteed aan communicatie omtrent een project, er wordt meer aandacht besteed aan waaruit bijvoorbeeld een functioneel ontwerp dient te bestaan.

Ik heb het idee gekregen dat in de praktijk de communicatie een sleutelrol inneemt bij het tot een succesvol kunnen afronden van een project.

Het opvallendste verschil tussen de theorie en de praktijk is de wijze waarop met methoden en rapporten wordt omgegaan. In de theorie is het zo dat de werkwijze die de gekozen methode voorschrijft strak aangehouden dient te worden.

In de praktijk bleek echter dat men hier nauwelijks waarde aanhecht, zolang maar duidelijk wordt wat en hoe er iets gedaan gaat worden.

10.2 Product Evaluatie

Voor het evalueren van de producten is zo objectief mogelijk gekeken naar de verschillende aan de opdrachtgever opgeleverde documenten en applicatie. Tijdens de evaluatie is vooral gekeken of het product nut heeft gehad voor de opdrachtgever en of de probleemstelling is opgelost.

10.2.1 Scope

De scope heeft als doel gehad snel een duidelijk beeld te verschaffen van de afbakening en de overeengekomen systeemeisen van de webwinkel applicatie.

Ik denk dat de scope die ik heb samengesteld hier goed in is geslaagd.

Regelmatig is de scope over en weer gestuurd tussen mij en de opdrachtgever om onduidelijkheden weg te werken. Ik ben tevreden over de uiteindelijke vorm van de scope en zal in het een volgende keer net zo opbouwen.

De scope is opgenomen in de bijlagen.

10.2.2 Functioneel ontwerp

Het opgeleverde functioneel ontwerp is niet een functioneel ontwerp in de zin van het woord.

Het functioneel ontwerp dient als een contract tussen de opdrachtgever en ASolutions.

Daarom bevat het functioneel ontwerp niet alleen de functionele eisen maar ook schermontwerpen, platform afspraken etc.

Zodoende dient het functioneel vooral als een duidelijke afspraak tussen de opdrachtgever en de afstudeerder.

Ik denk dat het samengestelde functioneel ontwerp voldoet in deze eis.

Toen het functioneel ontwerp eenmaal goedgekeurd was door de opdrachtgever, zijn geen nieuwe systeemeisen meer naar voren gekomen. Ik kan dus concluderen dat het goed heeft gediend als verduidelijking van de afspraken voor alle partijen.

Ik zal in een toekomstig project een zelfde soort document maken, maar zal het dan geen functioneel ontwerp noemen, aangezien het dit niet precies is. Een betere naam zou bijvoorbeeld "applicatie eisen overeenkomst" kunnen zijn.

10.2.3 Beheerderhandleiding

De opdrachtgevers hebben aangegeven dat zij geen behoefte hadden aan een uitgebreide uitleg van de werking van de opgeleverde applicatie.

Om die reden heb ik de beheerderhandleiding beperkt gehouden tot het geven van een model van het systeem en een korte uitleg van enige lastige ingewikkeldere functies zoals het uploaden van het productenbestand.

Ik denk dat de beheerderhandleiding de eventuele vragen van de opdrachtgevers die zich voordoen bij het gebruik en onderhoud van het systeem zal beantwoorden.

10.2.4 Webwinkel applicatie

Over de webwinkel applicatie ben ik tevreden.

Dit met name omdat de opgeleverde applicatie de probleemstelling oplost.

Er is een webwinkel opgeleverd die aan alle functionele eisen voldoet en zodoende mag ik concluderen dat het project succesvol afgerond is.

Daarnaast is aan de belangrijke eis dat de applicatie generiek moet zijn voldaan.

Bij een nieuwe klant voor een webwinkel kunnen de database en de model,- en controllerlaag gewoon gekopieerd worden, alleen de viewlaag zal moeten worden aangepast.

Dit laatste vind ik dit wel een prestatie en ik ben er daarom best trots op.

Bij een toekomstig vergelijkbaar project zou ik de webwinkel op een vergelijkbare wijze bouwen. Ik zou echter wel de lay-out zelf ontwikkelen want dan sluit die makkelijker aan op de gebouwde functies.

Referenties

Tijdens het project is gebruik gemaakt van een aantal informatiebronnen. Deze zijn onderverdeeld in literatuur en digitale informatie.

Literatuur

Programming Jakarta Struts, C. Cavaness, O' Reilly & Associates 2003, 0-569-00328-5
Praktisch UML 2e editie, J. Warmer, A. Kleppe, Addison Wesley 2001, 90-430-0494-4
Java 2 Grand Cru, S. Holzner, EASY COMPUTING, 2001, 90-5167-434-1
Reader SO-07 Methoden en technieken van systeemontwikkeling 787

Digitale informatie

http://www.google.nl	Zoekmachine voornamelijk gebruikt voor RAD en struts informatie
http://www.sun.com	Java documentatie
http://struts.apache.org	Struts framework homepage
http://www.mysql.com	MySQL documentatie
http://j2ee.masslight.com/Chapter4.html	MVC-model en struts informatie

Bijlagen

Bijlage A Plan van Aanpak

Planning webwinkel ontwikkeltraject

Auteur(s):	Alexander Bouwens
Laatste wijziging:	24-05-2004
Versie:	0.1
Status:	Voorlopig

Inhoudsopgave:

Inhoudsopgave:	46
1. Inleiding	47
1.1 <i>Achtergrond</i>	6
1.2 <i>Probleemstelling</i>	47
1.3 <i>Afbakening</i>	6
1.4 <i>Doelstelling</i>	47
2. Aanpak	47
2.1 <i>Uitgangssituatie</i>	6
2.2 <i>Methoden en Technieken</i>	48
3. Producten	48
4. Planning	48
4.1 <i>Faseplanning en timeboxes</i>	48

1. Inleiding

1.1 Achtergrond

A Solutions BV produceert administratieve software op maat voor klanten uit het bedrijfsleven. Het is een klein familie bedrijf dat al enkele jaren actief is op deze markt. A Solutions BV doet haar best zo goed mogelijk in te springen op de vraag in de markt, verandert de vraag dan verandert A Solutions BV mee.

1.2 Probleemstelling

Om aan de vraag uit het bedrijfsleven te blijven voldoen wil A Solutions BV een generieke webwinkel ontwikkelen. Er is een potentiële klant bereid om de functionele eisen van de te bouwen webwinkel mede te bepalen. Op deze manier hoopt A Solutions BV haar klanten beter van dienst te zijn en het klantenbestand uit te breiden.

1.3 Afbakening

In de Requirementsplanning fase zijn in overleg met de potentiële klant en medewerkers van A Solutions BV de functionele en technische eisen vastgesteld waaraan de webwinkel dient te voldoen.

Deze eisen zijn vastgelegd in de scope.

De uiteindelijke webwinkel zal op basis van deze eisen worden ontwikkeld, aanvullende eisen zullen niet in het ontwerp worden opgenomen.

1.4 Doelstelling

Doelstelling van de opdracht:

De doelstelling van de afstudeeropdracht is het ontwikkelen van een generieke webwinkelapplicatie, waarbij de interne werking van groter belang is dan het uiterlijk. Men zou graag zien dat deze applicatie aan de systeem eisen voldoet die in de scope beschreven staan.

2. Aanpak

2.1 Uitgangssituatie

- a. Benodigde software
 - Windows 2000/2003
 - Office 2003
 - Eclipse
 - Java 2
 - MySql
 - Tomcat server
- b. Benodigde hardware
 - Desktop pc
 - Server
 - Internet aansluiting
- c. Beschikbare rapporten
 - Businessplan met functionele eisen.

2.2 Methoden en Technieken

RAD (Rapid Application Development) zal worden gebruikt als projectbeheersingsmethode. Deze methode maakt gebruik van timeboxing en incrementeel ontwikkelen.

UML zal worden gebruikt als modelleringstechniek.

De volgende diagrammen zullen in ieder geval worden gemodelleerd:

- Use case diagram
- Klassendiagram
- Sequence diagram

3. Producten

- Plan van Aanpak
- Webwinkel applicatie
- Beheerders Handleiding webwinkel

4. Planning

4.1 Faseplanning en timeboxes

In de tabel hieronder staat de globale planning van de fases. Iedere fase is ingedetailleerder gedefinieerd in de timeboxes.

4.1.1 Faseplanning

Tabel 1, Timebox webwinkel project

ID	Task Name	Start	End	Duration	2004				
					Jun	Jul	Aug	Sep	
1	Requirementsplanning	17-5-2004	14-6-2004	4.20w					
2	User design	14-6-2004	21-7-2004	5.60w					
3	Construction	21-7-2004	16-9-2004	8.40w					
4	Cut over	16-9-2004	24-9-2004	1.40w					
5	Rapportage	24-9-2004	13-10-2004	2.80w					

De requirementsplanning fase van RAD is al van te voren ingepland omdat in deze fase de scope verkregen wordt waarop de timeboxes voor de overige fases gemaakt zullen worden. De planning van de requirementsplanning staat in tabel 3.

Tabel 2, Timebox Requirementsplanning fase

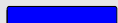
ID	Task Name	Start	End	Duration	May 2004		
					5-16	5-23	
1	Joint Requiremens Planning sessie	17-5-2004	21-5-2004	1w			
2	Scope	24-5-2004	14-6-2004	3.20w			

De timeboxes voor de overige fases van het project staan in tabel 4 t/m 7 weergegeven.

Tabel 3, Timebox User Design fase

ID	Task Name	Start	End	Duration	Jun 2004		Jul 2004	
							7-4	
1	Joint Application Design sessie	14-6-2004	18-6-2004	1w				
2	initieel klassediagram	18-6-2004	21-6-2004	0.40w				
3	Use Cases	21-6-2004	22-6-2004	0.40w				
4	Niet-functionele eisen	22-6-2004	22-6-2004	0.20w				
5	Sequence diagrammen	22-6-2004	24-6-2004	0.60w				
6	Toestandsdiagrammen	24-6-2004	30-6-2004	1w				
7	Compleet klassediagram	30-6-2004	6-7-2004	1w				
8	Schermontwerp	6-7-2004	8-7-2004	0.60w				
9	Prototype	8-7-2004	14-7-2004	1w				
10	Joint Application Design sessie	14-7-2004	21-7-2004	1.20w				

Tabel 4, Timebox Construction fase

ID	Task Name	Start	End	Duration	Jul 2004		Aug 2004					Sep 2004	
					7-25	8-1	8-8	8-15	8-22	8-29	9-5		
1	database laag	21-7-2004	23-7-2004	0.60w									
2	controller laag	23-7-2004	12-8-2004	3w									
3	model laag	12-8-2004	1-9-2004	3w									
4	view laag	1-9-2004	7-9-2004	1w									
5	creditcard betalingen	7-9-2004	8-9-2004	0.40w									
6	testen	8-9-2004	14-9-2004	1w									
7	user reviewboard	14-9-2004	16-9-2004	0.60w									

Tabel 5, Timebox Cut Over fase

ID	Task Name	Start	End	Duration	Sep 2004	
					9-12	9-19
1	Implementatie	16-9-2004	16-9-2004	0.20w		
2	Beheerdershandleiding samenstellen	16-9-2004	24-9-2004	1.40w		

Tabel 6, Timebox Rapportage fase

ID	Task Name	Start	End	Duration	Sep 2004	
					9-19	9-26
1	Verslag afronden	24-9-2004	7-10-2004	2w		

Bijlage B Scope

Afbakening

Bij de afbakening van het webwinkel project was het belangrijk om in de gaten te houden dat de grens van het project niet op een dusdanige wijze getrokken werd dat de probleemstelling niet meer opgelost zou kunnen worden.

Met dit in het achterhoofd is het project begrensd bij het ontwikkelen van één webwinkel applicatie. Deze applicatie is alleen gericht op het mogelijk maken van de verkoop van producten op het internet. Functionaliteiten als het eventueel inpassen in een bestaand informatiesysteem vallen buiten de reikwijdte van het project.

Tevens is besloten het project in twee delen op te splitsen, een functioneel deel en een grafisch deel. Het functionele deel zal worden ontwikkeld door de afstudeerder en het grafische deel door grafisch ontwerper mevr. M. Burger.

Functionaliteiten

Tijdens het bespreken van de functionaliteiten ging het erom dat vast kwam te staan wat precies de wensen van de opdrachtgevers waren en of deze wensen haalbaar waren. Een belangrijk aspect hierbij was dat de wensen en eisen van de opdrachtgevers de probleemstelling oplossen.

Alvorens dieper in te gaan op de functies van de webwinkel zijn eerst globale eisen opgesteld. De globale eisen zijn:

- Generieke applicatie, dus de code is vrijwel volledig herbruikbaar bij een andere webwinkel.
- Bezoekers kunnen producten bekijken en zoeken.
- Bezoekers kunnen producten aan hun winkelwagen toevoegen
- Bezoekers kunnen favoriete producten aan een persoonlijke boodschappenlijst toevoegen voor hergebruik, zodat zij bij een tweede bezoek niet opnieuw naar die producten hoeven te zoeken.
- Bezoekers kunnen producten bestellen.

Vervolgens is er dieper ingegaan op de globale eisen wat leidde tot de volgende functionaliteiten:

- Bestelfunctie
 - De bezoeker kan een product selecteren en het te bestellen aantal instellen.
 - Als de bezoeker op product toevoegen klikt dan wordt de winkelwagen gevuld met het product en het gewenste aantal.
 - Als het product al in de winkelwagen aanwezig is dan wordt het te bestellen aantal opgehoogd met het ingestelde aantal.
 - Wordt voor het eerst een product toegevoegd aan de winkelwagen dan moet de gebruiker de regio selecteren (Eu, Europa buiten EU, overige landen), vervolgens de verzendwijze (standard/ priority default is standard).
Na het selecteren berekent de webwinkel de verzendkosten en het totaalbedrag.
 - Als de bezoeker op bestellen klikt dan moet hij zijn persoonsgegevens invoeren en betaalt hij.
 - Na betaling ontvangt de bezoeker een orderbevestiging per e-mail.
 - De eigenaar van de webwinkel krijgt ook een email met de bestelling.
Het emailadres van de eigenaar van de webwinkel is instelbaar.
- Boodschappenlijstfunctie
 - De bezoeker kan een product toevoegen aan zijn persoonlijke boodschappenlijst
 - De bezoeker kan een product verwijderen uit de boodschappenlijst
 - De bezoeker kan een product toevoegen aan winkelwagen vanuit de boodschappenlijst
- Zoekfunctie
 - De bezoeker kan zoeken op productnaam
 - Zijn er geen resultaten dan wordt de zoekterm opgeslagen in de database en krijgt de bezoeker een optie te zien voor het aanvragen van een offerte om het niet bezoeker product alsnog te bestellen
- Offerte aanvraag functie
 - De bezoeker kan een offerte aanvragen door op "offerte aanvragen" te klikken.
 - Na versturing ontvangt de gebruiker direct een ontvangst bevestiging

- Product onderhoudsfunctie
 - De administrator kan inloggen op de onderhoudsmodule
 - De administrator kan de producten op de site wijzigen door een csv-bestand met productinformatie te uploaden naar de server.
 - De administrator kan de niet gevonden zoektermen opvragen en verwijderen uit de database.
 - De administrator kan afbeeldingen uploaden naar de webwinkel
- Betaalwijze
 - De bezoeker kan met creditcard betalen, de creditcard betalingsmodule zal geleverd worden door Buckaroo BV.
- Taalfunctie
 - De webwinkel moet in de volgende talen beschikbaar zijn:
 - Nederlands
 - Engels
 - De gebruiker kan zelf de taal van zijn voorkeur kiezen
- Webpollfunctie
 - De bezoeker kan middels een webpoll aangeven welke producten hij nog mist in de product voorraad.
- Informatieschermen
 - Er komen informatie schermen over de privacy policy en de verzendkosten.
- Generiek
 - De webwinkel moet generiek zijn, dat wil zeggen bruikbaar bij verschillende afnemers zonder al te veel aanpassingen.

Prioriteiten

Toen eenmaal de functionaliteiten bekend waren, is er overgegaan tot het stellen van de prioriteiten. Dit is gedaan met gebruikmaking van de MoSCoW (Must have, Should have, Could have, Won't have) regels. De tabel hieronder geeft een overzicht van de functionaliteiten en hun prioriteiten.

Functionaliteit	Prioriteit
Bestelfunctie	Hoog
Product onderhoudsfunctie	Hoog
Betaalwijze	Hoog
Generiek	Hoog
Boodschappenlijstfunctie	Middel
Zoek functie	Middel
Offerte aanvraagfunctie	Middel
Taalfunctie	Middel
Webpollfunctie	Laag
Informatieschermen	Laag

Gezien de beperkte tijdsduur van de afstudeerperiode is er besloten de onderdelen met de laagste prioriteit te laten vervallen.

Doelgroep van de webwinkel

Het is van belang de doelgroep goed te definiëren, dit omdat het opgeleverde product op de doelgroep moet aansluiten qua taal, lay-out etc. zodat het eindproduct ook daadwerkelijk bruikbaar voor de doelgroep is.

De potentiële klant wilde graag een webwinkel hebben voor Nederlanders in het buitenland die producten willen kopen die alleen in Nederland verkrijgbaar zijn. Aangenomen is dat het hier zal gaan om Nederlanders die van middelbare leeftijd zijn en zowel engels als Nederlands als voertaal hebben.

Daarnaast moet de webwinkel generiek worden. Het is dus zaak dat als er een afnemer komt die een webwinkel wilt, die een andere doelgroep voor ogen heeft, dat de niet generieke code van webwinkel dan eenvoudig aangepast kan worden aan de juiste doelgroep.

Bijlage C Functioneel ontwerp

Functioneel Ontwerp Boodschappen Uit Holland

Auteur(s):	Alexander Bouwens
Laatste wijziging:	24-05-2004
Versie:	1.0
Status:	Voorlopig
Copyright:	ASolutions BV

<i>Dit Functioneel Ontwerp is uitsluitend bestemd om te worden gebruikt door de opdrachtgever en mag niet door hem/haar, zonder voorafgaande toestemming van ASolutions b.v., worden verveelvoudigd, openbaar gemaakt of ter kennis van derden worden gebracht.</i>

Auteur(s) : Alexander Bouwens
 Versie : 1.0
 Status : Voorlopig

 Bestandsnaam : Functioneel Ontwerp Boodschappen Uit Holland 1.0.Doc
 Aanmaakdatum : 24-05-2004
 Laatste wijziging : 24-05-2004

 Aantal pagina's : 86

 Vrijgegeven door : Paraaf:

Wijzigingshistorie

Versie	Datum	Auteur	Opmerking
0.1	24-05-2004	Alexander Bouwens	Initiële versie
0.1	28-05-2004	Alexander Bouwens	
0.1	1-06-2004	Alexander Bouwens	
0.2	3-6-2004	Alexander Bouwens	Verbeterde versie
0.3	18-06-2004	Sven Borman	Kleine aanpassingen
0.4	28-6-2004	Alexander Bouwens	Verbeterde versie na commentaar
0.5	30-6-2004	Alexander Bouwens	Verbeterde versie na commentaar
0.6	5-7-2004	Alexander Bouwens	Verbeterde versie na commentaar
1.0	7-7-2004	Alexander Bouwens	Definitief voorstel
1.1	12-7-2004	Alexander Bouwens	Definitief

DISTRIBUTIELIJST

Versie	Namen
0.1	Alexander Bouwens, Sven Borman
0.2	Alexander Bouwens, Sven Borman
0.3	Alexander Bouwens, Anneke Aldershof
0.4	Alexander Bouwens, Anneke Aldershof, Sven Borman
0.5	Alexander Bouwens, Anneke Aldershof, Sven Borman
0.6	Alexander Bouwens, Anneke Aldershof, Sven Borman
1.0	Alexander Bouwens, Anneke Aldershof, Sven Borman
1.1	Alexander Bouwens, Anneke Aldershof, Sven Borman

Inhoudsopgave

Inhoudsopgave	54
1. Inleiding	55
1.1 Doel	55
1.2 Autorisatie.....	55
1.3 Platform	55
2. Webwinkel	56
2.1 Homepage.....	57
2.2 Webpoll.....	58
2.3 Meertaligheid.....	58
2.4 Productpagina	59
2.5 Product zoeken	61
2.6 Offerte aanvragen	61
2.7 Informatieve Popup	61
2.8 Winkelwagen	63
2.9 Bestelscherm.....	65
2.10 Boodschappenlijst	66
2.11 Administrator pagina	67
2.11.1 Beveiliging	67
2.11.2 Upload producten	68

1. Inleiding

1.1 Doel

Het doel van de webwinkel applicatie is het overzichtelijk en veilig verkopen van goederen via internet.

De gebruikers van de applicatie kunnen de aangeboden producten bestellen via creditcard. Wanneer een gebruiker een product zoekt dat niet in het standaardassortiment zit, kan een offerte voor dit product worden aangevraagd.

1.2 Autorisatie

In principe zijn alle webpages benaderbaar voor alle gebruikers. Alleen de administrator page, waarmee de producten kunnen worden geupload is door middel van een gebruikersnaam en wachtwoord beveiligd.

1.3 Platform

De website zal worden ontwikkeld in het Java Struts framework, waarbij gebruik wordt gemaakt van de TomCat ApplicationServer en MySQL als database. Het voordeel van deze ontwikkelomgeving is dat de webwinkel in principe op vrij veel platformen kan draaien.

De ApplicationServer TomCat is geschreven in java en draait dus op ieder platform waarop een java compiler beschikbaar is. Van MySQL is een versie beschikbaar voor de volgende platformen: Linux, Windows, MAC, FreeBSD, HP Unix, IBM AIX, Novell, OpenBSD, SGI Irix en Dec OSF.

Door gebruik van TomCat en MySQL is het dus mogelijk om de webwinkel op meerdere platformen te laten draaien.

2. Webwinkel

De webwinkel bestaat uit de volgende onderdelen:

- Homepage
- Webpoll
- Meertaligheid
- Productpagina
- Product zoeken
- Offerte aanvragen
- Informatieve popup
- Winkelwagen
- Bestelscherm
- Boodschappenlijst
- Administrator pagina (cms)

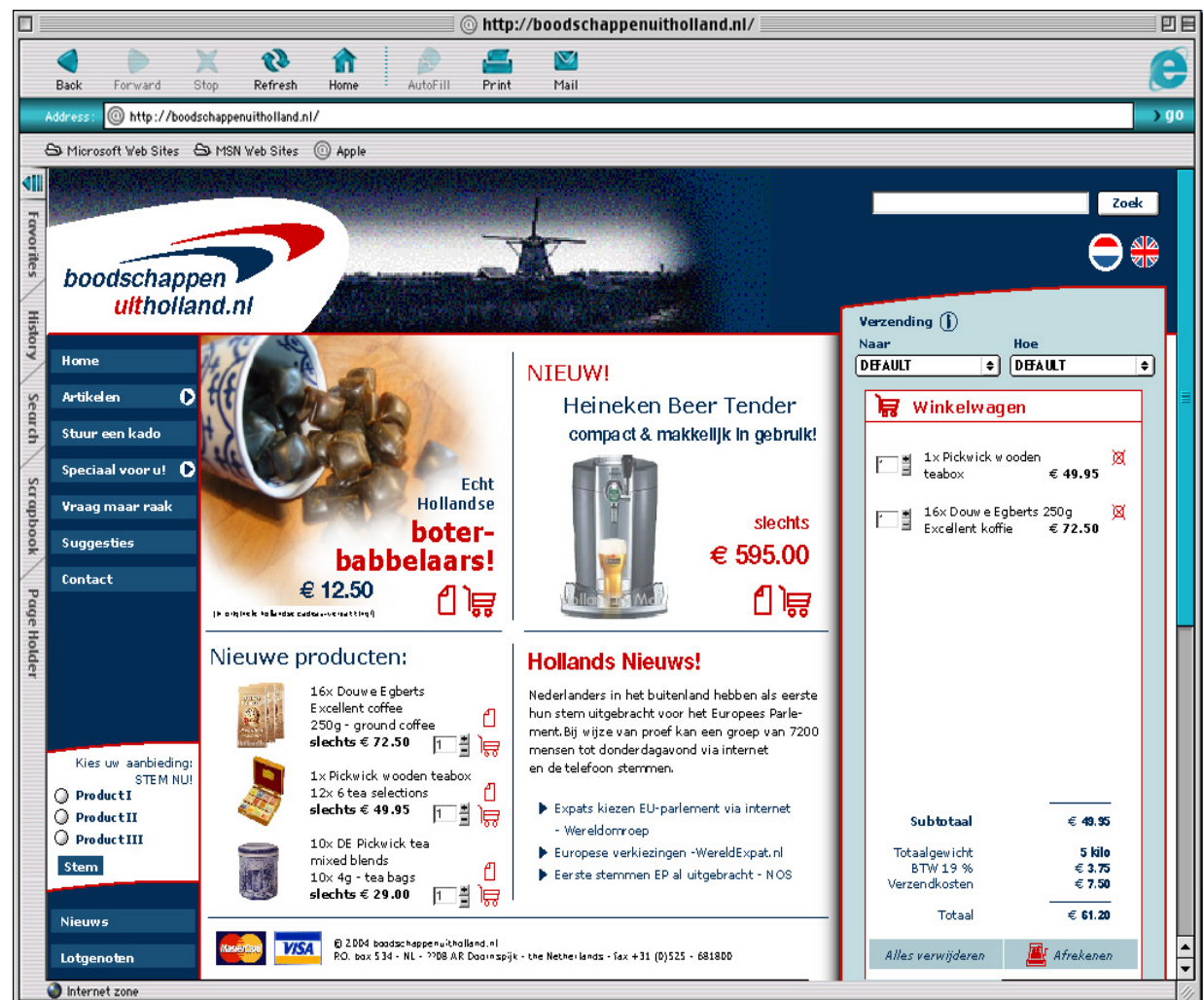
In de volgende paragrafen zullen deze onderdelen één voor één worden beschreven.

In dit document zijn verschillende voorbeeldschermen opgenomen om de uiteindelijke werking van de applicatie te verduidelijken.

Het is mogelijk dat de uiteindelijke schermen zowel qua opmaak als functionaliteit afwijken van de in dit document opgenomen schermen.

2.1 Homepage

Hieronder staat een schermafdruck van de mogelijke homepage van www.boodschappenuitholland.nl.



De homepage bestaat uit een aantal onderdelen te weten:

- Bovenin header met logo, zoekoptie, meertaligheidsoptie.
- Linkerzijde productcategorieën, boodschappenlijst, webpoll, offerteaanvraag, contact, en secundaire functionaliteit zoals forum en nieuws (vallen buiten de scope van de opdracht).
- Body overzicht van de aanbiedingen, nieuwe producten. Geselecteerde productcategorieën worden hier weergegeven.
- Rechterzijde winkelwagen
- Onderin footer, bevat links naar algemene info, verzendinginfo, betaalinfo, privacy statement, about us.

2.2 Webpoll

De webpoll is een onderdeel van de linker menubalk en komt in principe op alle basisschermen voor. Hij zal in eerste instantie vormgegeven worden door een keuzemenu waarin de klant zijn keuze maakt. Na het maken van de keuze krijgt de klant een dankbericht te zien voor het stemmen en ontvangt de administrator een mail met de gemaakte keuze.

2.3 Meertaligheid

De webwinkel moet in de volgende twee talen beschikbaar zijn:

- Nederlands
- Engels

De bezoeker kan zelf de taal van zijn voorkeur kiezen. Andere talen vallen buiten de scope van dit project.

De meertaligheidsoptie is altijd beschikbaar aangezien deze in de header staat die altijd hetzelfde blijft.

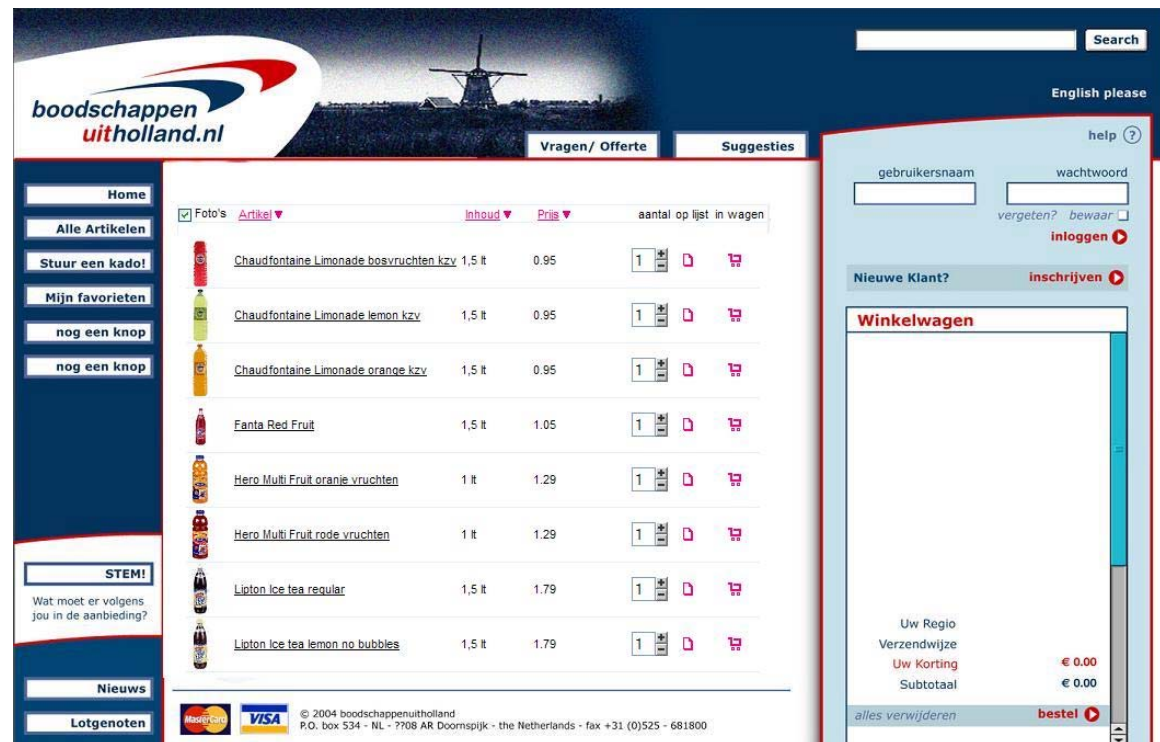
De default taal is Nederlands.

Bij een verandering van taal door de gebruiker wordt de pagina ververst en worden de Engelse/ Nederlandse namen, beschrijvingen en aanvullende informatie van de producten uit de database gehaald.

2.4 Productpagina

Wanneer de bezoeker vanuit de homepage naar de producten pagina gaat, worden de producten van de gekozen categorie in een lijst weergegeven van 10 producten per pagina (dit aantal is in te stellen in een property file op de WebServer en is dus alleen in te stellen door de administrator). Alleen de body wordt dan opnieuw gevuld met producten uit de gekozen categorie.

Hieronder staat de schermafdruck van de productpagina.



Per product worden de volgende gegevens getoond:

- De productnaam, met popup link
- Productprijs
- Productgewicht
- Productdetails
- De productafbeelding, met popup link
- Het door de klant in te stellen aantal te bestellen producten
- Bevat knop "aanvullende informatie aanwezig" met popup link.
- De knop "toevoegen aan winkelwagen"
- De knop "toevoegen aan boodschappenlijst"
- De optie om foto's wel of niet weer te geven.
- Er kan gesorteerd worden op prijs en naam.

De weergegeven informatie voldoet aan de volgende eisen:

Klein_plaatje

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	40
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Productnaam(nl/en)

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	40
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

Prijs

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

Gewicht (gram)

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

Details(nl/en)

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	50
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Aantal

Invoer	Tekstveld
Formaat	Numeriek 99
Lengte	2
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	1

Wanneer de bezoeker op de kleine afbeelding drukt, op de productnaam of op de knop “aanvullende informatie aanwezig”, dan zal een pop-up scherm worden getoond. In dit pop-up scherm wordt een grotere afbeelding van het product getoond en alle beschikbare informatie. Zie voor een beschrijving van deze pop-up paragraaf 2.7.

De gebruiker kiest een product en stelt het te bestellen aantal in, dit aantal staat default op 1. Als de gebruiker op product toevoegen klikt dan wordt de winkelwagen gevuld met het product en het gewenste aantal. Het totaal gewicht wordt opgehoogd.

Als het product al in de winkelwagen aanwezig is dan wordt het te bestellen aantal opgehoogd met het ingestelde aantal.

Zolang de gebruiker zijn land en de verzendwijze (standard of priority) nog niet geselecteerd heeft dan worden de verzendkosten en totale kosten nog niet berekend. Default zal de verzendwijze op standaard staan.

Na het selecteren berekent de webwinkel de verzendkosten op basis van totaalgewicht, land en verzendwijze en het totaalbedrag op basis van subtotale, incl. of excl. btw afhankelijk van land en verzendkosten.

Als de gebruiker op kassa klikt dan moet hij zijn persoonsgegevens invoeren, betaalgegevens invoeren en zijn bestelling bevestigen op de totaalpagina.

Na betaling ontvangt de gebruiker een orderbevestiging per email, de eigenaar van de webwinkel krijgt ook een email met de bestelling.

Als men op de knop "toevoegen aan boodschappenlijst" druk dan wordt het product toegevoegd aan de boodschappenlijst met het ingestelde aantal.

Het emailadres van de eigenaar van de webwinkel is instelbaar door de administrator.

2.5 Product zoeken

De gebruiker kan zoeken op productnaam

Zijn er geen resultaten dan wordt de zoekterm opgeslagen in de database en krijgt de gebruiker een "geen resultaten" melding te zien en een scherm voor het aanvragen van een offerte om het niet beschikbare product alsnog te bestellen in de body.

In het administrator gedeelte van de website wordt in eerste instantie geen mogelijkheid gegeven om de opgeslagen zoektermen te bekijken, te downloaden en eventueel te resetten. De resultaten kunnen opgevraagd worden bij ASolutions.

2.6 Offerte aanvragen

Als een gebruiker een product niet kan vinden op de webwinkel, is het mogelijk om een offerte hiervoor aan te vragen. De bezoeker drukt hiervoor op de knop "offerte aanvragen" of krijgt dit deel na een mislukte zoekpoging voorgeschoteld en hij/zij kan dan aangeven voor welke producten een offerte moet worden aangevraagd. Na versturing van de offerte aanvraag, krijgt de bezoeker een ontvangst bevestiging per email toegestuurd.

Geachte "klant",

Dank voor de aanvraag voor een offerte voor:

<tekst van de klant die hij op het offertescherm heeft ingetikt>

We gaan voor u aan het werk en houden contact met u!

Binnen 2 werkdagen ontvangt u in ieder geval meer informatie van ons!

Met vriendelijke groet,

Boodschappen Uit Holland

offerte@boodschappenuitholland.nl

2.7 Informatieve Popup

De links naar de informatieve popups bevinden zich in het kleine plaatje, productnaam en de knop "aanvullende informatie aanwezig" in alle schermen waar producten getoond worden.

De popups bevatten de volgende onderdelen:

- Productnaam
- Productprijs
- Productgewicht
- Productdetails
- Grote productafbeelding
- Aanvullende informatie
- Knop "toevoegen aan boodschappenlijst"
- Knop "toevoegen aan winkelwagen"

De getoonde informatie moet aan de volgende eisen voldoen:

Productnaam(nl/en)

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	40
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Prijs

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Gewicht(gram)

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Details(nl/en)

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	50
Wijzigbaar	Ja
Verplicht	Nee
Standaard waarde	

Groot_plaatje

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	40
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Aanvullende_informatie(nl/en)

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	400
Wijzigbaar	Ja
Verplicht	Nee
Standaard waarde	

2.8 Winkelwagen

De winkelwagen bestaat uit de volgende onderdelen en zal altijd te zien zijn behalve bij de betaalschermen.:

- Informatie knop “verzendinginformatie”
- Bestelland
- Verzendwijze
- Productoverzicht, het productoverzicht op de winkelwagen bevat per product:
 - De productnaam
 - Productprijs
 - Het aantal te bestellen producten
 - De knop “verwijderen uit winkelwagen”
- Verzendkosten
- Subtotaal
- Totale kosten
- Totaal gewicht
- De knop “kassa”
- BTW kosten
- Knop “Alles verwijderen”

De weergegeven informatie voldoet aan de volgende eisen:

Productnaam (nl/en)

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	40
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

Prijs

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

Gewicht

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

Aantal

Invoer	Tekstveld
Formaat	Numeriek 99
Lengte	2
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	1

Totaal gewicht (in kg)

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

Verzendkosten

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

Subtotaal

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

Totale kosten

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

BTW kosten

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

Automatisch worden de verzendkosten en totale kosten berekend bij het toevoegen en/of verwijderen van een product. Bij het toevoegen van een product wordt automatisch het totale gewicht berekend, overschrijdt dit de maximum toegestane hoeveelheid dan krijgt de klant een bericht te zien. Op dit bericht staat de melding van de overschrijding en de opties die hij kan gebruiken om dit probleem op te lossen (bv. Twee bestellingen maken, een offerte aanvragen of een product uit de winkelwagen verwijderen).

Door op de knop "kassa" te drukken, wordt de winkelwagen definitief gemaakt en komt de bezoeker terecht in het bestelscherm. Zie voor een beschrijving hiervoor de volgende paragraaf. Een lege winkelwagen bestellen is onmogelijk. Bij het verwijderen van alle producten krijgt de klant een melding of hij dit wel zeker weet.

2.9 Bestelscherm

Dit scherm komt naar voren als de bezoeker een aantal producten in zijn/haar winkelwagen heeft gedaan en daarna op de knop “kassa” heeft gedrukt. Als eerste zullen de persoonsgegevens van de bezoeker moeten worden ingevoerd. Ook het adres waar de producten moeten worden afgeleverd zal moeten worden ingevoerd.

Nadat dit is gedaan kan de bezoeker zijn betaalgegevens invoeren en betaald hij/zij, dit gebeurt in de schermen van www.buckaroo.nl met visa of mastercard.

De opmaak, inhoud en werking van het bestelscherm is afhankelijk van de gekozen betaalmaatschappij. Aangezien veel maatschappijen zelf schermen leveren voor betaling via internet kan er op dit moment nog geen duidelijkheid verschaft worden over de inhoud, vormgeving en werking van dit scherm. Hierna moet de bezoeker een orderbevestiging krijgen van de bestelling.

De orderbevestiging naar de klant heeft de volgende layout:

Geachte heer/mevr “klantnaam”,

U heeft bij Boodschappen Uit Holland de volgende artikelen besteld:

- “productnaam” “productaantal” “productprijs”
- “productnaam” “productaantal” “productprijs”
- “productnaam” “productaantal” $\frac{\text{“productprijs”}}{\text{“totaalprijs”}} +$

Uw bestelling wordt bezorgd binnen <indicatie verzendduur> op het volgende adres:

Afleveradres,
Indicatie verzendduur.

Contactinfo bij vragen/problemen/klachten.

Wij wensen u veel plezier van uw aankoop

Met vriendelijke groet,

Boodschappen Uit Holland

De email naar de eigenaar van de webwinkel met de bestelling ziet er als volgt uit:

Geachte “eigenaar”,

Bij Boodschappen Uit Holland zijn de volgende artikelen besteld door dhr/mevr “klantnaam” :

- “productnaam” “productaantal” “productprijs”
- “productnaam” “productaantal” “productprijs”
- “productnaam” “productaantal” $\frac{\text{“productprijs”}}{\text{“totaalprijs”}} +$

verzendinginfo (wijze, adres, indicatie duur)

Met vriendelijke groet,

Boodschappen Uit Holland

2.10 Boodschappenlijst

De boodschappenlijst ziet er vrijwel hetzelfde uit als de product pagina en bestaat uit de volgende onderdelen per product:

- De kleine afbeelding met popup optie
- De productnaam met popup optie
- Productprijs
- Productgewicht
- Product details
- De knop “verwijderen uit boodschappenlijst”
- De knop “toevoegen aan winkelwagen”
- Het in te stellen aantal te bestellen producten
- Knop “aanvullende informatie aanwezig”

Productnaam (nl/en)

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	40
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

Prijs

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

Details(nl/en)

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	50
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Gewicht

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Nee
Verplicht	Ja
Standaard waarde	

Klein_plaatje

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	40
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

2.11 Administrator pagina

Hieronder staat een schermafdruck van de administrator pagina.

The screenshot shows the administrator interface of the website boodschappen.nl. The header features the site logo and a search bar. A left sidebar provides navigation options. The central area displays a message for administrators. On the right, a login panel is visible with fields for username and password, and a login button. The footer contains payment logos and contact details.

2.11.1 Beveiliging

Voordat de administrator pagina kan worden bekeken, zal eerst om een username en password worden gevraagd. Wanneer hier de juiste gegevens worden ingevoerd zal pas de administrator pagina worden getoond.

De pagina bevat 2 invoervelden te weten, username en password. De waarden die in deze velden ingevoerd dienen te worden kunnen alleen gewijzigd worden in (telefonisch) overleg met ASolutions. De velden moeten aan de volgende eisen voldoen:

Username

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	40
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Password

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	10
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

2.11.2 Upload producten

De upload pagina bevat een optie om een CSV-bestand met producten te uploaden.

De CSV file die geupload wordt heeft de volgende layout in MS Excel:

Product_nl	Product_en	Details_nl	Details_en	Info_nl	Info_en	P r i j s	Gewicht	Leverbaar	Nieuw	Aanbieding	Kl_plaatje	Gr_plaatje
------------	------------	------------	------------	---------	---------	-----------	---------	-----------	-------	------------	------------	------------

Na opslag als CSV file ziet de inhoud er als volgt uit:

Product_nl;Product_en;Details_nl;Details_en;Info_en;Info_nl;Prijs;Gewicht;Leverbaar;Nieuw;Aanbieding;Kl_plaatje;Gr_plaatje;

De waarden in de CSV file moeten aan de volgende eisen voldoen:

Product_nl

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	40
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Product_en

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	40
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Details_nl

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	50
Wijzigbaar	Ja
Verplicht	Nee
Standaard waarde	

Details_en

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	50
Wijzigbaar	Ja
Verplicht	Nee
Standaard waarde	

Info_nl

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	400
Wijzigbaar	Ja
Verplicht	Nee
Standaard waarde	

Info_en

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	400
Wijzigbaar	Ja
Verplicht	Nee
Standaard waarde	

Prijs

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Gewicht

Invoer	Tekstveld
Formaat	Numeriek 99999,99
Lengte	8
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Leverbaar

Invoer	Tekstveld
Formaat	Datum
Lengte	11
Wijzigbaar	Ja
Verplicht	Nee
Standaard waarde	

Nieuw

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	3
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	Ja of Nee

Aanbieding

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	3
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	Ja of Nee

Klein_plaatje

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	40
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Groot_plaatje

Invoer	Tekstveld
Formaat	Alfanumeriek
Lengte	40
Wijzigbaar	Ja
Verplicht	Ja
Standaard waarde	

Bijlage D Beheerderhandleiding

Beheerderhandleiding webwinkel applicatie

Auteur: Alexander Bouwens
Bedrijf: ASolutions - Advanced Software Solutions

1. Inleiding

Deze handleiding het tot doel de webwinkel applicatie toe te lichten.
In de handleiding zal het uploaden van producten worden behandeld, daarna zal het model (klassendiagram) van de applicatie gegeven worden.

2. Upload functie

Het vullen van de database geschiedt door middel van het uploaden van een csv-bestand.
Dit csv-bestand moet aan het hieronder beschreven format voldoen en kan worden bewerkt in microsoft Excel.

De bovenste regels van het bestand wordt gevormd door de headers.

In tabel 1 staan de headers beschreven. De headers in tabel 1 zijn voor de overzichtelijkheid gedeeltelijk onder elkaar geplaatst, in het csv-bestand staan ze echter allemaal naast elkaar.

Tabel 1, Headers

Product_nl	Product_en	Details_nl	Details_en	Info_nl	Info_en	Prijs	BTW	Gewicht
Leverbaar	Nieuw	Aanbieding	plaatje	CategorieNL	CategorieEN	Hoofd CategorieNL	Hoofd CategorieEN	

Deze headers mogen niet gewijzigd worden, vinden er echter wel wijzigingen aan de headers plaats dan zullen de producten niet succesvol geupload kunnen worden.
Als wijzigingen aan de headers toch noodzakelijk blijken te zijn neem dan contact op met ASolutions.

Onder de header komen de producten te staan, elk product op een afzonderlijke regel.
Onder de headers leverbaar, nieuw en aanbieding kunnen of leeg gelaten worden of de volgende waarden moeten in gevuld zijn, J of j of N of n.

Onder de header plaatje wordt de naam van het bijbehorende productplaatje ingevoerd met de extensie (.jpg, .gif etc.). Het pad (url) naar het plaatje moet niet toegevoegd worden.
Het plaatje zelf wordt na het uploaden van de producten geupload met behulp van de functie uploaden plaatjes.

Het csv-bestand is een afspiegeling van de database, dit houdt in dat oude producten die eerst in de database stonden vervangen zullen worden door de producten in het csv-bestand.

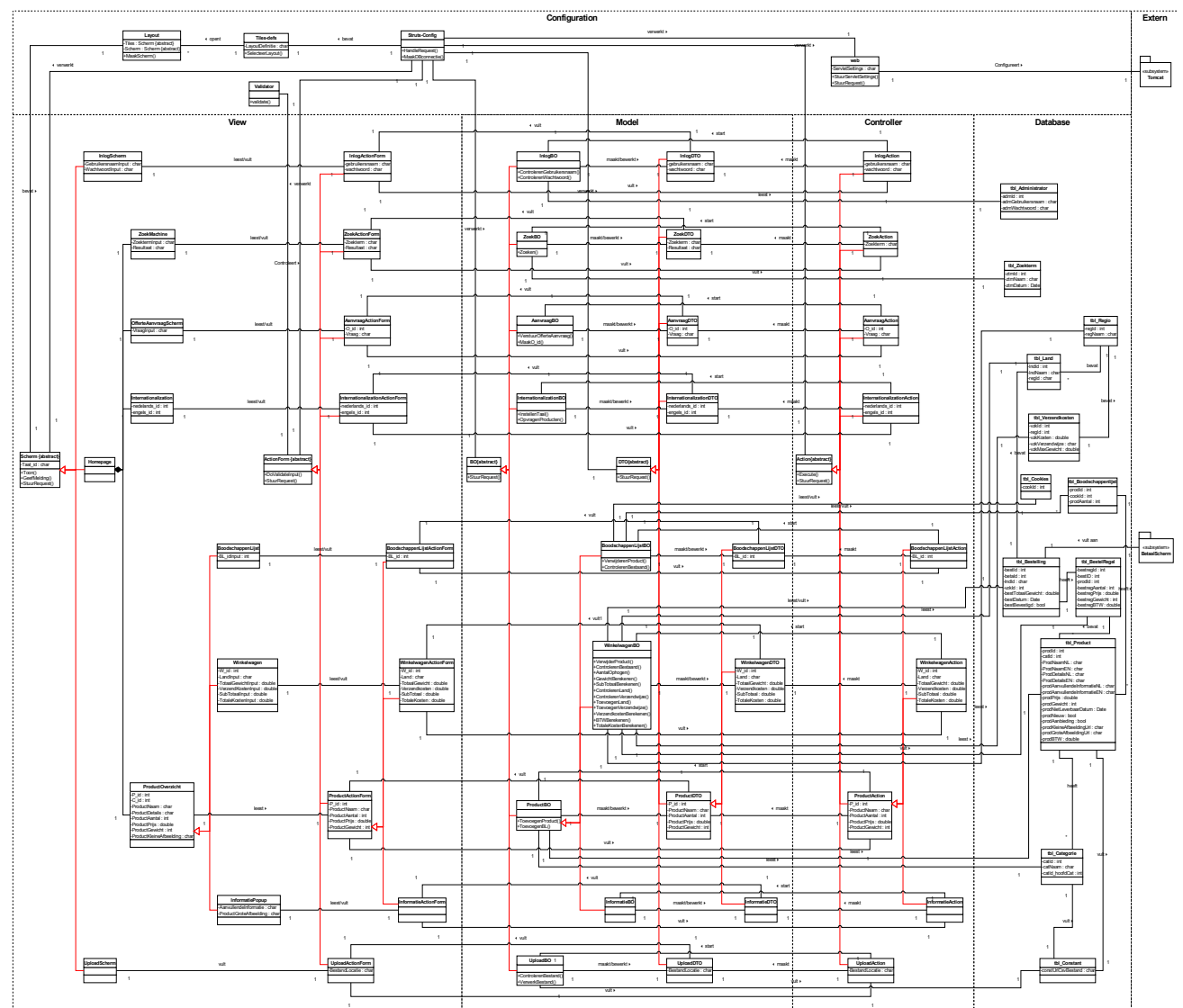
3. Het model

In dit hoofdstuk wordt het model van de applicatie gegeven. Eerst zal een overzicht gegeven worden van het complete klassendiagram vervolgens zullen de verschillende lagen van het klassendiagram behandeld worden.

3.1 overzicht compleet klassen diagram

Dit overzicht van het complete klassendiagram is alleen opgenomen om een beeld te verschaffen van de loop van de associaties die tussen de verschillende klassen en lagen.

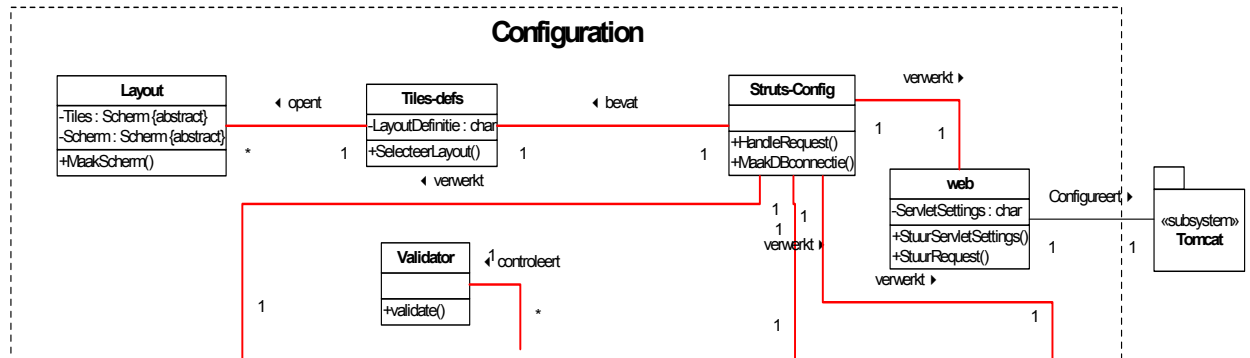
Figuur 1, overzicht compleet klassendiagram.



3.2 Configuratielaag

De configuratielaag is geen officiële laag uit het MVC-model, maar is wel een wezenlijk onderdeel van struts. De laag bestaat uit naar XML geconverteerde klassen die het struts framework aansturen en horen officieel in de controller laag thuis. De reden dat ze toch in een aparte laag zijn gezet is om de cruciale rol die klassen spelen in het framework te benadrukken. In figuur 2 staat de configuratielaag weergegeven.

Figuur 2, configuratielaag



De belangrijkste klasse in deze laag is de struts-config klasse.

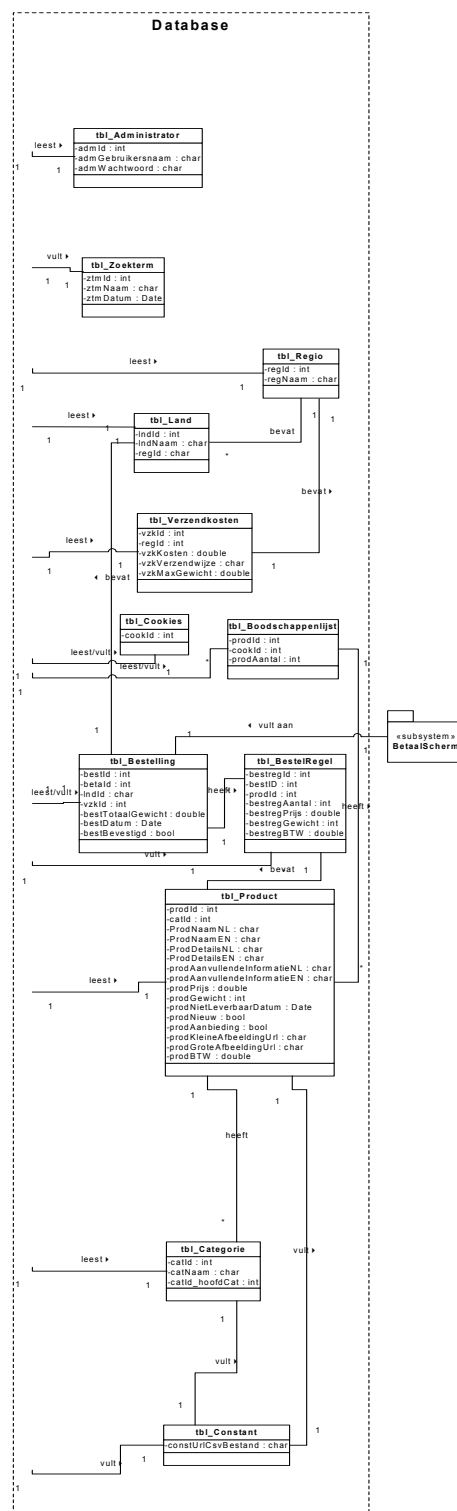
Deze klasse stuurt alle opdrachten (request) en applicatie reacties (responses) naar de juiste klasse en zorgt er zo voor dat de juiste functionaliteit wordt uitgevoerd bij een opdracht.

De web klasse dient voor de configuratie van de gekozen server. De lay-out en tiles-defs klassen zorgen voor de juiste scherm opbouw. De validator klassen zorgt tenslotte voor de validering van ingevoerde data.

3.3 Databaselaag

De databaselaag bevat alle klassen die database tabellen representeren. Deze laag staat alleen in contact met de modellaag. De modellaag is de enige laag van de applicatie waar met database informatie gewerkt mag worden. In figuur 3 is de databaselaag weergegeven.

Figuur 3, databaselaag

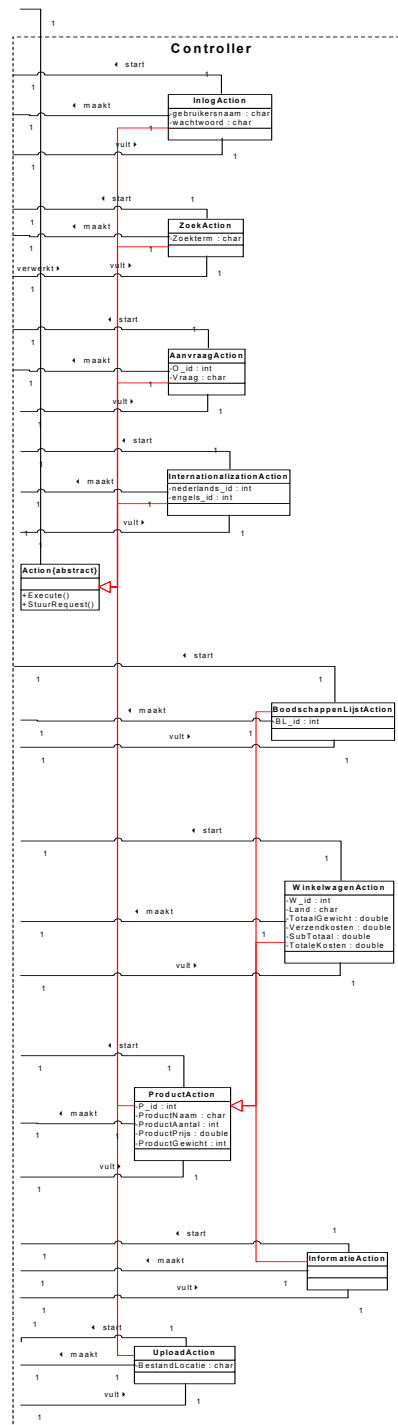


3.4 controllerlaag

De controllerlaag is verantwoordelijk voor het aansturen van de webwinkel applicatie. De klassen in deze laag ontvangen de opdrachten (request) en applicatie reacties (responses) van de struts-config klasse. De klassen bepalen vervolgens wat de functie in de modellaag moet uitvoeren.

Als de functie in de modellaag klaar is met zijn taak dan genereert de klasse in de controllerlaag een applicatie reactie (response) en koppelt daar eventueel mee te zenden data aan. Figuur 4 geeft een overzicht van de controllerlaag.

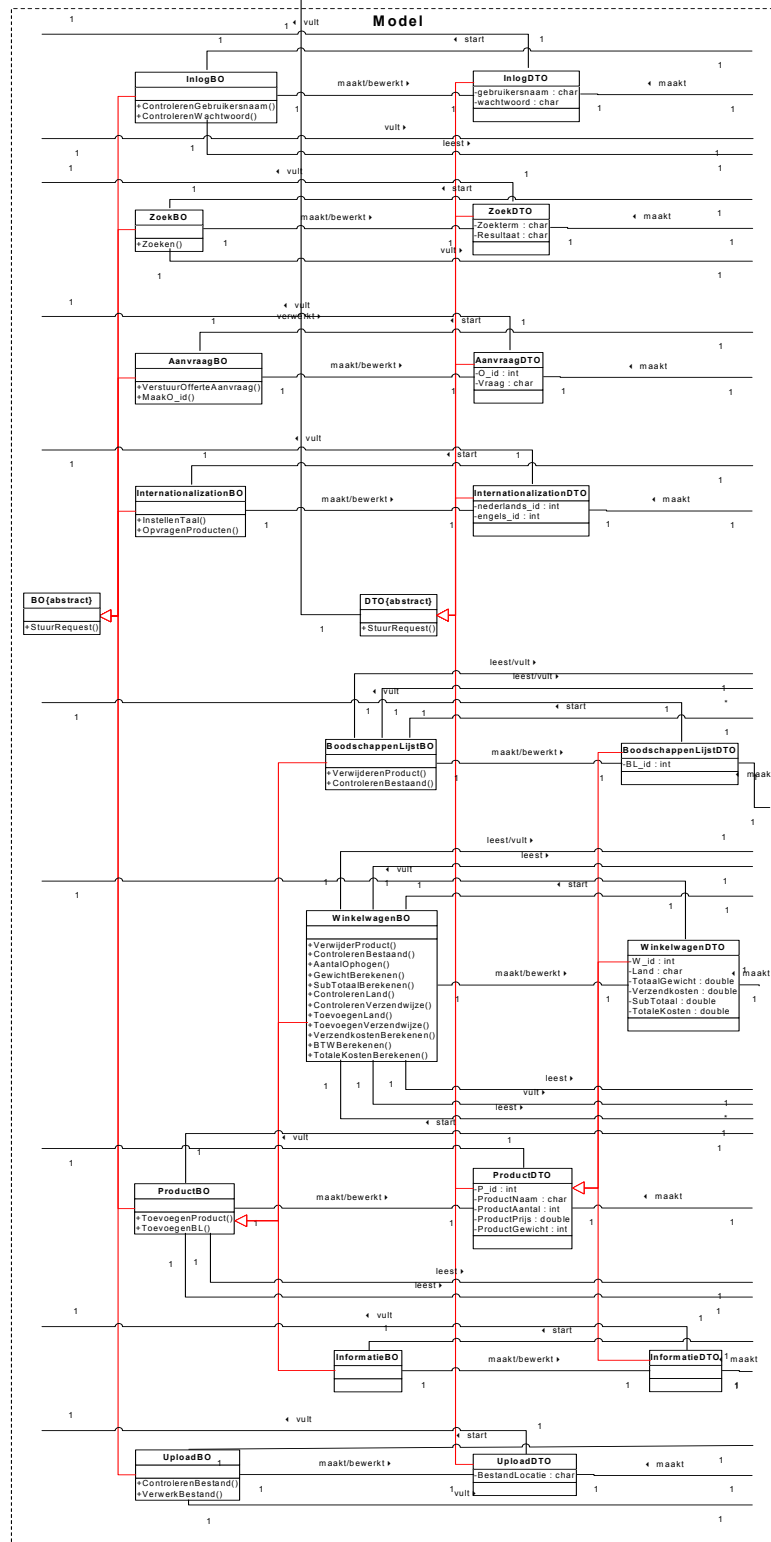
Figuur 4, de controllerlaag



3.5 Modellaag

De modellaag bevat de daadwerkelijke functionaliteiten van de applicatie. De laag wordt volledig aangestuurd door de controllerlaag en is de enige laag die contact maakt met de database. De klassen in deze laag voeren hun taak uit en vullen een data transfer object (dto) na afloop, dat doorgestuurd wordt naar de controllerlaag.

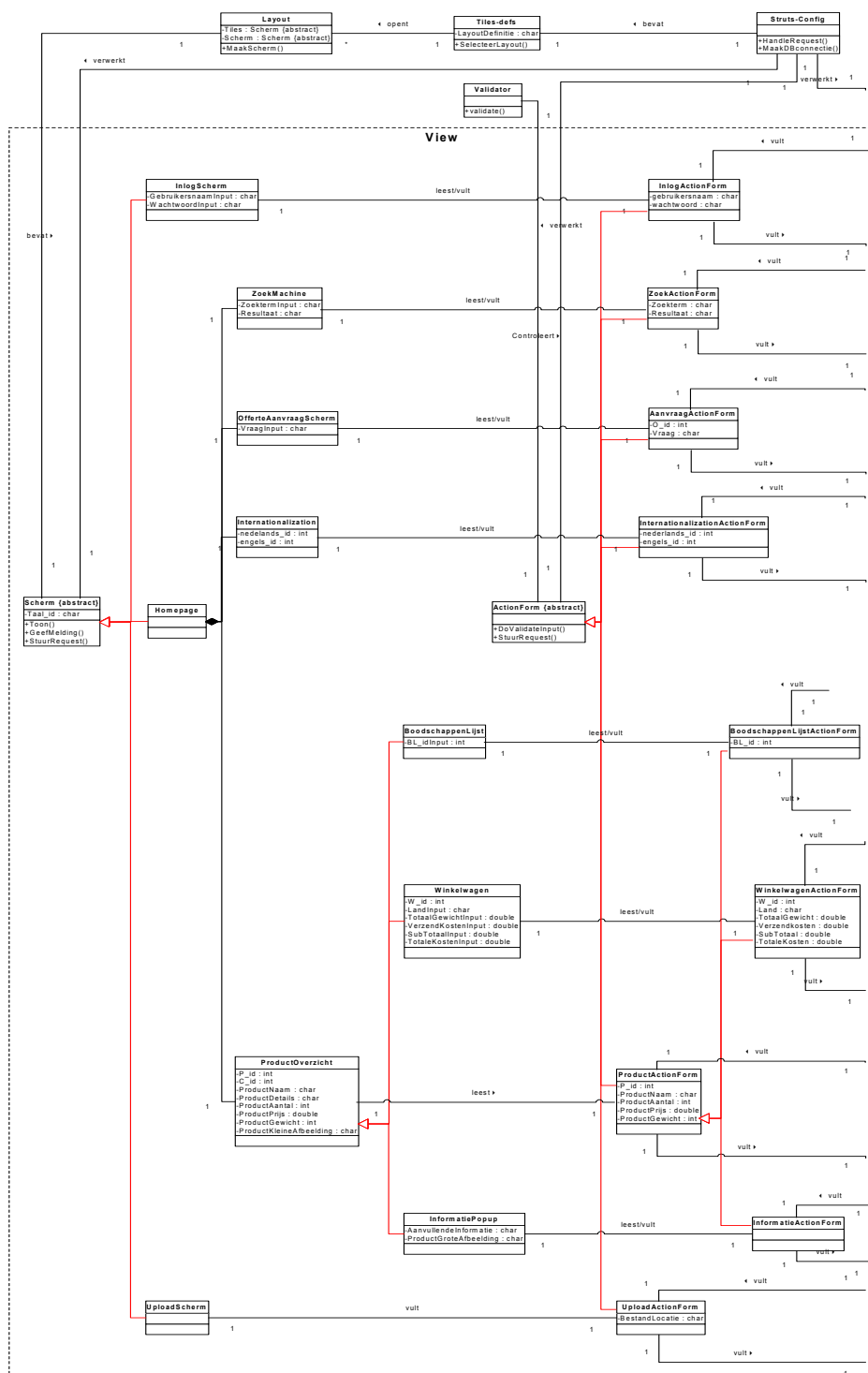
Figuur 5, de modellaag



3.6 Viewlaag

De viewlaag bevat naast alle schermen ook een bijzonder type data transfer object (dto) namelijk een actionform. Het actionform is een klasse die alle waarden van een bepaald htmlform bevat. Als een htmlform gesubmit wordt dan worden de waardes uit het htmlform gekopieerd naar het actionform. Vervolgens worden de waarden in het actionform gecontroleerd op ongeldige invoer. Als de waarden geldig zijn dan wordt het actionform doorgestuurd naar de controllerlaag, zo niet dan verschijnt er een melding op het scherm dat er geldige waarden ingevuld moeten worden.

Figuur 6, de viewlaag



Bijlage E Testset en testresultaten

Test Scenario's

- Nieuwe klant plaatst bestelling.
- Bestaande klant plaatst bestelling.
- Bestaande klant plaatst bestelling via catalogus, product pop-up windows, boodschappenlijst en homepage (door elkaar).
- Nieuwe klant plaatst bestelling via homepage, catalogus, product pop-up windows en boodschappenlijst (door elkaar; boodschappenlijst is leeg).
- Klant zoekt product dat wel bestaat in catalogus en plaatst daarna bestelling uit catalogus.
- Klant zoekt product dat niet bestaat in catalogus.
- Klant zoekt product dat niet bestaat in catalogus en plaatst daarna offerte en plaatst daarna bestelling uit catalogus.
- Nieuwe klant plaatst bestelling – zoekt product via zoekveld, bestelt, doet dit 3 x en daarna voegt hij nog iets van de productpagina toe.
- Klant verwijdert enkele producten uit de winkelmand en plaatst daarna bestelling
- Klant verwijdert enkele producten uit de winkelmand en voegt andere producten toe.
- Klant plaatst bestelling dat max. gewicht overschrijdt.
- Klant plaatst bestelling dat max. gewicht overschrijdt → past bestelling aan door producten te verwijderen.
- Klant plaatst bestelling dat max. gewicht overschrijdt → voegt nog meer producten aan de winkelmand toe.
- Klant plaatst bestelling dat meerdere malen max. gewicht overschrijdt.
- Klant wijzigt verzendregio.
- Klant wijzigt verzendwijze.
- Klant wijzigt verzendregio en verzendwijze.
- Klant vraagt algemene verzend/betaal/privacy info op.
- Klant leegt winkelmand en begint opnieuw.

Test Resultaten

Bij de eerste test dienden het productoverzicht, het zoekresultaten scherm en de boodschappenlijstscherms nog aangepast te worden. Bij de tweede test zijn alle scenario's gehaald.