

“Nieuwsbrief beheersysteem”

Beheermodule om op een eenvoudige manier nieuwsbrieven te genereren en te versturen.

Geheel naar wens van de gebruiker.



Afstudeerverslag

Afstudeerperiode:

- Februari - Juni 2004

Datum:

- Vrijdag 11 juni 2004

Differentiatie:

- Vormgeving en ontwerp van Interactie

Afstudeerbedrijf:

- Interpulse Automatisering B.V.

Auteur:

- R.J.T. ten Hove, 20007702

Bedrijfsbegeleider:

- Dhr. M. van Pelt

Examinatoren:

- Dhr. P.B. van der Sluijs

- Mevr. A.D. van Duuren

Referaat

Ten Hove, R.J.T.

Afstuderen

Ontwerp, bouw en implementatie van nieuwsbrief beheersysteem in opdracht van Interpulse Automatisering B.V.

Het afstudeerrapport beschrijft de werkzaamheden uitgevoerd tijdens de afstudeerperiode. Het project "nieuwsbrief beheersysteem" is uitgevoerd in kader van de opleiding Vormgeving en ontwerp van InterActie aan de Haagse Hogeschool.

DESCRIPTOREN

Iterative application development (IAD)

Hypertext Markup Language (HTML)

Interpulse Automatisering B.V.

Active Server Pages (ASP)

E-mail marketing

Alan Cooper

Jacob Nielsen

SQL server

Voorwoord

Voor u ligt het afstudeerverslag van Reinoud ten Hove. Dit document is de tekstuele beschrijving van de uitgevoerde activiteiten bij de ontwikkeling van het nieuwsbrief beheersysteem. Tijdens het project heb ik de op school geleerde theorieën in de praktijk kunnen brengen en me kunnen verdiepen in nieuwe en reeds bekende technieken.

Uiteraard wil ik als eerste enkele mensen bedanken voor de totstandkoming van dit verslag. Ik heb met erg veel plezier gewerkt aan het project bij Interpulse Automatisering B.V. te Leiden. Hierbij wil ik Martijn van Pelt bedanken voor de begeleiding en alle medewerkers van Interpulse. Zij hebben mij veel geleerd en ik kijk dan ook met veel plezier terug op deze afstudeerperiode.

Daarnaast heb ik de begeleiding vanuit school zeer gewaardeerd, mijn dank gaat uit naar mevr. Van Duuren en dhr. Van der Sluijs.

Juni 2004, Leiden

Reinoud ten Hove

Inhoudsopgave

1. INLEIDING	5
2. ORIËNTATIE	6
2.1 INTERPULSE AUTOMATISERING B.V.	6
2.2 OPDRACHTOMSCHRIJVING	6
2.3 PLAN VAN AANPAK	7
2.4 GEHANTEERDE METHODEN EN TECHNIKEN	9
2.5 PLANNING EN UIT TE VOEREN ACTIVITEITEN	13
2.6 OP TE LEVEREN PRODUCTEN	13
3. DEFINITIESTUDIE.....	14
3.1 ANALYSEREN VAN HUIDIGE SITUATIE	14
3.2 UITVOEREN DOELGROEP ANALYSE.....	16
3.3 ONDERZOEK NAAR ASP EN ASP.NET	17
3.4 ONDERZOEKEN VAN E-MAIL MARKETING	18
3.5 UITVOEREN CONCURRENTIE ANALYSE.....	20
3.6 OPSTELLEN USABILITY CHECKLIST.....	21
3.7 OPSTELLEN SYSTEEMEISEN	22
3.8 BESCHRIJVEN SYSTEEMCONCEPT.....	24
3.9 OPSTELLEN PILOTPLAN.....	29
4. PILOTONTWIKKELING	31
4.1 PILOTONTWIKKELING PILOT DATABASE.....	31
4.2 PILOTONTWIKKELING PILOT ARTIKELSYSTEEM	35
4.3 PILOTONTWIKKELING PILOT DOELGROEPBEHEER EN IMPORTEERSYSTEEM	40
4.4 PILOTONTWIKKELING PILOT NIEUWSBRIEF SAMENSTEL MODULE.....	43
5. INVOERING	46
5.1 INTEGRATIE SYSTEEM IN INTERPULSE CMS	46
5.2 OPERATIONALISEREN STAND-ALONE BEHEERSYSTEEM	47
5.3 ACCEPTATIE TEST VAN HET SYSTEEM	47
6. EVALUATIE	48
6.1 PROCESEVALUATIE.....	48
6.2 PRODUCTEVALUATIE	52

1. Inleiding

E-newsletters that are informative, convenient, and timely are often preferred over other media. However, a new study found that only 11% of newsletters were read thoroughly, so layout and content scan ability are paramount.

Jacob Nielsen. Alertbox 17 februari 2004

Elektronische nieuwsbrieven zijn de toekomst, steeds meer bedrijven sturen opgemaakt nieuwsbrieven naar klanten en andere geïnteresseerden. En, uit onderzoek blijkt dat steeds meer mensen digitale nieuwsbrieven lezen, mits deze op gebied van vormgeving en inhoud goed in elkaar zitten. De afstudeeropdracht omvat het ontwerp, de bouw en de invoering van een nieuwsbrief beheersysteem. Hieronder zal ik een korte beschrijving geven van de inhoud van de verschillende hoofdstukken uit dit afstudeerverslag.

Hoofdstuk 2: Oriëntatiefase

In dit hoofdstuk zal ik een inleiding geven met betrekking tot het project. Wat zijn de probleem -en doelstellingen en hoe ga ik deze doelstellingen bereiken. Ik zal de keuze van de methode en ontwikkelstrategie toelichten en de gehanteerde planning beschrijven. Uiteindelijk zal ik een korte opsomming geven van de op te leveren producten.

Hoofdstuk 3: Definitiestudiefase

Tijdens de definitiestudiefase heb ik een aantal onderzoeken uitgevoerd die belangrijk waren voor het project. Deze uitkomsten resulteerde in een lijst van systeemeisen waaruit vervolgens een systeemconcept en pilotplan voortvloeide. In hoofdstuk 3 een beschrijving.

Hoofdstuk 4: Pilotontwikkelingsfase

De pilotontwikkeling bestaat uit het ontwerp, de bouw en het testen van de verschillende pilots. In hoofdstuk 4 zal ik per pilot een beschrijving geven.

Hoofdstuk 5: Invoeringsfase

De laatste fase in het proces is de invoering van het systeem in de gebruikersomgeving. Hoe ik dit aangepakt heb zal ik in hoofdstuk 5 beschrijven.

Hoofdstuk 6: Evaluatie

In het laatste hoofdstuk zal ik mijn persoonlijke evaluatie geven over het geleverde werk. Ik zal het proces en de producten hierbij apart evalueren.

In de verschillende hoofdstukken zal ik steeds enkele voorbeelden geven van de uitwerkingen van de verschillende technieken. De volledige uitwerkingen zijn te vinden in de externe bijlagen, hier zal ik verder niet meer naar verwijzen.

2. Oriëntatie

2.1 Interpulse Automatisering B.V.

De afstudeerstage is uitgevoerd bij Interpulse Automatisering B.V. (nader te noemen Interpulse) te Leiden. Sinds 1999 is Interpulse actief in Leiden en omgeving. Deze organisatie houdt zich bezig op een viertal werkgebieden. Dit zijn de volgende afdelingen: consultancy, systeemontwikkeling, netwerk -en systeembeheer en internetdiensten. Interpulse is een groeiend bedrijf met 13 medewerkers, waarvan op dit moment drie medewerkers op parttime basis werkzaam zijn.

De afstudeerder is tijdens de periode werkzaam geweest bij de afdeling internetdiensten. Deze afdeling, bestaande uit een drietal medewerkers, verzorgt het ontwerp en de ontwikkeling van complete websites en content management systemen (CMS). Daarnaast biedt de afdeling degelijke hosting faciliteiten en snelle domeinregistratie.

2.2 Opdrachtschrijving

In deze paragraaf zal ik een beschrijving geven van de huidige problemen en de daaruit voortkomende doelstellingen.

2.2.1 Probleemstelling

Bedrijven hebben aan Interpulse te kennen gegeven dat er een behoefte is om op een eenvoudige manier nieuwsbrieven te genereren, zodat zij aantrekkelijke aanbiedingen naar hun klanten kunnen sturen.

De e-mail wordt als spam ingeschat. In de praktijk blijkt dat wanneer een nieuwsbrief binnenkomt in de mailbox deze regelmatig verwijderd wordt zonder dat er naar gekeken is.

Op dit moment verzorgt een derde partij de mailing. De reden hiervoor is dat Interpulse gebrek heeft aan expertise betreffende het onderwerp 'email marketing' en er een gemis is van een dusdanig product welke nieuwsbrieven kan genereren. Het inschakelen van zo'n derde partij heeft een aantal nadelen:

- Het inschakelen van de derde partij brengt extra kosten met zich mee voor Interpulse en haar klanten.
- De mailing kan niet altijd op het gewenste tijdstip de deur uit, door de afhankelijkheid van de derde partij.
- Interpulse moet de vormgeving aanpassen, doordat ze zich te houden hebben aan richtlijnen van derde partij. Dit levert veel tijdverlies op.

2.2.2 Doelstellingen

Men wil de derde partij uitschakelen, waardoor het gehele proces in eigen beheer is en men niet meer afhankelijk is van hen. Het doel van de afstudeeropdracht is het ontwikkelen van een beheermodule, waarmee bedrijven zelfstandig een nieuwsbrief kunnen genereren; deze oplossing zal bestaan uit de volgende componenten:

- Klanten, enerzijds van Interpulse, anderzijds "klanten van klanten" van Interpulse, de mogelijkheid bieden om zich op een website in te schrijven voor een nieuwsbrief, zodat ze altijd op de hoogte zijn van de laatste ontwikkelingen.
- De nieuwsbrief zal in het oog moeten springen door de herkenbare huisstijl van de verzender.
- Het systeem moet ook los van het Interpulse Content Management Systeem (CMS) aangeboden kunnen worden.
- Het systeem moet eenvoudig bij externe websites, die niet in het beheer van Interpulse zijn (maar wel gehost), te implementeren zijn.

2.3 Plan van aanpak

Om de uiteindelijke doelstellingen te behalen heb ik een plan van aanpak geschreven voor het gehele project. De onderdelen uit dit document, die bijdrage aan dit eindverslag, zal ik hieronder bespreken.

2.3.1 Opdrachtformulering

De opdracht houdt het volgende in: Een nieuwsbrief beheersysteem ontwikkelen die de genoemde problemen uit de paragraaf 'probleemstelling' op zullen lossen. Met behulp van dit systeem:

- Kunnen geïnteresseerden zich in -en uitschrijven voor de nieuwsbrief;
- Kunnen gebruikers eenvoudig artikelen schrijven;
- Kunnen beheerders eenvoudig nieuwsbrieven aanmaken en verzenden naar geïnteresseerden;
- Kunnen beheerders eenvoudig (e-mail)adressen importeren;
- Kunnen beheerders eenvoudig vormgeving toepassen op de nieuwsbrief.

Het project omvat de definitiestudie, pilotontwikkeling en de invoering van de in de paragraaf 'werkzaamheden' genoemde activiteiten. De nadruk bij het project ligt bij het proces, de systeemdokumentatie is daarom van groot belang voor het slagen van het project. Het uiteindelijke systeem zal daarnaast moeten voldoen aan de wensen van de opdrachtgever, met wie ik veelvuldig contact zal hebben.

2.3.2 informatie en gebruikersgroepen

In overleg met de opdrachtgever hebben we besproken welke gebruikersgroepen in de huidige situatie betrokken zijn bij het proces en hoe dit in de toekomst plaats zal vinden. In deze fase heb ik reeds een schets gemaakt van de nieuwe situatie, dit geeft de opdrachtgever een duidelijk beeld van het idee van de ontwikkelaar ten opzichte van het systeem. Door al in een vroeg stadium te praten over het systeem, wordt de opdrachtgever verder getriggerd om dieper na te denken over dit systeem. Resultaat hiervan was extra informatie. In onderstaande tabel een weergave van gebruikersrollen.

Huidige (oude) situatie	Toekomstige situatie
Ontvanger nieuwsbrief	Ontvanger nieuwsbrief (geïnteresseerde)
Interpulse	Gebruiker
Derde partij	Beheerder
Bedrijf (opdracht verstrekker)	

Interpulse komt in de nieuwe situatie niet meer voor, behalve als gebruiker of beheerder van het systeem. Met deze informatie hebben we onder meer bepaald in hoeverre het contact met deze groepen zou verlopen. De contactmomenten zijn als volgt:

- Week 1 tot week 4: Wekelijks en waar nodig met extra momenten in de oriëntatie fase (opdrachtgever en gebruikers)
- Week 5 tot week 15: Eén maal binnen de pilot en één maal bij einde van de timebox per pilot. Afhankelijk van de omvang van de iteratieslag zal er een extra afspraak gemaakt worden eventueel met gebruikers. (opdrachtgever en gebruikers)
- Week 16: In deze week zal het systeem ingevoerd worden. Hiervoor is een extra moment ingelast met de betrokken gebruikersgroepen. Tijdens dit moment zal er een laatste verificatie plaats vinden. (opdrachtgever en gebruikers)

2.3.3 Kwaliteitseisen en succesfactoren

De kwaliteit zal gewaarborgd worden door het toepassen van prototyping en organiseren van workshops en interviews. Het prototypen van de systeemdelen zal uitlopen in eventuele aanpassingen, zodat er eventueel een extra iteratieslag gemaakt kan worden.

De kwaliteitseisen zijn:

- Een begrijpelijk systeem waar geen aparte cursus voor gegeven hoeft te worden. Het systeem moet intuïtief zijn;
- Moeilijk ervaren componenten moeten gebruikersondersteuning bieden;
- Ingevoerde data mag niet zonder melding verloren gaan;

- Het systeem is alleen toegankelijk voor geautoriseerde gebruikers;
- Het systeem is toegankelijk voor elke geautoriseerde gebruiker met een pc met internet verbinding;
- Het systeem moet snel, binnen 10 seconden (56k6), geladen kunnen worden;
- De systeembeheerder moet door middel van de documentatie en / of het commentaar bij de code inzicht kunnen krijgen in de opbouw van het systeem.

2.4 Gehanteerde methoden en technieken

De mogelijkheid om duidelijk een stappenplan vooraf te definiëren is één van de vele redenen om tijdens het project een methode te gebruiken. Meer motivatie over de keuzes betreffende methoden en technieken zijn te vinden in deze paragraaf.

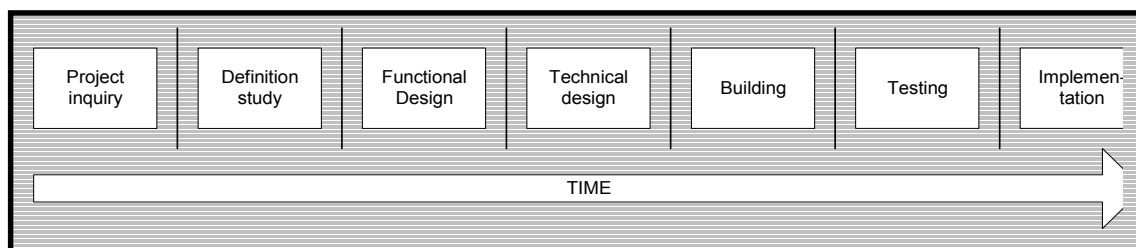
2.4.1 IAD

Iterative Application Development (IAD) is een uitgebreide set van activiteiten om een systeem te ontwikkelen. Iterative, vertaald iteratief, wil zeggen dat de ontwikkelfasen verschillende malen doorlopen worden. De volgende fases komen terug:

- Definitiestudiefase
- Pilotontwikkelingsfase
- Invoeringsfase

IAD versus SDM

Voor ik de keuze heb gemaakt voor een specifieke methode heb ik een tweetal methodes naast elkaar gelegd. In dit onderzoek heb ik vooral gekeken naar SDM en IAD. Wat meteen opvalt is dat SDM slechts één ontwikkelstrategie kent en IAD een viertal strategieën onderkent. Zo is SDM altijd lineair (ook wel waterval) en IAD iteratief.



Figuur 2.1: SDM ontwikkeltraject

Onderstaand zal ik enkele voor -en nadelen aangeven welke voor mij van toepassing zijn binnen het project. Om te beginnen heeft SDM de volgende voordelen:

- Logische geordende volgorde van taken;
- Producten komen door stapsgewijze verfijning tot stand.

Nadelen:

- Doordat alle eisen in het begin vastgesteld moeten worden is er weinig ruimte voor herziening of aanvulling van deze lijst;
- Het lang uitblijven van resultaten kan het vertrouwen bij gebruikers en opdrachtgever ondermijnen;
- Het is, weinig motiverend om continu met papierwerk bezig te zijn. Dit papierwerk levert 'white noise' op, ofwel geritsel van vele stapels papier.

IAD heeft vanzelfsprekend ook een aantal nadelen of valkuilen, maar daarnaast onder andere de volgende voordelen:

- Betrokkenheid van de opdrachtgever en de gebruikers, waardoor de perceptie van de ontwikkelaars snel duidelijk wordt en deze, wanneer nodig, bijgesteld kan worden;
- Snelle resultaten, waardoor hinderlijke knelpunten snel opgelost kunnen worden;
- Betere beheersing van het volledige project, doordat het systeem evolueert naar zijn uiteindelijke vorm.

Nadelen / valkuilen:

- Scope creeping; wanneer de systeemeisen steeds opnieuw bijgesteld worden loopt het project het gevaar nooit af te komen;
- Veel tussentijdse versies kan leiden tot problemen of verliezen bij slechte toepassing van versiebeheer.

Om de keuze voor de methode IAD te motiveren heb ik hieronder een tabel weergegeven met project gerelateerde eigenschappen die een bruikbare methode moet hebben. Daarbij staat een weging vermeld: 1 is niet zo belangrijk en 4 is erg belangrijk.

Project gerelateerde eigenschap	IAD	SDM	Weging
Logische volgorde	++	++	3
Betrokkenheid gebruikers / opdrachtgever	++	+/-	4
Gebruikers blijven motiveren	++	-	3
Flexibiliteit	+	+	1
Evoluerend karakter	++	+	3
Grafische gebruikers interface validatie middels bijvoorbeeld prototyping of andere tussentijdse techniek	++	+/-	2
Bruikbaarheid bij vastgestelde deadline	++	+	3

De methode IAD biedt de mogelijkheid om als gebruiker en opdrachtgever betrokken te zijn bij de ontwikkeling van het systeem. Dit is interessant voor deze groepen, maar helpt ook het systeem aan de eisen te laten voldoen. Eventuele knelpunten zullen door deze nauwe samenwerking in een vroeg stadium opgelost kunnen worden. Toch zijn IAD en SDM en bijvoorbeeld ook DSDM, wat betreft fasering in grote mate aan elkaar gelijk. Bijkomend voordeel is dat de opdrachtgever en ikzelf reeds ervaring hebben met de methode IAD, waardoor ik me niet eerst hoeft te verdiepen in een nieuwe methode waardoor ik tijd

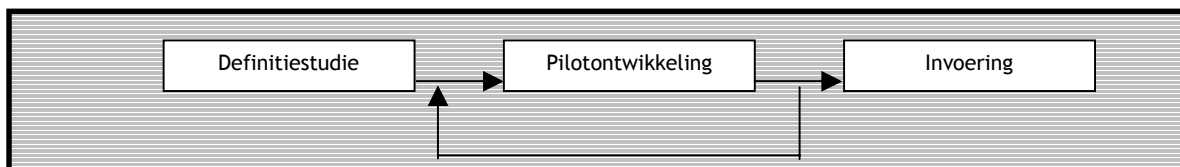
bespaar. Ook de opdrachtgever voelde wel voor de methode IAD. Dit was een belangrijke voorwaarde om de methode IAD succesvol te kunnen gebruiken.

Keuze van strategie

De ontwikkelstrategie welke ik, als ontwikkelaar, het meest geschikt achtte voor het realiseren van het systeem is het incrementeel ontwikkelen. Incrementen zijn kleine bouwsteentjes die binnen de methode evolueren tot het uiteindelijke systeem. De keuze voor deze strategie wordt voor een klein gedeelte mede bepaald door het feit dat ik al ervaring heb met de strategie, daarnaast heb ik de volgende belangrijke redenen:

- De reikwijdte van het project is goed te overzien;
- Deze eisen kunnen goed van tevoren worden gespecificeerd;
- Het systeem moet in één keer ingevoerd worden.

De strategie incrementeel ontwikkelen doorloopt alleen de fase pilotontwikkeling meerdere malen. Schematisch ziet de fasering van incrementeel ontwikkelen er als volgt uit:



Figuur 2.2: incrementeel ontwikkelen

Op het moment dat duidelijk was welke fasen doorlopen zouden worden konden de fasen verder ingevuld worden. Om een indruk te geven van deze fasen zal ik de verschillende fasen kort beschrijven.

Definitiestudie

Het maken van de definitiestudie heeft mij geholpen bij het analyseren van de doelen die ik met het project moest halen, evenals de beperkingen. In deze fase worden de wensen en eisen van de groepen geïnventariseerd en vervolgens geprioriteert. Uiteraard in nauwe samenwerking met de gebruikers en de opdrachtgever. Uit de gesprekken en onderzoeken zullen systeemeisen opgesteld worden.

Na het opstellen van de systeemeisen wordt er al een stap in de richting van de oplossing gezet, in de vorm van een systeemconcept. De onderdelen van dit systeemconcept worden vervolgens geclusterd in coherente eenheden. Hieruit vloeit het pilotplan voort. De keuze om het project dan ook op te delen in werkende stukken, de zogenaamde pilots, was voor mij dan ook een logische stap omdat dit er voor zorgde dat ik stap voor stap een werkend systeem ontwikkelde.

Pilotontwikkeling

De fase pilotontwikkeling wordt meerdere keren doorlopen, per pilot zal er een ontwikkelingsfase worden gestart. Tijdens deze fase buigen de gebruikersgroepen en de opdrachtgever zich gezamenlijk over de pilotdelen, in de meeste gevallen in de vorm van prototypes. In de fase pilotontwikkeling was ik in staat, de eerder gedefinieerde pilots, verder op te delen in kleine incrementen. Deze pilotdelen kunnen gepland worden middels de timebox strategie waar later in dit verslag nog op terug gekomen zal worden.

Invoering

De invoering van het systeem zal ik in één keer bewerkstelligen. Deze manier van invoeren is in overleg met de opdrachtgever gedaan omdat deze het niet zinvol achtte wanneer er een incompleet systeem zou worden ingevoerd. Binnen IAD is dit een goede mogelijkheid, doordat de gebruikers al tijdens de ontwikkeling redelijk vertrouwd raken met het systeem. De bekendheid groeit als het ware met het systeem mee.

2.4.2 GUIDE

De methode IAD biedt een duidelijk stappenplan waarmee het systeem ontwikkeld kan worden. De methode beschrijft een analyse -of oriëntatiefase voor, waarin onder andere de huidige situatie geanalyseerd wordt. In deze definitiestudiefase heb ik ervoor gekozen de technieken van GUIDE te hanteren. GUIDE staat voor Graphical User Interface Design and Evaluation. Ik heb gekozen voor GUIDE, omdat deze techniek uitstekende manieren biedt om taken van gebruikers te analyseren en te modelleren. Uiteindelijk zal het systeem een weergave worden van deze gemodelleerde gebruikerstaken.

Voor dit project heb ik gebruik gemaakt van de taakanalyse in de vorm van taakdiagrammen en taakscenario's zoals GUIDE deze voorschrijft. Daarnaast beschrijft Dhr. Schneidermann in het boek GUIDE een achttal 'gouden regels', welke als richtlijnen gezien kunnen worden om een systeem vorm te geven. Deze design principes heb ik toegepast bij het opstellen van de systeemeisen. Als laatste heb ik gekozen voor het opstellen van een style guide waarin de vormgevingseigenschappen van het systeem vastgelegd worden.

2.4.3 UML

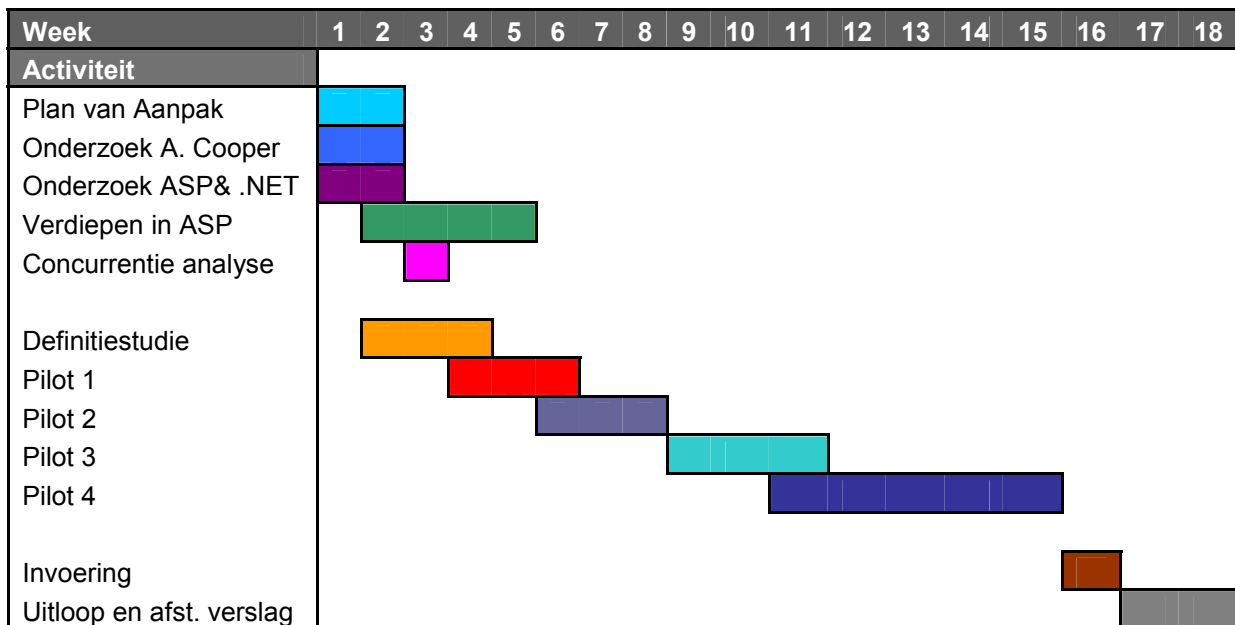
Bij de ontwikkeling van systemen is het belangrijk, al in een vroeg stadium, een beeld weer te geven van het systeem. Een uitstekende manier om dit te bewerkstelligen zijn de modellen van Unified Modeling Language (UML). Voorbeelden hiervan zijn de use case diagrammen en het klassendiagram. UML biedt de mogelijkheid om blauwdrukken te vervaardigen van het systeem op een begrijpelijke en gestandaardiseerde manier.

Daarnaast vond ik het belangrijk het systeem te vervaardigen uit het oogpunt van de gebruiker. Hiervoor heb ik de use-case diagrammen uit UML gehanteerd. Deze use-cases

gaven mij én de gebruikers een duidelijk beeld van de toekomstige situatie, waardoor ze mij, waar nodig, bij konden sturen.

2.5 Planning en uit te voeren activiteiten

Na het bepalen van de methode met de daarbij technieken was ik in staat de activiteiten te plannen. Het plannen heb ik gedaan met behulp van het GANTT-CART diagram. Om het overzicht te bewaren heb ik de 'sub'-activiteiten uit het diagram gelaten.



Figuur 2.3: GANTT CART diagram

2.6 Op te leveren producten

De volgende producten zullen opgeleverd worden tijdens en na afronding van het project.

Op te leveren ontwerp documentatie:

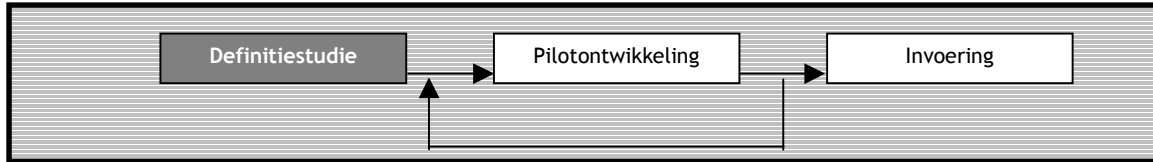
- Plan van aanpak;
- Definitiestudie;
- Pilotontwikkelpannen.

Op te leveren digitale producten:

- Nieuwsbrief beheersysteem;
- Systeemdocumentatie;
- Gebruikershandleiding.

3. Definitiestudie

In de fase definitiestudie heb ik een aantal onderzoeken uitgevoerd, met betrekking tot de huidige situatie en de toekomstige situatie. Uit deze analyse heb ik een systeem geschetst. Belangrijk onderdeel was de planning voor de volgende fase in de vorm van een pilotplan.



3.1 Analyseren van huidige situatie

Om tot een systeem te komen die voldoet aan alle wensen en eisen van de opdrachtgever en de gebruikers, was het zinvol in een vroeg stadium te analyseren welke taken uitgevoerd worden in de oude situatie. Voor deze analyse is er gebruik gemaakt van de technieken van GUIDE. Deze taakanalyse biedt uitstekende mogelijkheden om te achterhalen welke 'doelen' de gebruikers hebben en welke stappen ze daarbij doorlopen. Voor mij als ontwikkelaar is dit belangrijk omdat in het uiteindelijke systeem deze doelen, in de meeste gevallen, terug moeten komen.

3.1.1 Scenario's opstellen

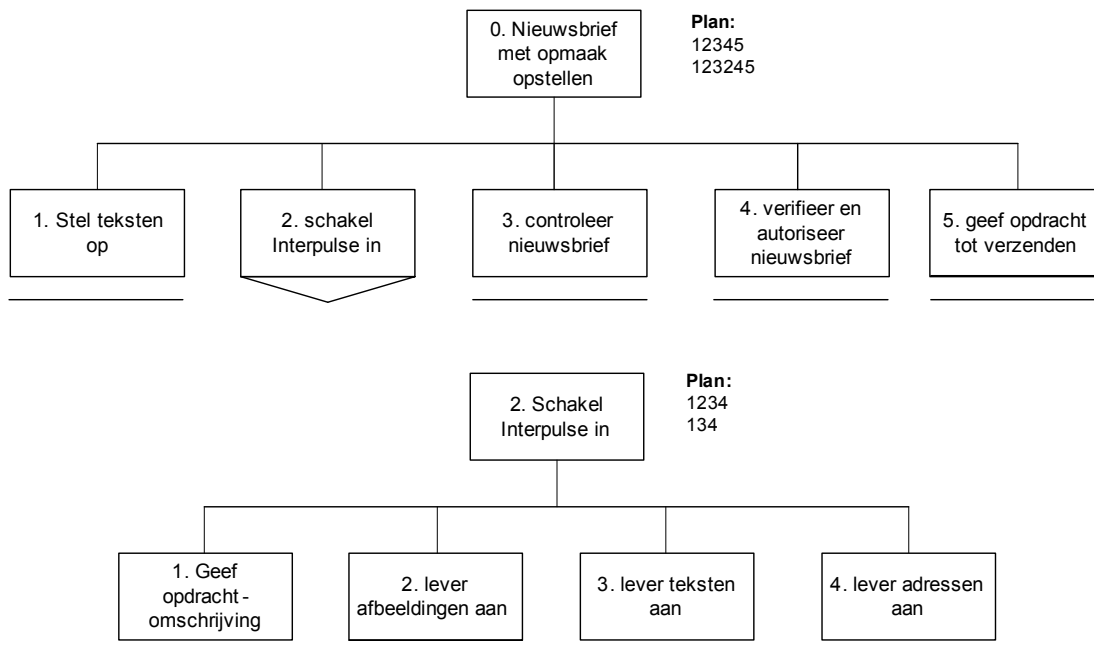
Als eerste heb ik gesprekken gevoerd met de opdrachtgever en de gebruikers. Uit deze gesprekken heb ik scenario's opgesteld en opnieuw, deels via email, voorgelegd aan de betrokkenen. Hieruit kwam een tweede en definitieve versie van de korte scenario's. De scenario's bieden mij de mogelijkheid meer inzicht te krijgen in de manier van werken en denken van de gebruiker. Wanneer de gedachten van de ontwikkelaar en de gebruiker overeenstemmen is de stap tot een sluitend systeemconcept weer dichterbij.

Een voorbeeld van een scenario is hieronder afgebeeld, het gaat hier om de gebruikersgroep (toekomstige) 'gebruikers' die een nieuwsbrief opstellen mét opmaak. Hiervoor moet in de oude situatie Interpulse ingeschakeld worden.

Scenario 3	Opgemaakte mailing opstellen en voorleggen aan Interpulse
Achtergrond:	De gebruiker geeft aan Interpulse aan dat zij een nieuwsbrief op willen stellen en digitaal willen verzenden aan de klanten.
Taak:	De klant geeft aan welke vormgeving toegepast moet worden en levert deze aan. Daarbij leveren zij tevens de teksten van deze nieuwsbrief aan in MS Word. Concept wordt ontvangen van Interpulse, na een laatste controle en verificatie ontvangt de gebruiker deze net als de klanten (geadresseerden) in de mailbox.

3.1.2 Taakdiagrammen opstellen

Een volgende stap was het opstellen van de taakdiagrammen, zoals GUIDE deze beschrijft. Het doel van deze diagrammen is het meer inzicht geven in de concrete taken die in de oude situatie uitgevoerd worden door de verschillende gebruikersgroepen. Per gebruikersgroep heb ik de verschillenden taken gemodelleerd. Een voorbeeld van een taak is het opstellen van een nieuwsbrief met opmaak, in de oude situatie is dit behoorlijk tijdrovend, blijkt ook uit het onderstaande taakdiagram.



Figuur 3.1: Taakdiagram huidige situatie

3.1.3 Probleemanalyse

Nu de scenario's en taken van de betrokkenen duidelijk op een rij staan kunnen de huidige problemen beter geanalyseerd worden. Tijdens de diverse gesprekken met de opdrachtgever en de gebruikers kwamen een aantal hoofdproblemen naar voren.

- Spam;
- Tijdrovend;
- Kosten;
- Eigen beheer.

De nieuwsbrief zoals deze ontvangen wordt door de geïnteresseerden, heeft niet altijd het juiste effect. De reden hiervan is dat ontvangers de nieuwsbrief inschatten als spam. Dagelijks worden er per emailaccount in de meeste gevallen (75%) vele spam berichten ontvangen. Deze worden niet gelezen. Helaas worden de nieuwsbrieven (platte tekst versie) ook als spam ingeschat en regelmatig verwijderd uit de mailbox. Het effect van de nieuwsbrief geeft nog niet het gewenste resultaat voor de (toekomstige) gebruikers.

Het opstellen van een nieuwsbrief met opmaak is een tijdrovende klus voor de gebruikers. Voordat de nieuwsbrief daadwerkelijk de deur uit is, is dit minimaal 8 werkdagen verder. Reden hiervoor is het gehele proces dat doorlopen moeten worden lang is. In eerste instantie moeten de teksten opgesteld worden en de partijen op de hoogte gebracht worden. Deze plannen de werkzaamheden in en wanneer het mee zit kan dit op korte termijn gebeuren. Na het opleveren van een eerste versie moet de gehele nieuwsbrief zowel tekstueel als op het vormgevingsgebied gecontroleerd worden. Het komt vaak voor dat de eerste versie niet volledig naar wens is en hierdoor moet het proces min of meer herhaald worden. Concluderend duurt het proces veel te lang en kunnen hoog geprioriteerde berichten niet altijd op tijd de deur uit.

Door het inschakelen van de derde partij komen er extra kosten voor de 'gebruiker'. Deze kosten kunnen hoog oplopen bij het verzenden van meerdere nieuwsbrieven per jaar. Deze kosten komen voor rekening van de klant. Interpulse heeft daarnaast de opdracht niet geheel in eigen beheer. Door deze afhankelijkheid van de derde partij worden zij aangesproken op 'fouten' of vertraging van deze derde partij. Daarnaast heeft Interpulse zich te houden aan de eisen van de derde partij. Hier valt te denken aan de aan te leveren bestandsformaten en de indeling van de vormgeving.

3.2 Uitvoeren doelgroep analyse

Om een beter beeld van de toekomstige gebruikers te vormen heb ik ervoor gekozen om een doelgroeponderzoek uit te voeren. Ik heb gekozen voor de persona techniek die Alan Cooper ontwikkeld heeft op basis van een oude techniek die gebruikt werd in de reclame wereld.

3.2.1 *Waarom persona's?*

Vaak heeft men het in het ontwerp- en bouwproces over 'de gebruiker' die bepaalde dingen wel of niet zou willen of zou begrijpen. Maar dat is eigenlijk veel te abstract. Over welke gebruiker hebben we het? Het idee van mijzelf van wat de gebruiker weet, kan en verwacht is misschien wel veel te veel beïnvloed door mijn eigen achtergrond en ervaring.

Er dreigt bovendien het gevaar van de spraakverwarring: de opdrachtgever heeft misschien een heel andere gebruiker in het hoofd dan de ontwikkelaar. En de ideeën van beide personen kunnen wel heel ver af liggen van de werkelijke verwachtingen van de gebruiker. Persona's kunnen helpen om in de huid te kruipen van één of meerdere bezoekers van het systeem. Ze kunnen bovendien helpen bij een goede communicatie tussen de verschillende disciplines die meewerken aan het project.

Het belangrijkste van deze techniek is namelijk dat de betrokkenen een gedeeld idee krijgen voor wie we het allemaal doen, voor wie het systeem gemaakt wordt. Dat lukt niet als bijvoorbeeld alleen de accountmanagers de persona's verzint en ze vervolgens presenteert

aan de rest van het team. Om deze reden heb ik, in de vorm van een brainstormsessie, met de gebruikers en in samenwerking met de opdrachtgever een drietal persona's ontwikkeld.

3.2.2 Doel van de techniek

De persona's moeten alle beslissingen die genomen worden sturen. Bij elke beslissing over het systeem moeten we, als betrokkenen, de mensen waar we het voor doen in ons achterhoofd houden. Een voorbeeld kan zijn een beslissing van lettergrootte of dit zou de keuze van de interface mogelijkwijs beïnvloeden. Wanneer er gekozen wordt voor lettergrootte 10px, kan dit problemen geven met persona X wanneer deze bijvoorbeeld 65+ is. Een voorbeeld van één van de drie personen, zoals we deze geschetst hebben, staat in onderstaande tabel.

Persona 3	
Persona: Kees van Zelst	
	<p>Naam: Kees van Zelst Woonplaats: Leiden Leeftijd: 56</p>
<p>Voor het bestuur van de stichting 'Leiden ontzet' verricht Kees veel werk. Hij besteedt er wekelijks vele uren aan, naast zijn werk als boekhouder bij een notariskantoor. Vooral in de maanden augustus en september is het druk met de werkzaamheden rond 'Leiden ontzet'. 3 oktober is het uiteraard een groot feest, met haring en wittebrood.</p> <p>Kees verzorgt voor de stichting de uitnodigingen en informatiebladen richting sponsors. Hij typt sinds een jaar zijn teksten uit in MS Word en kent enkele basisfuncties, zoals dikgedrukt en onderstreept. Kees heeft niet zulke goede ogen en heeft soms moeite met het lezen van teksten vanaf het scherm.</p>	

3.3 Onderzoek naar ASP en ASP.NET

Het doel van dit onderzoek is het onderzoeken van de programmeertalen ASP en ASP.NET. Hieruit zal uiteindelijk een weloverwogen voorkeur komen. In dit onderzoek zullen we de talen Active Server Page (ASP) en ASP.NET betrekken. Andere talen, zoals PHP zullen buiten dit onderzoek gelaten worden doordat Interpulse enkel gebruik maakt van de Microsoft producten.

3.3.1 Het onderzoek

De beide talen heb ik naast elkaar gelegd. Door test scriptjes te maken en een literatuurstudie uit te voeren, op diverse internet bronnen en met hulp van enkele boeken,

heb ik beide talen met elkaar vergeleken. In onderstaande tabel heb ik de eigenschappen, welke van belang zijn voor het project, op een rij gezet. Daarbij heb ik de weging vermeld, waarbij 4 zwaarwegend is en 1 nauwelijks belangrijk is. De eigenschappen zijn tot stand gekomen door gesprekken met de opdrachtgever en met collega's van Interpulse.

Eigenschappen	ASP	.NET	Weging
Database connectie (SQL server)	++	++	4
Form handling	++	++	4
Serverside scripting	++	++	1
Stabiliteit	++	++	2
Communicatie SMTP server	++	++	4
Beveiliging	++	++	3
Moeilijkheidsgraad (-- is moeilijk)	+	--	3
Debugging	+/-	+	2

Wanneer we de benodigdheden vergelijken met de mogelijkheden van de beide scripttalen kom ik tot de conclusie dat ze beide volstaan. ASP.NET kan gemakkelijk aan de eisen voldoen, maar ASP kan dit ook. Hierdoor lijkt de weg vrij om ASP.NET toe te passen, maar is dit wel zo verstandig?

3.3.2 *Ervaring collega's*

Uit de gesprekken met de collega's en uit mijn vooronderzoek blijkt ASP veel te lijken op PHP, het open-source alternatief. Deze taal beheers ik al redelijk en is niet heel moeilijk om aan te leren. ASP.NET daarentegen is veel uitgebreider, maar ook vele malen moeilijker om aan te leren. Deze taal lijkt veel meer op de 'echte' programmeertalen, zoals C++ of Delphi. De collega's raden mij dan ook aan ASP te gebruiken.

3.3.3 *Eindconclusie*

Met het bronnenonderzoek en de gesprekken met de collega's in het achterhoofd heb ik samen met de opdrachtgever besloten ASP te gebruiken. Deze taal kan aan de gestelde eisen voldoen en is eenvoudiger te leren dan het uitgebreide ASP.NET. Daarnaast kunnen bij een eventuele (toekomstige) overstap naar ASP.NET de scriptfiles hergebruikt worden. De ASP.NET server die bij Interpulse gebruikt wordt kan gewoon ASP pagina's 'lezen'.

3.4 **Onderzoeken van e-mail marketing**

Doordat het systeem uiteindelijk de mogelijkheid biedt om nieuwsbrieven per e-mail te verzenden vond ik het raadzaam me te verdiepen in de wereld van emailmarketing. Want, reclame per e-mail is geen enkel probleem indien de ontvanger daar expliciet om heeft gevraagd, maar hoeveel mensen ergeren zich niet aan spam. Wanneer een bedrijf die gebruik maakt van het nieuwsbrief beheersysteem geassocieerd wordt met spammers dan kan dit

slechte gevolgen hebben en heeft de nieuwsbrief dus een negatief effect. In het ergste geval kan Interpulse daar ook mee geassocieerd worden. Binnen de emailmarketing worden een tweetal principes gehanteerd, namelijk opt-out en opt-in. Een gedeelte van het onderzoek heb ik hieronder afgedrukt.

3.4.1 *Opt-out*

Rechtszaken hebben spammers de mogelijkheid geboden om reclame per e-mail te sturen naar gebruikers die daar niet om hebben gevraagd, mits deze 'geïnteresseerden' de mogelijkheid wordt geboden zich uit te schrijven voor de ontvangst van verdere reclame. Bij opt-out kan in principe iedereen een ander inschrijven. Sterker nog, bij opt-out kan de spammer zelf iemand inschrijven. Dit eerste principe heet opt-out en is dus eigenlijk zeer ineffectief.

Als je na het ontvangen van een spambericht via opt-out aangeeft voortaan geen email van het betreffende bedrijf te willen ontvangen, is er geen garantie dat je adres daadwerkelijk wordt verwijderd uit de database. Sommige spammers gebruiken zo'n uitschrijving slechts als bevestiging dat het adres actief gelezen wordt. Zulke adressen zijn voor spammers extra interessant en door die bevestiging wordt het adres dus alleen maar waardevoller voor hen. Oneerlijke spammers kunnen ervoor kiezen dat adres later opnieuw spam toe te sturen, of zelfs om het adres op een cd te branden en het aan anderen te verkopen. Reden te meer om deze onbetrouwbare methode niet te gebruiken binnen het nieuwsbrief beheersysteem.

3.4.2 *Opt-in*

Tegenover het opt-out principe staat het opt-in principe. De ontvanger neemt dan zelf het initiatief en geeft aan dat hij de reclame per e-mail wil ontvangen. Dat kan bijvoorbeeld naar aanleiding van een bezoek aan de website zijn. Maar opt-in alleen blijkt niet genoeg. Op die manier kan iedereen, ook de spammer zelf, elk willekeurig adres aanmelden. De enige betrouwbare oplossing is, naar mijn mening, dan ook confirmed opt-in.

3.4.3 *Confirmed opt-in*

Na aanmelding krijg je als eigenaar van het emailadres een verzoek om bevestiging opgestuurd met daarin de vraag of je daadwerkelijk wilt aanmelden voor de mailinglist (of voor een nieuwsbrief). Indien de eigenaar van het bewuste emailadres geen bevestiging terugstuurt, moet er vanuit gegaan worden dat iemand anders dat adres heeft opgegeven, of dat de eigenaar zich heeft bedacht. Indien de houder van dat emailadres de vraag bevestigt, geeft hij expliciet aan dat hij instemt met de ontvangst van e-mail via die mailinglist.

Het is dus de bedoeling dat bij een inschrijving de gebruiker expliciet toestemming geeft voor het ontvangen van e-mail van een mailinglist. Geeft hij die toestemming niet en ontvangt hij wel e-mail van die mailinglist, zijn die e-mails op te vatten als spam.

3.5 Uitvoeren concurrentie analyse

Tijdens de definitiestudiefase heb ik, om een beeld te krijgen van de concurrenten en mogelijke functionaliteiten te achterhalen, een onderzoek gedaan naar de systemen van concurrenten. Het analyseren van de systemen is zeer zinvol bij het opstellen van de systeemeisen.

Enkele functionaliteiten hebben mij getriggerd en op nieuwe ideeën gebracht. Een voorbeeld van een functionaliteit is de mogelijkheid om speciale tekens toe te voegen, welke ik uiteindelijk heb overgenomen. Daarnaast is het niet onbelangrijk om minstens een vergelijkbaar product neer te zetten naast het product van de concurrent.

Van de 4 bedrijven welke ik onderzocht heb, geef ik hieronder de 'winnaar' die uit het onderzoek kwam. M-Acquaint (www.macquaint.com) is een handelsmerk dat onderdeel is van Myriadm. Onder de productnaam M-Acquaint bieden ze een nieuwsbrief systeem aan. Op dit moment is versie 2.6 online. Een aantal grote namen maken gebruik van dit systeem, zoals Alcatel en Golden Tulip. De prijzen van het systeem zijn als volgt opgebouwd:

Abonnementsvorm	Aantal nieuwsbrieven	Kosten per jaar
Basic	5000	€ 500,00
Standard	20.000	€ 1.000,00
Professional	40.000	€ 1.500,00
Large	80.000	€ 2.000,00
Custom	80.000 +	in overleg

Alle genoemde prijzen zijn exclusief BTW. De kosten voor een abonnement worden gefactureerd bij het aangaan van het abonnement. U betaalt hierbij een éénmalig opstartbedrag van € 100,00

Enkele voorbeelden van functionaliteiten die aangeboden worden heb ik hieronder afgebeeld:

- Opmaak sjablonen
- WYSIWYG-editor
- Afbeeldingencatalogus
- Speciale tekens
- Importermogelijkheden adresbestand
- Exportermogelijkheid adresbestand
- Response bekijken
- Helpstelsysteem

Het systeem van M-Acquaint ziet er goed uit en beschikt over veel functionaliteiten. Uit het onderzoek komt dit systeem dan ook als 'winnaar' uit de gehouden concurrentie analyse.

3.6 Opstellen usability checklist

Aandacht besteden aan de interface vind ik erg belangrijk bij het ontwikkelen van het systeem. Het systeem moet voor iedere gebruiker begrijpelijk en benaderbaar zijn. Jacob Nielsen heeft een groot aantal richtlijnen opgesteld in diverse boeken. Deze heb ik gedocumenteerd in de vorm van een checklist.

Bij de ontwikkeling van het systeem zal de checklist bovenop de stapel van papieren moeten liggen, zodat te allen tijde rekening gehouden kan worden met de inhoud hiervan. Een onderdeel van de checklist heb ik hieronder afgebeeld. De V staat voor in orde, de X voor niet gelukt.

Richtlijn	V	X
Use percentages rather than fixed pixel number for page and table width--this will adapt the web page to different screen sizes and resolutions to avoid the hated horizontal scrolling.		
Images need to have width and height attributes.		
Images no larger than 544 pixels wide by 376 pixels tall or they won't fit on WebTV screens.		
Use frames sparingly--they take away the ability to bookmark or create a link to a specific page.		
Break long text documents into theme related chunks on multiple pages for ease of reading		
Provide printer friendly version of long text documents (no frames, all on one page).		
Text should mostly be left justified for ease of scanning/reading.		
Web page titles (what shows on top browser bar) should be concise, 2 to 6 words long.		
Web page titles should start with content specific, unique words that are easy to scan on a list of bookmarks or search engine results (i.e. don't start all page titles "UGA Libraries...)		

Uit de lijst komt ondermeer naar voren dat er percentages gebruikt moeten worden in plaats van absolute waarden, met betrekking tot de lettergrootte en dergelijke. Deze werkwijze heeft als voordeel dat de pagina geschikt is voor iedere schermresolutie en daarnaast leesbaar voor iedere type gebruiker. Wanneer ik bijvoorbeeld rekening houdt met Kees, een van de persona's, komt dit goed uit. Hij zal hierdoor nooit onleesbare teksten voorgeschoteld krijgen. In hoofdstuk 4.4.2 kunt u een ingevuld voorbeeld vinden van een aantal opgestelde richtlijnen.

3.7 Opstellen systeemeisen

In deze paragraaf zal ik de systeemeisen beschrijven waaraan het gehele systeem moet voldoen. De eisen zijn opgedeeld in basissysteem eisen, interface-eisen, integriteiteisen, performance-eisen en de operationele eisen. De systeemeisen vormen het belangrijkste referentiemateriaal bij een volgende iteratie, wanneer getoetst wordt in hoeverre de ontwikkelde pilots werkelijk aan de afgesproken eisen hebben voldaan. De geprioriteerde eisen zijn achterhaald door middel van diverse gesprekken en een workshop met de opdrachtgever en een gebruiker.

3.7.1 Creatieve workshop

Tijdens de workshop hebben we functionaliteiten bedacht, door middel van de think aloud (dwingen tot hardop denken) techniek en elkaar vervolgens te triggeren door de resultaten op een whiteboard te noteren. Uit deze workshop kwam de volgende lijst van functionaliteiten:

Functionaliteiten	
1. WYSIWYG - editor	9. Html en tekst versie van nieuwsbrief
2. Artikel beheer module	10. Vormgeving met uitstraling van commercieel product
3. Template keuze module (opmaak)	11. Click through rate
4. Verzenden van nieuwsbrief	12. Twee manieren uitschrijving, dmv mail en de website
5. Importeer module (uit excel)	13. Dubbele email adressen verwijderen
6. Gebruikersondersteuning (bv. Wizard) bij moeilijk ervaren taken	14. Add embedded image (offline afbeeldingen weergave)
7. Gebruikersbeheer (groepen?)	15. Wizard om zelf template te bouwen
8. Personaliseren email	16. Kleuren etc aangeven

Vervolgens heb ik iedere deelnemer een A4 gegeven met een tabel waarin de kolommen basis, comfort en luxe worden weergegeven. Met een duidelijke toelichting van wat deze indeling exact betekend, heb ik individueel de opdracht gegeven de functionaliteiten in een kolom te voegen.

Als laatste heb ik gevraagd een toelichting te geven op de keuzes en daarbij extra aandacht besteed aan de grote verschillen, waarom persoon A een functionaliteit in categorie basis en waarom persoon B in luxe. Enkele personen wijzigde het formulier, na de discussie, nog eenmaal. Het resultaat is hieronder afgebeeld, de nummers refereren naar de hierboven vermelde functionaliteiten.

Basis			Comfort			Luxe		
Opdr.gever	Gebruiker	Ontwik.	Opdr.gever	Gebruiker	Ontwik.	Opdr.gever	Gebruiker	Ontwik.
<u>1</u>	<u>1</u>	<u>1</u>	6		6	<u>11</u>	<u>11</u>	<u>11</u>
<u>2</u>	<u>2</u>	<u>2</u>	<u>8</u>	<u>8</u>	<u>8</u>	13		13
<u>3</u>	<u>3</u>	<u>3</u>	<u>9</u>	<u>9</u>	<u>9</u>	<u>15</u>		<u>15</u>
<u>4</u>	<u>4</u>	<u>4</u>	10	10				
<u>5</u>	<u>5</u>	<u>5</u>						
6	<u>7</u>	<u>7</u>						
<u>7</u>	<u>12</u>	10						
<u>12</u>	<u>14</u>	<u>12</u>						
13	<u>16</u>	<u>14</u>						
<u>14</u>		<u>16</u>		15			<u>15</u>	
<u>16</u>								

Deze gegevens resulteerde uiteindelijk in input voor het opstellen van de systeemeisen, welke ik in de volgende paragraaf zal bespreken.

3.7.2 Systeemeisen opstellen en prioriteren

Met de input van de voorgaande onderzoeken en workshop was ik in staat per categorie systeemeis een uitgebreide tabel samen te stellen. Ik heb er voor gekozen reeds in deze fase te prioriteren, zodat al een beeld gegeven kan worden gevormd van wat de werkelijke behoeften van de gebruikers en de opdrachtgever zijn en welke eisen meer tot extra's gekwalificeerd kunnen worden. De volgende tabel geeft een weergave van de basis systeemeisen:

Basis
<ul style="list-style-type: none"> - Systeem moet te openen zijn in Internet Explorer 5 en hoger - Systeem moet te openen zijn met een Mac en een Personal Computer - Er moet een koppeling bestaan tussen het systeem enerzijds, en de databases anderzijds - Systeem moet te bedienen zijn met behulp van 'wat je ziet is wat je krijgt', zoals MS Word - Het systeem moet de mogelijkheid bieden een nieuwsbrief te verzenden naar geselecteerde adreslijsten - 'Geïnteresseerden' die de nieuwsbrief niet op prijs stellen moeten zich uit kunnen schrijven. - Personen die geïnteresseerd zijn, moeten zich aan kunnen melden op de website - Nieuwsbrief moet offline te lezen zijn met behoud van opmaak en afbeeldingen. - Beheerders moeten reeds verstuurd nieuwsbrieven kunnen raadplegen - Het systeem mag alleen toegankelijk zijn voor geautoriseerde gebruikers
Comfort
<ul style="list-style-type: none"> - Systeem moet cross browser zijn, zodat Netscape, Safari en Internet Explorer het systeem kan benaderen.

- Een nieuwsbrief moet in samenwerkingsverband, maar op verschillende plaatsen, geschreven kunnen worden.
- Het systeem moet de mogelijkheid bieden om de nieuwsbrief gepersonaliseerd te versturen naar de geïnteresseerden
- Ontvangers die geen html versie willen of kunnen ontvangen moeten een alternatief krijgen
- De beheerder van het systeem moet de gebruikers van het systeem kunnen beheren

Luxe

- Browsers met extra mogelijkheden, zoals Internet Explorer, moeten extra functionaliteiten bieden ten opzichte van de browsers met minder mogelijkheden.
- De verzender moet op de hoogte gebracht kunnen worden van de response van de mailing
- De beheerder moet in staat zijn om zelfstandig een opmaak template voor de nieuwsbrief te realiseren

Zoals in de oriëntatiefase al besproken is, kleven er aan de methode IAD ook nadelen. Eén van de nadelen is het zogenaamde 'scope-creeping'. Om te voorkomen dat het project oneindig in ontwikkeling blijft heb ik met de opdrachtgever de harde afspraak gemaakt dat de lijst van systeemeisen niet meer uitgebreid kan worden. Wanneer dit, in uiterst geval, toch gebeurt zal het onderdeel als luxe eis worden opgenomen.

3.8 Beschrijven systeemconcept

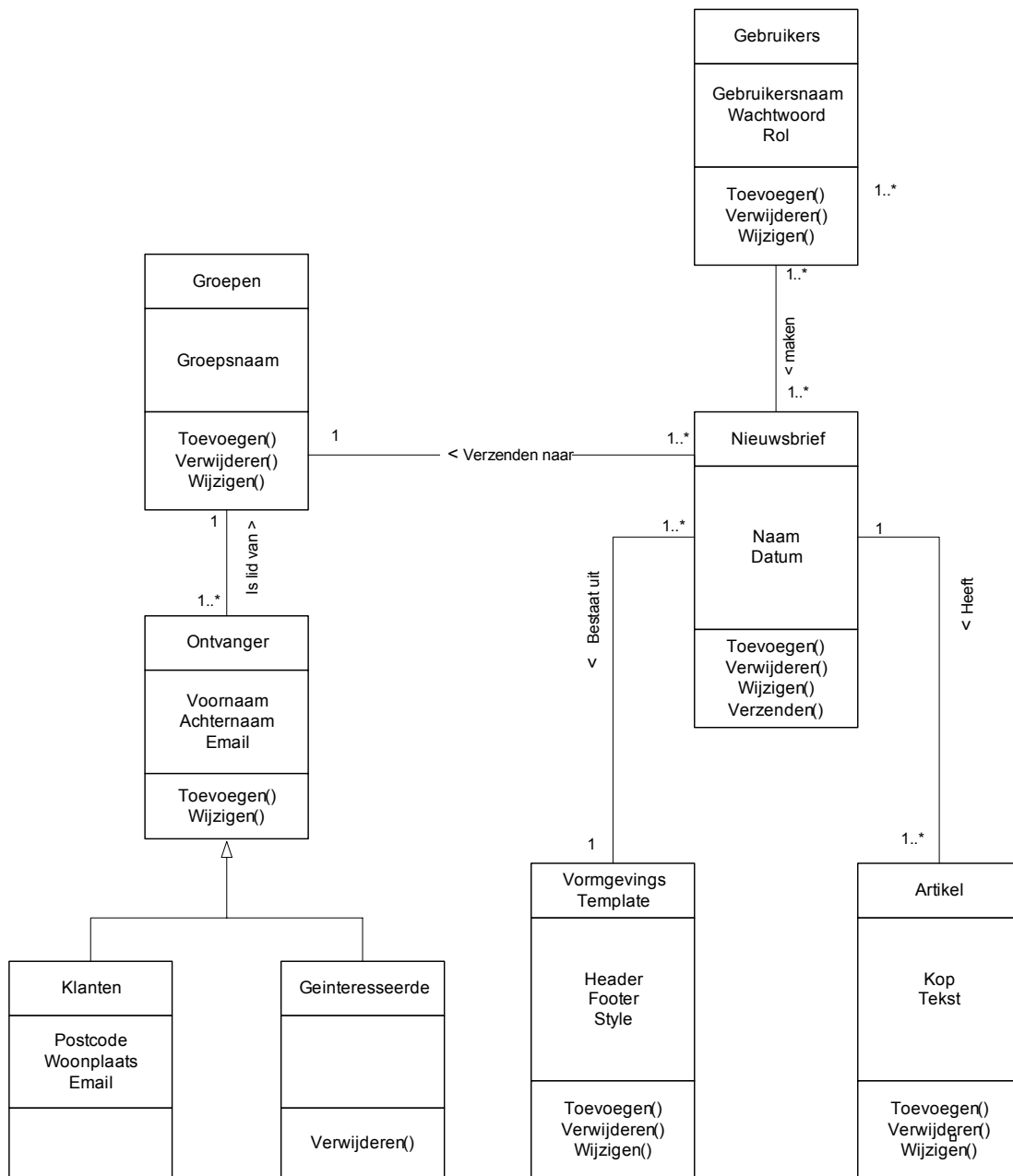
Een volgende stap binnen de definitiestudiefase was het opstellen van het systeemconcept. Het op papier vastgestelde concept beschrijft op globaal niveau de oplossing van de probleemstelling zoals deze bij aanvang van het project is vastgesteld. Het systeemconcept is vanuit het oogpunt van de gebruiker opgesteld en wordt uitgedrukt in taken van deze gebruikers ofwel actoren.

3.8.1 Klassendiagram

Het klassendiagram geeft de mogelijkheid het systeem te representeren op de werkelijkheid en biedt de mogelijkheid om op het niveau van de gebruikers mét de gebruikers te praten. In de fase definitiestudie heb ik er voor gekozen de objecten die betrokken zijn bij het systeem in een vroeg stadium van het project te signaleren en te modelleren in de vorm van een klassendiagram. Deze weergave gaf mij al vroeg in het project een duidelijk beeld van het te ontwikkelen systeem en mede hierdoor was ik in staat om eenvoudiger met de opdrachtgever te praten.

Uit de verschillende gesprekken met de opdrachtgever kwamen steeds dezelfde objecten (zelfstandige naamwoorden) naar boven die ik verwerkte in het klassendiagram. Met dit model had hij ten eerste het gevoel dat er al iets concreets op tafel lag en ten tweede het idee had dat zijn bewoordingen op de juiste manier geïnterpreteerd werden. Als laatste kwam hij mede door het model op allerlei nieuwe ideeën. Bijkomend voordeel bij het

gebruik van dit model was de herkenbaarheid bij de opdrachtgever. Deze wist globaal de betekenis van dit model en kon mijn inzichten met behulp van het model begrijpen.

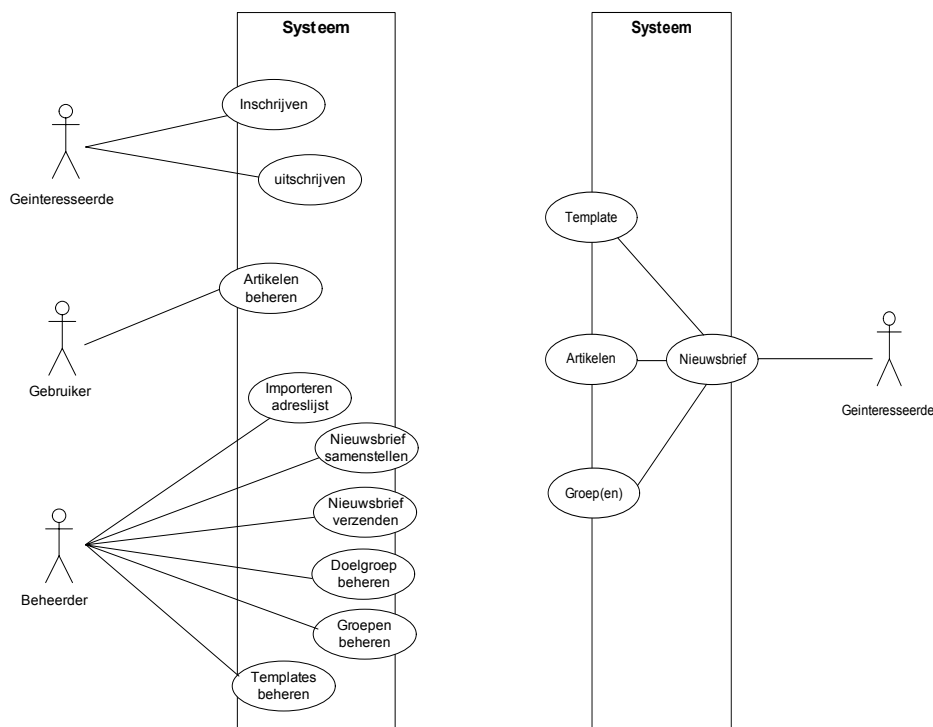


Figuur 3.2: Klassendiagram

Een belangrijke nadruk bij dit klassendiagram is de globale invalshoek van dit model. Het klassendiagram diende uitsluitend als weergave van het systeemconcept. In een later stadium in het proces zal dit model vervangen worden door het EER-model. In het EER-model zullen wijzigingen en aanvullingen voorkomen ten opzichte van dit klassendiagram. Het klassendiagram is daarbij de belangrijkste input voor het EER-model.

3.8.2 Use case diagram

Het use case diagram geeft vervolgens aan hoe het gedrag van de gebruiker is ten opzichte van het systeem. Use cases zijn een waardevol instrument om systeemeisen te verzamelen vanuit het gezichtspunt van de gebruikers. De actoren: geïnteresseerde, gebruiker en beheerder, voeren een reeks van gebeurtenissen uit op het systeem. Het samenstellen van de nieuwsbrief resulteert in de output zoals deze in de rechter use case is gemodelleerd. Door de verschillende gebruikersgroepen te verweven met het systeem, heb ik ervoor proberen te zorgen dat niet alleen 'computerfreaks' het systeem kunnen gebruiken, maar ook de echte gebruikers.



Figuur 3.3: Use cases

3.8.3 Style Guide

In de beknopte style guide wordt er een korte beschrijving gegeven van de GUI zoals die in de applicatie gebruikt zal worden. Er wordt besproken welke kleuren gebruikt gaan worden en hoe de GUI wat betreft de navigatie er uit zal zien.

Belangrijke eigenschap is de Interpulse herkenbaarheid. Hiervoor hanteert men een standaard 'header' die ik ook zal toepassen in het nieuwsbrief beheersysteem.



Figuur 3.4: Interpulse header

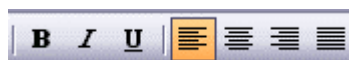
Wat betreft de typografie en lettergroten zal ik de onderstaande eigenschappen hanteren. De pixelgrootte zal ik in het systeem opnemen in percentages, dit om het systeem voor iedere resolutie toegankelijk moet zijn en dit afgedwongen moet worden voor de persona Kees.

Vormgevingseigenschappen van het systeem	
Type	Eigenschappen
Body, table	11px - Verdana, Arial, Tahoma - zwart (black) (85%) Font voor Mac ondersteuning: Helvetica,Sans-serif
Header 1	16px - Verdana, Arial, Tahoma - blauw (#33619F) (110%) Font voor Mac ondersteuning: Helvetica,Sans-serif
Header 2	14px - Verdana, Arial, Tahoma - grijs (gray) (95%) Font voor Mac ondersteuning: Helvetica,Sans-serif
Tekstveld	11px - Verdana, Arial, Tahoma - zwart (black) (85%) Font voor Mac ondersteuning: Helvetica,Sans-serif
Link (a)	Als body, + onderstreept (80%) mouse over: rood (red)
Navigatiebalk	10 px - Verdana, Arial, Tahoma - grijs (gray) -onderstreept - (80%) mouse over: rood (red) - Font voor Mac ondersteuning: Helvetica,Sans-serif

Uit de tabel komt onder andere naar voren dat ik een speciaal font voor de Mac gebruik. De Mac ondersteunt de overige fonts niet en geeft daarom op deze manier de teksten op een gegarandeerd duidelijk font weer.

Een andere belangrijke eigenschap van het systeem is de taakbalk. Volgens de theorieën van Alan Cooper heeft een herkenbare structuur veel waarde voor de bruikbaarheid van het systeem. In zijn *"About Face, The Essentials of User Interface Design"* beschrijft Alan Cooper de verschillen tussen "Recall en recognition". Recognition (herkenning) is voor de gebruiker veel makkelijker dan Recall (herinneren).

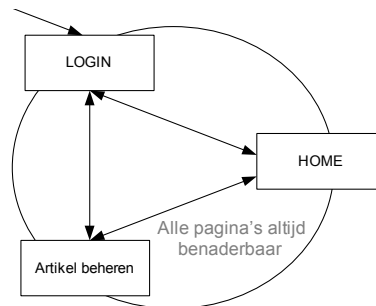
Herkenbare onderdelen als werkbalken en andere Microsoft 'standaards' komen bekend over bij de gebruikers van het systeem. Door de herkenning ontlast het systeem het geheugen en kan deze zich richten op andere onderdelen. Daarom heb ik er voor gekozen een MS Word stijl te creëren bij het opstellen van artikelen. Een voorbeeld van een MS Word taakbalk ziet er als volgt uit:



Figuur 3.5: MS Word stijl

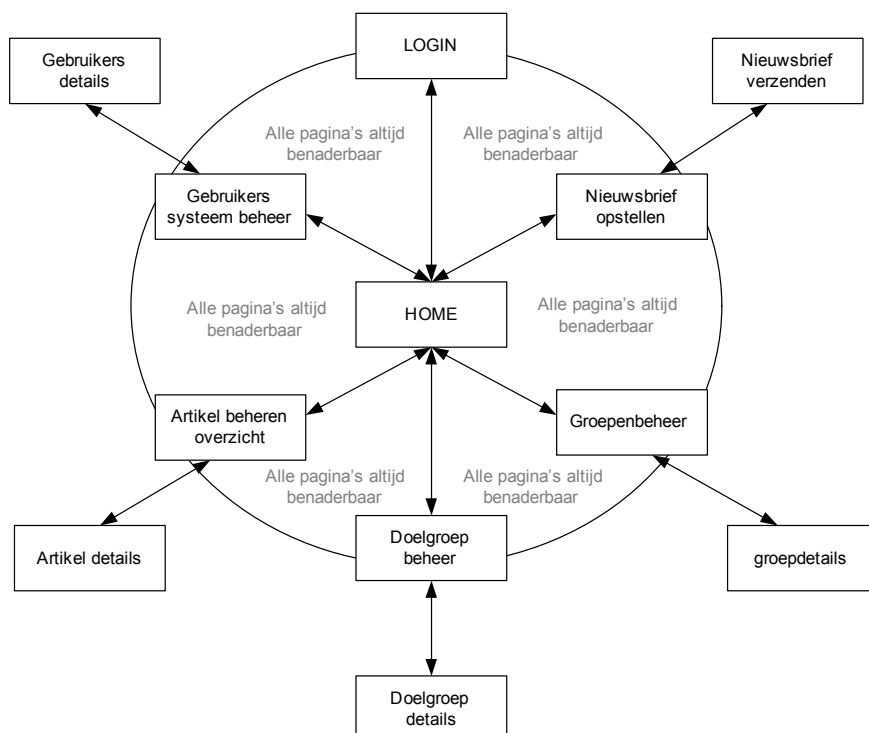
3.8.4 Navigatieschema's

Door het toepassen van autorisatie zijn er twee verschillende navigatieschema's, namelijk voor de beheerders en de gebruikers. De geïnteresseerde heeft geen toegang tot het systeem. De gebruikers en beheerders beginnen altijd bij de login pagina. Het navigatiesysteem voor de gebruiker is beknopt en ziet er als volgt uit:



Figuur 3.6: Navigatieschema 1

Voor de beheerder ligt de gehele weg open, deze heeft toegang tot het gehele systeem. Ook de beheerder komt binnen bij het login scherm. Na het inloggen komt de beheerder op de home pagina, waarvandaan de overige pagina's geopend kunnen worden. Het model moet vanaf home van binnen naar buiten gelezen worden.



Figuur 3.7: Navigatieschema beheerder

3.9 Opstellen pilotplan

De pilotindeling, zoals deze hieronder vermeld is, is tot stand gekomen aan de hand van de systeemeisen en het beschreven systeemconcept. De volgorde is van boven naar beneden.

- Database en autorisatie
- Artikelontwerp systeem
- (Doel)groep beheer
- Nieuwsbrief samenstel -en verzendsysteem

Deze volgorde is op deze manier tot stand gekomen, doordat iedere bovenliggende pilot afhankelijk is van de daaropvolgende pilot. Volgens de IAD methode is elke pilot een op zichzelf werkend stukje software die samen met de andere pilots het systeem vormen. De pilot database zou aan deze eigenschap niet voldoen. Toch heb ik gekozen voor deze pilotindeling, omdat de database een belangrijk onderdeel van het systeem is. Zonder de database is geen enkele pilot te realiseren. Daarbij is de doorlooptijd database van dien aard dat deze naar mijn mening wel geschikt is als losse pilot.

3.9.1 Timeboxing

Timeboxing betekent effectief dat tijd en geld worden gefixeerd en dat functionaliteiten worden gevarieerd. Het beheersen van functionaliteiten gebeurt door middel van prioriteitstelling. Initieel wordt er een lijst opgesteld met prioriteitstellingen, zoals IAD deze hanteert. Op basis van deze lijst wordt een begroting gemaakt, die vervolgens wordt gefixeerd. Bij optredende verschuivingen wordt niet meteen gegrepen naar het aloude middel van meer tijd/geld. Nee, er wordt middels de prioriteitenlijst bepaald wat dan wel binnen de grenzen blijft horen. In het project is geen sprake van 'financiële begrotingen', maar de deadline van het project staat wel vooraf zwart op wit, daardoor is deze techniek zeer bruikbaar om het project te beheersen.

3.9.2 Juicy bits first

Door de bouweenheden te sorteren en te categoriseren op prioriteit worden de meest belangrijke én vooral aansprekende onderdelen als eerste ontwikkeld. Hierdoor kunnen er optimale resultaten geboekt worden binnen de gestelde tijd, zelfs wanneer het einde van de timebox is bereikt voordat alles is voltooid. Hierdoor kan de opdrachtgever te allen tijde, maar vooral in een vroeg stadium, tevreden gesteld worden. Hij zal hierdoor enthousiast worden en werkt hij makkelijker mee wat betreft het contact.

3.9.3 80 / 20 regel

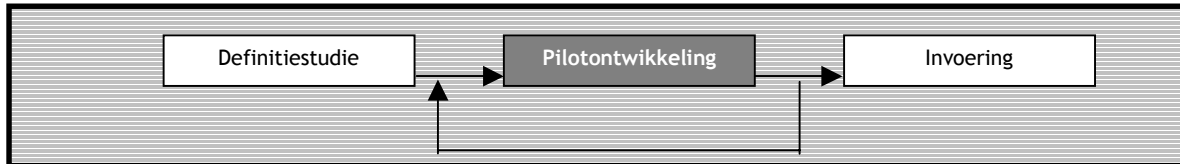
De 80 / 20 regel stelt dat slecht 20% van de tijd benodigd is om 80% van de functionaliteiten van het systeem te realiseren. De overige 80% is benodigd om de overige 20% te realiseren. Door de pilotdelen te benoemen met een prioritering zal ik aangeven welke bouwstenen belangrijk zijn en welke minder.

3.9.4 Prototyping

Door de pilot tussentijds regelmatig voor te leggen aan de opdrachtgever kan er feedback meegenomen worden in de ontwikkeling. Tijdens de bouw zal, na afronding van variërend één tot drie pilotdelen, een contactmoment zijn met de opdrachtgever. Deze variatie is afhankelijk van de grootte van het pilotdeel.

4. Pilotontwikkeling

Na volledige goedkeuring en overeenstemming van het pilotplan brak de volgende fase aan in het ontwikkeltraject. De fase pilotontwikkeling bestaat ook weer uit een drietal fases met mogelijke iteratieslagen. In de opeenvolgende paragrafen zal ik steeds enkele stappen uit het proces beschrijven. Elke actie zal ik éénmaal beschrijven, maar deze acties zijn per pilot uitgevoerd.



4.1 Pilotontwikkeling pilot database

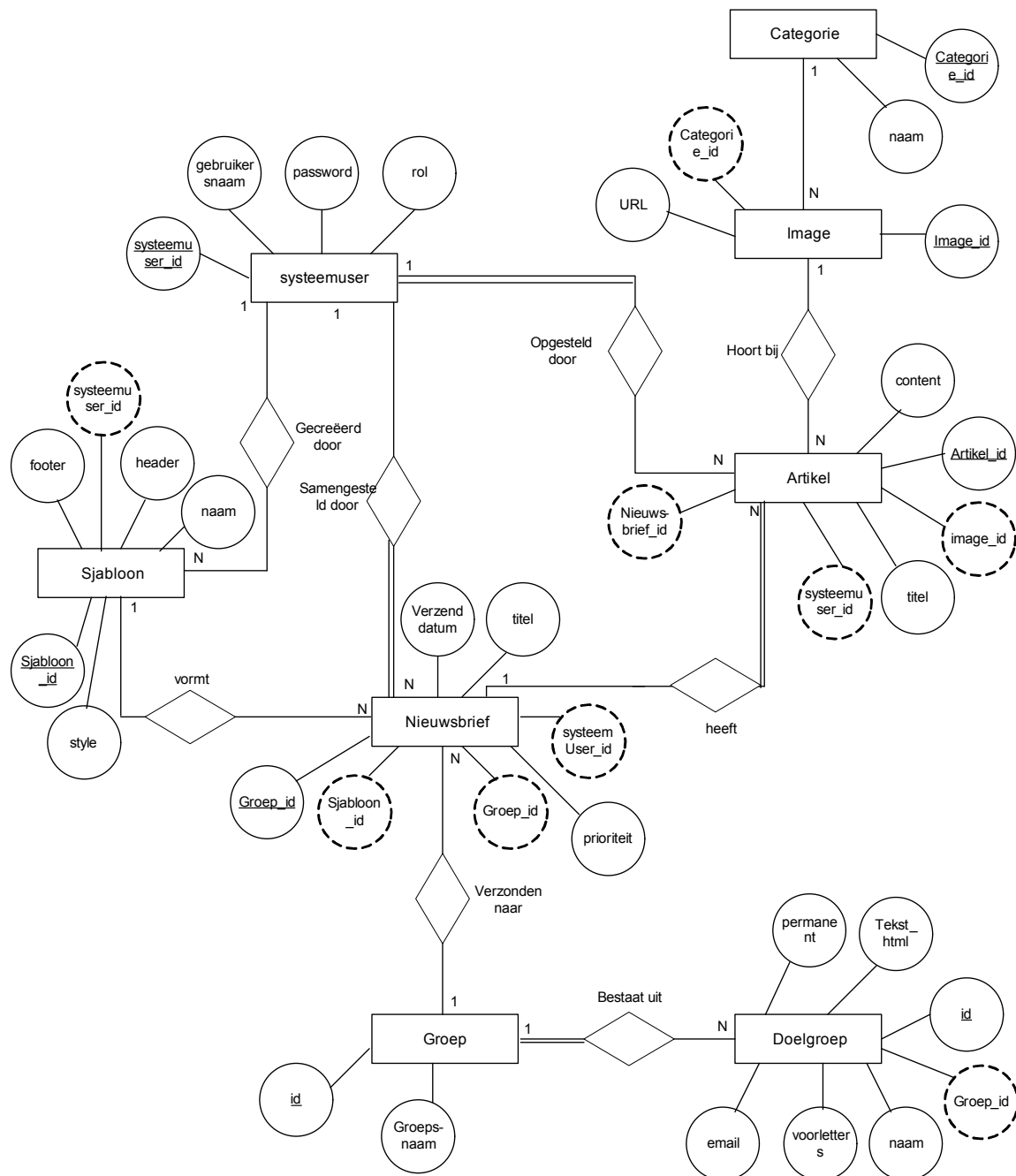
Pilotdelen zijn stukjes software die samen het systeem vormen. In deze pilot zijn niet alle pilotdelen op code gebaseerde software eenheden. De pilot database bestaat nauwelijks uit ASP code, maar is wel de belangrijke basis van het uiteindelijke systeem.

4.1.1 Ontwerp -en bouwfase database

In de ontwerpfase heb ik, voor elke pilot apart, een pilotontwikkelplan ofwel ontwerpproject geschreven. Hierin kwam de tekstuele uitwerking en de gedetailleerde planning van deze fase. Het eerste pilotdeel van de pilot database is het EER-model. De modelleringstechniek geeft de relaties weer tussen de verschillende objecten en in hoeverre deze afhankelijk van elkaar zijn. Dit houdt in dat van te voren precies wordt uitgedacht hoe de structuur van de database eruit gaat zien. Uitgangssituatie is het klassendiagram zoals deze opgenomen is in de definitiestudie. Het EER-model was vervolgens de input van het volgende pilotdeel. Op de volgende pagina heb ik het EER-model afgebeeld.

Ten opzichte van het klassendiagram zijn er enkele wijzigingen doorgevoerd. Zo zijn enkele namen van tabellen gewijzigd doordat er interpretatieverschillen waren tussen de opdrachtgever en de ontwikkelaar. Een voorbeeld van zo'n verschil was de benaming van gebruiker en systeemuser. Het onderscheid tussen ontvangers en gebruikers ging niet altijd goed. Deze verschillen heb ik door deze wijziging rechtgetrokken. Door het globale karakter van het klassendiagram zijn er in deze fase nieuwe attributen aan entiteiten gegeven. Ook is er een kolom 'image' bijgekomen, hier zal ik in paragraaf 4.2 nog op terugkomen.

Tijdens de realisatie van de verschillende pilots kwamen er ook enkele wijzigingen van de database naar boven. Zo is bijvoorbeeld het sjabloon uitgebreid met een kolom 'temp_style'. De kolom 'style' was niet doeltreffend genoeg, en was alleen dekkend voor het tekstuele gedeelte van het sjabloon. Met de uitbreiding was het wel mogelijk om bijvoorbeeld achtergrondkleuren en randen van de nieuwsbrief te specificeren.



Figuur 4.1: EER-model

Om tot een fysieke database te komen is dit EER-model niet voldoende. Het model mist nog enkele eigenschappen, zoals constraints en de juiste normaalvorm. Om deze toch te achterhalen en vast te leggen heb ik ervoor gekozen een representatiemodel en een implementatiemodel op te stellen.

Representatiemodel

Het representatiemodel geeft de tabellen weer zoals deze uiteindelijk in de database geïmplementeerd zullen worden. Deze zijn tot de vierde normaalvorm genormaliseerd. Deze

normalisatie betekent dat de relaties functioneel afhankelijk van elkaar zijn, geen overtolligheden en tegenspraak bevatten en niet meerwaardig afhankelijk zijn. Deze verwerking voorkomt problemen in de toekomst, door mogelijk inconsistente gegevens die de integriteit van de database aantasten.

Artikel(artikel_id, titel, content, image_id, systeemuser_id, nieuwsbrief_id)

Image_id references image null toegestaan

Systeemuser_id references systeemuser null niet toegestaan

Nieuwsbrief_id references nieuwsbrief null niet toegestaan

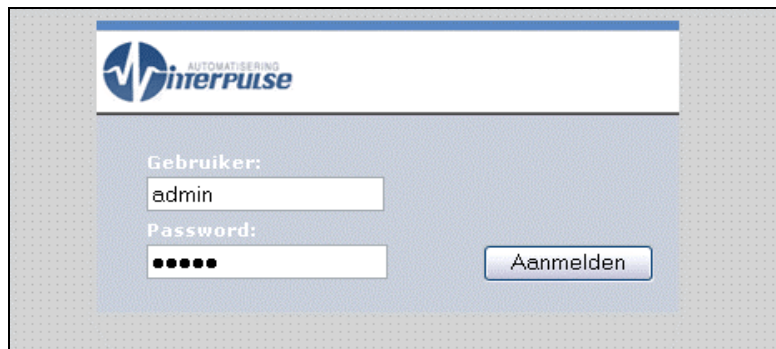
implementatiemodel

De volgende stap in de pilot database is het opstellen van het implementatiemodel. Met dit model kan de fysieke database gecreëerd worden. Een klein gedeelte van het implementatiemodel ziet er als volgt uit:

```
Create table artikel (
    artikel_id      int          IDENTITY(1 , 1) CONSTRAINT PK_artikel_id PRIMARY KEY,
    titel           varchar(30),
    content         varchar(1500),
    image_id        int          CONSTRAINT FK_image_id REFERENCES image(image_id)
                                ON DELETE NO ACTION
                                ON UPDATE NO ACTION ,
    systeemuser_id  int          CONSTRAINT FK_systeemuser2_id REFERENCES
                                systeemuser(systeemuser_id)
                                ON DELETE NO ACTION
                                ON UPDATE NO ACTION      NOT NULL,
    nieuwsbrief_id  int          CONSTRAINT FK_nieuwsbrief_id REFERENCES
                                nieuwsbrief(nieuwsbrief_id)
                                ON DELETE NO ACTION
                                ON UPDATE NO ACTION      NOT NULL)
```

In de Enterprise manager van SQL server is het zeer eenvoudig om de bovenstaande code in te voeren en te controleren. Met behulp van het gehele implementatiemodel heb ik de database gecreëerd en vervolgens een databasemodel uit SQL server laten rollen. Het databasemodel geeft de exacte tabellen met de juiste benamingen. De volgende stap is het uitvoeren van enkele testquery's. Door te beginnen met enkele INSERT query's kan de database gevuld worden met testdata. Deze data kan vervolgens opgeroepen worden met diverse SELECT query's. Uit de testen bleek dat de database naar wens werkte. Ook de DELETE query's functioneerde volledig naar wens.

Met bovenstaande testjes was ik niet tevreden, omdat deze testen nog niet in relatie staan met de ASP pagina's. Daarom heb ik ervoor gekozen om een doelgerichtere test uit te voeren. Het eindresultaat hiervan was het login gedeelte.



Figuur 4.2: Login

4.1.2 Uitvoeren prototyping en test workshop

Al voor de bouw van de pilot database heb ik besloten deze pilot niet verder te testen met de gebruikers. Het laatste pilotdeel, login, was naar mijn mening al voldoende om de technische werking en betrouwbaarheid van het systeem te verifiëren. Ook de gebruikers heb ik niet benaderd na afronding van deze pilot. Zij zouden niet geïnteresseerd zijn in een login die nog nergens opuit komt en al helemaal niet om een databasemodel die voor hun onbegrijpelijk is.

4.1.3 Beschrijving iteratieslagen

Tijdens de realisatie van de verschillende pilotdelen had ik regelmatig contact met de opdrachtgever. Hierbij bespraken we de laatste versie van het systeem, die op een laptop als prototype draaide. Door deze werkwijze was de opdrachtgever op de hoogte van de laatste ontwikkelingen en kon ik bij vragen meteen reageren. Tijdens deze gesprekken kwamen er, in enkele gevallen, zaken naar boven die een wijziging in de database noodzakelijk maakte. Een voorbeeld is een wijziging die naar boven kwam tijdens de bouw van de nieuwsbrief samenstel module. De opdrachtgever wilde namelijk dat het systeem de mogelijkheid moest bieden om de nieuwsbrief naar meerdere groepen te verzenden en dat sommige artikelen toch bij meerdere nieuwsbrieven konden horen. Uit de eerste gesprekken, tijdens de oriëntatiefase bleek dit nog anders te zijn.

Door deze, toch wel flinke, wijziging heb ik met de opdrachtgever besproken hoe we deze stap in zouden plannen. Uit dit gesprek bleek dat de wijziging toch wel hoog op de verlanglijst stond van de opdrachtgever. Daarom hebben we in overleg, en tegen de 'scope creepe' afspraak in, besloten deze wijziging meteen door te voeren in de database, welke resulteerde in de volgende twee extra tabellen:

- Groep_nieuwsbrief
- Artikel_nieuwsbrief

Om de wijziging ook voor de toekomst vast te leggen heb ik de resultaten van de pilotontwikkelingsfase aangepast. Zo zijn het representatie, implementatie -en EER-model geëvolueerd naar een nieuwe versie. Deze wijziging heeft mij totaal een extra dag gekost, maar conflicteerde verder niet met de planning, doordat de wijziging op tijd naar boven kwam.

4.2 Pilotontwikkeling pilot artikelsysteem

Het artikel ontwerp systeem draagt zorg voor de content van het systeem. Hier kunnen de beheerder(s) en de gebruikers artikelen schrijven, zodat deze in de uiteindelijke nieuwsbrief gepubliceerd kunnen worden.

4.2.1 Ontwerp -en bouwfase artikelsysteem

De ontwerpfase van deze tweede pilot is anders dan voorgaande pilot. Deze pilot bestaat in veel grotere mate uit ASP code en vormgeving. De hierop volgende pilot zal in grote mate overeenkomen met het artikelsysteem. Deze eigenschappen zal ik dan ook alleen in deze pilot bespreken.

Basis:

- WYSIWYG - editor
- Artikelopslag

Comfort:

- Afbeeldingen catalogus
- Gebruikersondersteuning

Het zwaartepunt, op technisch gebied, van deze pilot was de WYSIWYG editor die 'lastig' te ontwikkelen is. Doordat ik in de planning 3 dagen gereserveerd had voor de ontwikkeling van dit systeem moest ik hard doorwerken. Uiteindelijk heb ik over dit pilotdeel 4 dagen gedaan. Hierdoor heb ik alle pilotdelen één dag opgeschoven. Maar, de ontwikkeling van de artikelopslag bleek vervolgens voorspoediger te gaan dan verwacht, waardoor toch alle pilotdelen binnen de gehele timebox ontwikkeld konden worden.

Taakbalk

Binnen het systeem heb ik een taakbalk concept ontwikkeld die bruikbaar blijkt voor het systeem. Met behulp van de opdrachtgever, die deze balk getest heeft, heb ik een versie ontwikkeld die zeer flexibel in gebruik is en makkelijk herbruikbaar en uitbreidbaar is. Uit de eerder uitgevoerde onderzoeken bleek dat de persona's een redelijk kennisniveau hebben van MS Word. Daarom heb ik voor dit systeem gekozen voor de Word methode met betrekking tot de taakbalk en de WYSIWYG editor.

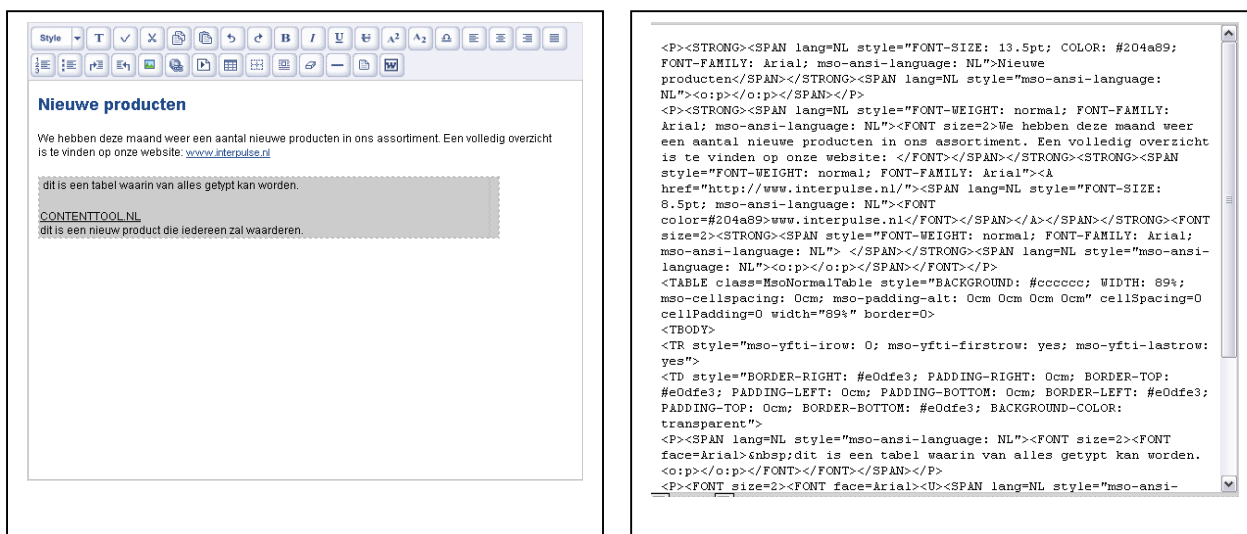


Figuur 4.3: Taakbalk

Aan bovenstaande taakbalk kunnen eenvoudig nieuwe items toegevoegd of verwijderd worden. Daarnaast kan de balk per scherm anders ingevuld worden, maar is de basis steeds herbruikbaar.

WYSIWYG-editor

De what you see is what you get editor is een werkveld waarmee gebruikers van het systeem handelingen uit kunnen voeren op teksten. Zo kan de gebruiker teksten vet maken of bijvoorbeeld een geordende lijst maken. De WYSIWYG-editor maakt op de achtergrond de HTML code aan die opgeslagen wordt in de database. De gebruiker ziet alleen het resultaat en niet deze onleesbare code. Hieronder heb ik het verschil tussen HTML en dezelfde code in de WYSIWYG editor weergegeven.

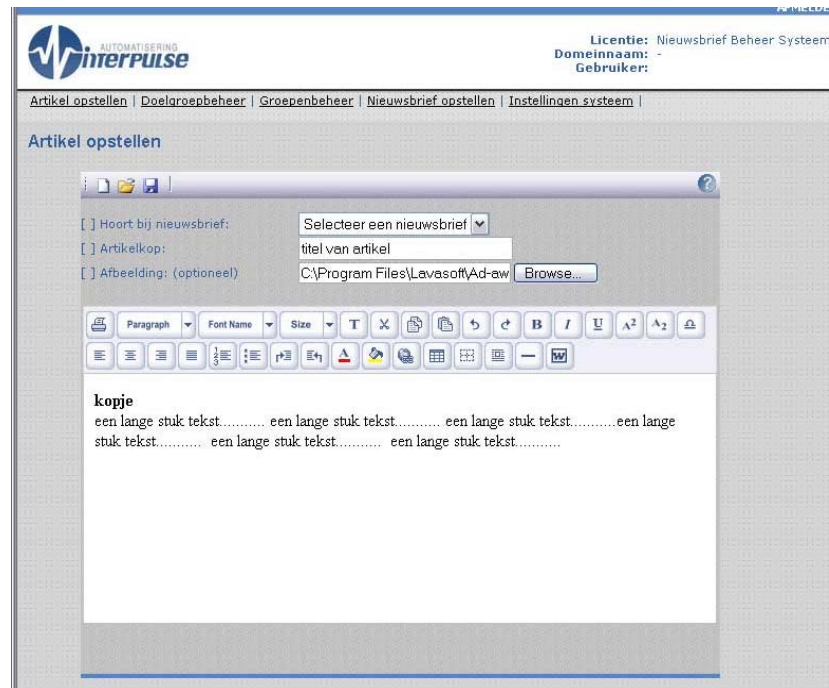


Figuur 4.4: WYSIWYG

Een nadeel van deze WYSIWYG-editor is de afhankelijkheid van Internet Explorer 5 of hoger. Met de implementatie van de WYSIWYG-editor zou ik niet kunnen voldoen aan de systeemeis 'platformafhankelijk'. Om hier een oplossing voor te bieden heb ik een functie geschreven die bij binnenkomst van de pagina eerst controleert welke browser en browserversie de bezoeker heeft. Wanneer de browser niet voldoet aan de eisen, zal de pagina zonder WYSIWYG-editor getoond worden. Maar, om toch teksten te kunnen beheren zal er in plaats van de editor een 'normaal' tekstveld getoond worden die elke browser ondersteund.

Prototype

Om de opdrachtgever snel tevreden te stellen en hem het gevoel te geven dat er bergen verzet worden, wat natuurlijk ook het geval was, heb ik na de ontwikkeling van het eerste pilotdeel een prototype ontworpen. In dit prototype heb ik de taakbalk geïntegreerd en de WYSIWYG editor. De opdrachtgever was erg content met deze eerste weergave en werd meteen enthousiaster. Hier kwam een voordeel van de 'Juicy bits first' techniek naar boven.



Figuur 4.5: Prototype

Afbeeldingen catalogus

Uit verschillende interviews met de opdrachtgever bleken nieuwsbrieven regelmatig (90%) voorzien te worden van afbeeldingen. In eerste instantie een 1 op 1 relatie, maar dit idee wijzigde zich in de fase pilotontwikkeling database. Het moest een één op veel relatie worden. Afbeeldingen moeten gebruikers zelf uit kunnen kiezen en aan het systeem toe kunnen voegen. Anderzijds vind de beheerder het in enkele situaties handig om zelf afbeeldingen aan het systeem toe te voegen waaruit vervolgens de schrijver van het artikel kan kiezen. Een voorbeeld die de opdrachtgever schetste was een groot aantal foto's van producten uit het assortiment van zijn bedrijf. De beheerder maakt deze foto's en plaatst ze in het systeem, vervolgens kunnen schrijvers de foto uitzoeken die bij het best bij het artikel past. Daarnaast zouden er afbeeldingen kunnen zijn die iedere nieuwsbrief terug kunnen komen.

Met bovenstaande informatie heb ik een idee geschetst die de opdrachtgever graag uitgewerkt wilde zien. Het betrof een afbeeldingencatalogus met diverse categorieën met daarbij de mogelijkheid om bestanden up te loaden en in een, eventueel zelf aangemaakte,

categorie te plaatsen. Met de presentatie van de eerste uitgewerkte versie was de opdrachtgever tevreden, waardoor de catalogus afgemaakt kon worden.

Gebruikersondersteuning

De gebruiker moet te allen tijde hulp kunnen krijgen. En de help moet ook over het onderwerp gaan waar de gebruiker zich op dat moment begeeft. Zo heeft iedere pagina een eigen help knop (?) in de taakbalk. Ik heb er voor gekozen deze pagina in een pop-up te realiseren, zodat de gebruiker meteen het scherm kan zien waar de vraag betrekking tot heeft en er geen werk verloren gaat.



Figuur 4.6: Help file

Ik heb gekozen voor korte, concrete beschrijvingen waarmee de gebruiker stapsgewijs kan zien wat hem te doen staat. Vragen dienen snel beantwoord te worden, zonder omhaal van woorden. William Horton schrijft in zijn boek over online documentation. Hij geeft aan op welke manier het beste hulp geboden kan worden aan de systeemgebruikers. Enkele kernregels van W. Horton, welke ik onder andere toegepast heb, zijn:

- Gebruik korte, eenvoudige en bekende woorden;
- Gebruik afbeeldingen om zaken duidelijk te maken, zoals pictogrammen;
- Laat de help file niet het werk van de gebruiker verstoren;
- Beantwoord één vraag per keer en beschrijf deze kort.

Ook de structuur van de helpfile heb ik zo intuïtief mogelijk gehouden. Zoals gezegd komt de gebruiker de helpfile binnen bij het onderdeel waar de gebruiker op help klikt. Daarnaast kan de gebruiker een overzicht van alle helponderwerpen raadplegen.

4.2.2 Uitvoeren prototyping en test workshop

Na afronding van de pilot heb ik een bijeenkomst georganiseerd met de gebruikers en de opdrachtgever. Met behulp van een scenario moesten de gebruikers een artikel aanmaken en hier een afbeelding aan toevoegen. Deze werkwijze is gebaseerd op de black box testtechniek. Hierbij ligt de nadruk bij de in -en output en niet zo zeer op wat het systeem, achter de schermen, uitvoert. Wat het systeem uitvoert is niet interessant voor de gebruikers, daarom is de blackbox testtechniek geschikt voor de gebruikerstesten.

De gebruiker krijgt het doel of scenario te horen van de ontwikkelaar en gaat vervolgens aan de slag. In onderstaande tabel staat het ideale scenario. Bij de observatie wordt gekeken naar de handelingen van de gebruiker maar ook naar de verbale en non-verbale reactie. De gebruiker wordt dan ook gevraagd om hard op te denken (think aloud techniek).

Scenario: "tekst kopiëren uit externe bron"			
(Toon de post en de query op scherm + lees database uit)			
Taak	Opmerkingen	Succes	Fout
Open een webpagina of word document met tekst	<i>Gebruiker opent een website naar keuze: nu.nl</i>	V	
Selecteer een stuk tekst naar keuze	<i>Hij opent een nieuwsbericht en selecteert een stuk tekst</i>	V	
Kopieer de tekst	<i>goed</i>	V	
Login het systeem	<i>Verkeerd wachtwoord, maar gaat daarna goed.</i>	V	
Ga naar nieuw artikel toevoegen	<i>Komt binnen bij artikeloverzicht, maar moet even zoeken naar de nieuw knop.</i>		X
Vul de velden in en plak tekst in tekstveld	<i>Gaat goed, laat geen velden leeg</i>	V	
Klik op opslaan	<i>"Nu weet ik waar die knop zit" zegt de gebruiker lachend. Wijst op de taakbalk.</i>	V	
Bekijk resultaat in overzicht	<i>Taak afgerond en door naar volgende taak.</i>	V	

Uit deze test kwam een eerste hindernis naar boven voor de gebruikers. De taakbalk zoals deze gerealiseerd was bleek moeilijk te begrijpen voor enkele gebruikers. Vooral de eerste keer was dit een twijfelmoment in het stappenplan zoals die gevolgd werd. Daarom heb ik ervoor gekozen hier een oplossing voor te vinden. Hier zal ik verder op in gaan in de volgende paragraaf.

Naast de gebruikerstest heb ik zelf tijdens de bouw diverse testen uitgevoerd. Deze methode was gebaseerd op de white box testtechniek. Bij deze techniek ligt de nadruk bij de systeemwerking, wat interessant is voor mij als ontwikkelaar. Wat voert het systeem van binnen, voor de gebruiker niet zichtbaar, uit. Hiervoor heb ik een aantal mogelijke situaties opgesteld en uitgewerkt zoals bij de black box testtechniek. Te denken valt aan onverwachte of corrupte invoer. De aanpak was eenvoudig uit te voeren. Door het vooraf opgestelde scenario uit te voeren en de database query op het scherm te schrijven kwamen al een aantal zaken naar boven die verbeterd konden worden. Deze zaken waren zo kleinschalig dat ik meteen bij het testen de wijziging realiseerde. Een voorbeeld van zo'n fout was de datumnotatie zoals deze weggeschreven werd naar de database. Vanwege consistentie moet dit in het Nederlands zijn, maar was in Engelse notering.

4.2.3 Beschrijving iteratieslagen

Zoals reeds beschreven in paragraaf 4.2.2 bleek de taakbalk voor enkele gebruikers minder logisch dan verwacht. Hierdoor heb ik een extra pictogrammen box gerealiseerd. De gebruikers bleken deze items wel te kunnen vinden waardoor er in het systeem een alternatief werd geboden voor de taakbalk.



Figuur 4.7: Pictogrammenbox

Daarnaast heb ik rechts onderin het scherm een extra opslaan knop geplaatst. Gebruikers die al redelijke internet ervaring hebben bewogen met de muis automatisch naar die plaats, rechts onderin, toe. Vervolgens heb ik na deze iteratie opnieuw een testmoment geprikt met een gedeelte van de gebruikers. Uit deze testen kwam een positief resultaat naar boven, waardoor de iteratie, binnen de timebox, met succes is uitgevoerd.

4.3 Pilotontwikkeling pilot doelgroepbeheer en importeursysteem

De pilot doelgroepbeheer en het importeursysteem maken het de beheerder eenvoudig om geïnteresseerde aan te maken, gegevens te wijzigen of te verwijderen. Daarnaast zal dit systeem de mogelijkheid moeten bieden om gegevens van geïnteresseerden te importeren met behulp van een Excel importeerfunctie.

4.3.1 Ontwerp -en bouwfase doelgroepbeheer en importeursysteem

Basis:

- Doelgroepbeheer
- Groepenbeheer
- Importeermodule
- In / uitschrijven module gebruikers

Comfort:

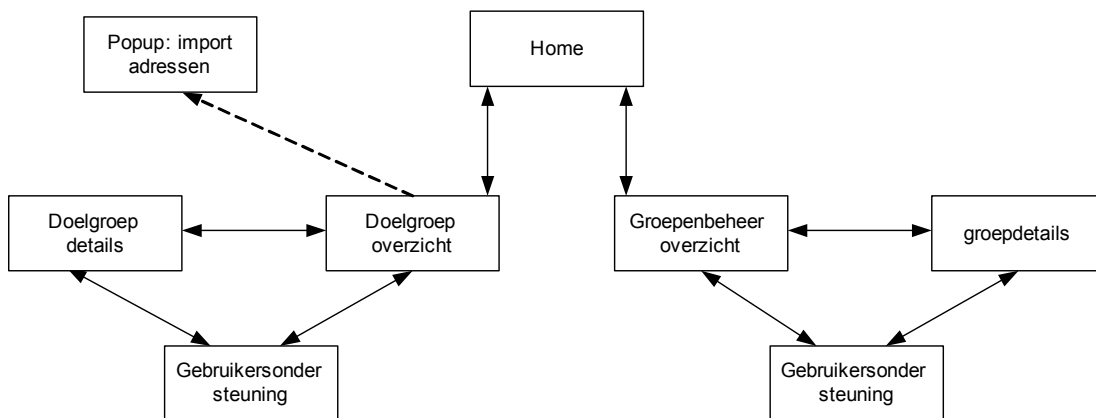
- Gebruikersondersteuning

Luxe:

- Dubbele emailadres handlingmodule

Gespecificeerd navigatieschema

Om een gedetailleerder beeld van de pilots te krijgen heb ik er voor gekozen een meer gespecificeerd navigatieschema op te stellen tijdens deze fase. Dit schema, zoals hieronder weergegeven, geeft mij als ontwikkelaar en de opdrachtgever een helder beeld van de pilot.



Figuur 4.8: Gespecificeerd navigatieschema

Taakscenario toekomstige situatie

Vervolgens heb ik er voor gekozen een aantal korte scenario's op te stellen van de verschillende gebruikersgroepen. Deze scenario's geven een weergave van de toekomstige situatie. Voor de gebruikers geldt dat zij alleen toegang hebben tot het artikel ontwerp systeem. Voor de beheerder geldt dat het gehele systeem bereikbaar is.

Importeren gebruikers

Reeds in een vroeg stadium kwam naar boven dat er behoefte was aan een importeermodule die grote hoeveelheden gebruikers kan importeren. Voordeel hiervan is dat niet alle doelgroepleden (de personen die de nieuwsbrief zullen ontvangen) handmatig ingevoerd hoeven te worden. Op deze manier kunnen klantgegevens snel toegevoegd worden aan het systeem. De input van dit importeesysteem is een excel bestand. De rijen in het bestand worden uitgelezen en in de database geïmporteerd.

Gedetailleerde planning

Een belangrijk onderdeel van de pilotonwikkelplannen was het plannen van de activiteiten die uitgevoerd moesten worden. Hiervoor heb ik per pilot een gedetailleerde planning opgesteld. In de planning gaf ik een weergave van de geplande dagen ten opzichte van de activiteiten. Hieronder een voorbeeld van een detailplanning van deze pilot.

Werkdag in timebox	1	2	3	4	5	6	7	8	9	10
Activiteit										
pilotontwikkelplan	S	C	D							
Doelgroep beheer				S/C	D					
Groepenbeheer						S/C	D			
Importeermodule								S		
In en uitschrijfmodule										
Gebruikersondersteuning										
Del dubbele mailadres										
S = start activiteit C = concept D = definitieve versie										

Werkdag	11	12	13	14	15	16	17	18		
Activiteit										
pilotontwikkelplan										
Groepenbeheer										
Importeermodule	C	D								
In en uitschrijfmodule			S/C/D							
Gebruikersondersteuning				S	C	D				
Del dubbele mailadres							S/C	D		
S = start activiteit C = concept D = definitieve versie										

Binnen de fase pilotontwikkeling heb ik gebruik gemaakt van timeboxing. Hierdoor ben ik in staat het project te beheersen en kan ik er zorg voor dragen dat de basis functionaliteiten met zekerheid gerealiseerd zullen worden. Daarnaast staat de oplevering van de pilot reeds van te voren vast. Binnen het project zullen we de gehele pilots opdelen in timeboxes van variërende tijdlengtes. Het resultaat van deze indeling is, dat in ieder geval de meest belangrijke onderdelen gerealiseerd worden en mogelijk, afhankelijk van eventuele vertragingen, de comfort en luxe eisen.

4.3.2 Uitvoeren prototyping en test workshop

Ook voor deze pilot heb ik een aantal scenario's opgesteld die de gebruiker doorloopt met behulp van de think aloud en observatietechniek. Het zou dubbel op zijn om hier nogmaals op in te gaan, daarom laat ik dit in deze en volgende pilot achterwege.

4.4 Pilotontwikkeling pilot nieuwsbrief samenstel module

De pilot nieuwsbrief samenstel module draagt zorg voor de samenstelling en de uiteindelijke verzending van de nieuwsbrief. De beheerder kan met behulp van deze pilot de nieuwsbrief bekijken, controleren en definitief verzenden naar geselecteerde groepen.

4.4.1 Ontwerp -en bouwfase doelgroepbeheer en importeursysteem

De laatste pilot bestaat uit de volgende onderdelen:

Basis:

- Artikelen samenstellen tot nieuwsbrief
- Selectie geadresseerden
- Selectie template
- Verzendmethode naar SMTP

Comfort:

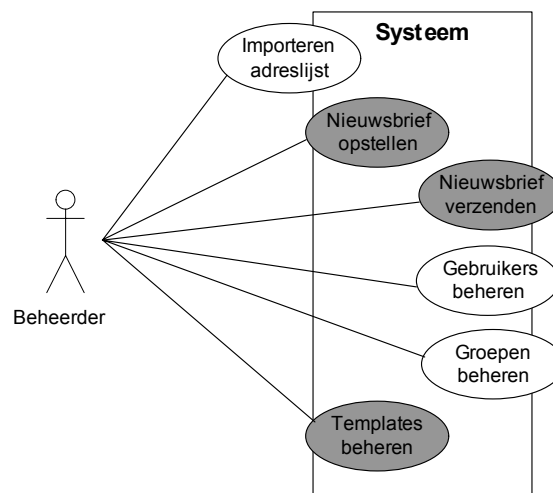
- Email personalisering
- Nieuwsbrief geschiedenis
- Gebruikersondersteuning

Luxe:

- Template ontwikkel wizard

Gespecificeerde use case

Om voor de opdrachtgever de zaken concreter te maken heb ik de use case uit de definitiestudie weer naar boven gehaald. Door de onderdelen die betrekking op de pilot hadden grijs te markeren kreeg de opdrachtgever een duidelijker beeld van het te ontwikkelen gedeelte van het systeem.



Figuur 4.9: Gespecificeerde use case

SMTP mailserver

Een niet onbelangrijk gedeelte van het systeem is de verzending naar de mailserver. Bij deze verzending moet rekening gehouden worden met een aantal zaken, zoals de 'soort' van

verwerking van de nieuwsbrieven. Zo kan bij verzending van grote hoeveelheden nieuwsbrieven (> 250) de mailserver problemen krijgen, doordat deze verstopt raakt. Om dit probleem op te lossen heb ik er voor gekozen om de mailtjes eerst naar de 'queue' (wachtrij) te sturen. Hierdoor komen alle berichten netjes aan de beurt. Bijkomend voordeel is dat de nieuwsbrieven nooit verloren gaan. Wanneer de mailserver problemen heeft en deze offline moet zullen de berichten onthouden worden in het geheugen van de server. Zodra de server weer online is zullen de nieuwsbrieven alsnog verzonden worden.

Een andere belangrijke eis was de mogelijkheid om nieuwsbrieven offline leesbaar te laten zijn. Met de teksten zal dit geen probleem zijn, maar afbeeldingen worden bij veel nieuwsbrieven gedownload van een bepaald internet adres. Door het gebruik van een functie in ASP is het mogelijk gemaakt om aan een email 'embedded' plaatjes toe te voegen. De afbeeldingen worden dus met de email meegestuurd. Niet als bijlage, maar geïntegreerd. Hieronder zal ik een voorbeeld schetsen.

Externe URL

Wanneer een ontvanger een nieuwsbrief download met behulp van bijvoorbeeld Outlook express en hierna de verbinding met het internet sluit zullen de afbeeldingen niet meer te zien zijn.

```
<html>

...
<img src= "http://www.interpulse.nl/
mailing.jpg">
...
</html>
```



Embedded image

Het gebruik van de embedded image functie zullen de afbeeldingen zelfs zonder internet verbinding zichtbaar zijn.

```
<%
Mail.AddEmbeddedImage "images/mailing.jpg"
, "image1"
Mail.body = "<html>....<img
src='\"cid:image1\"'> ...</html>"
%>
```



4.4.2 Uitvoeren prototyping en test workshop

In deze paragraaf zal ik een aantal activiteiten beschrijven die uitgevoerd zijn bij het testen van deze pilot. Ook voor deze activiteiten geldt dat ze als voorbeeld dienen en eenmaal opgenomen worden in het verslag.

Controle usability lijst

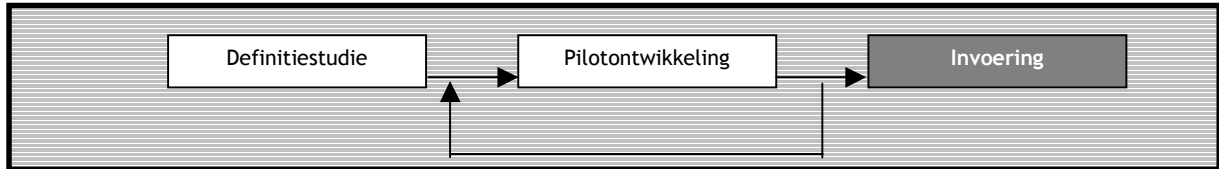
Tijdens de oriëntatiefase van het project heb ik reeds een usability checklist opgesteld met behulp van de literatuur van Jacob Nielsen. Om te controleren in hoeverre de pilot voldoet, heb ik de lijsten ingevuld. Om een indruk te geven heb ik hieronder een gedeelte van de checklist weergegeven, deze lijst heeft betrekking tot de content. De V staat voor in orde, de X voor niet gelukt.

Richtlijn	V	X
Homepage should answer "Where am I" and "What does this site do".	V	
Is text short and concise?	V	
Use short paragraphs, headings, subheadings, and bullets for ease of scanning.	V	
Paragraphs should ideally start with a topic sentence and keep to one idea per paragraph.	V	
Can user tell in a glance what the page is about and what it can do for them?	V	
Help should contain examples, be task oriented, show how to do step by step, but be brief.	V	

Wanneer een onderdeel uit de lijst niet voldoet zal gekeken worden naar een oplossing. Wanneer deze oplossing te veel tijd zou kosten om te realiseren kan, in het uiterste geval, overwogen worden om de richtlijn te negeren. Hier zal dus een afweging gemaakt moeten worden tussen de gebruiker en het technische aspect.

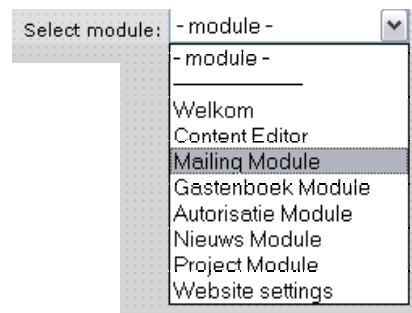
5. Invoering

De invoeringsfase heeft als doel het complete systeem operationeel te maken in de gebruikersomgeving. Het is niet mogelijk en gewenst om het nieuwsbrief beheersysteem in stappen in te voeren, daarom zal de fase invoering slechts een keer doorlopen worden.

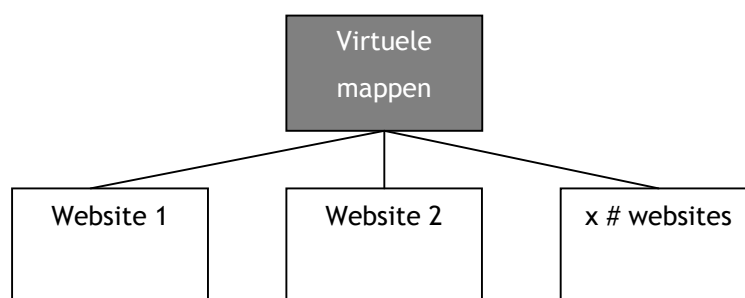


5.1 Integratie systeem in Interpulse CMS

Interpulse biedt haar klanten al enige tijd een content management systeem aan waarmee websites eenvoudig beheerd kunnen worden. Het nieuwsbrief beheersysteem moest als optionele module aan het systeem gekoppeld worden. In de praktijk betekent dit een koppeling tussen de navigatie van het CMS en van het nieuwsbrief beheersysteem. Hiervoor is gebruik gemaakt van een 'module selector' in de vorm van een pulldown menu. Deze module selector wordt in de header van het systeem geplaatst.



Om in de toekomst fouten snel op te kunnen lossen heb ik in overleg met Interpulse besloten het systeem in de virtuele mappen van de webserver te plaatsen. Deze mappen kunnen aan websites gekoppeld worden, waardoor deze websites van dezelfde bestanden gebruik maken. Schematisch ziet dit er als volgt uit:



Figuur 5.1: Virtuele mappen

Voordeel van dit overkoepelend gebruik is de eenvoudige mogelijkheid tot uitbreidbaarheid en / of oplossen van centrale problemen. Hierdoor is het niet nodig bij iedere website apart de wijziging door te voeren, maar dit hoeft slechts één keer. Nadeel van de virtuele mappen is de impact bij mogelijke fouten. Wanneer er iets fout is in de centrale bestanden, zal dit bij alle nieuwsbrief beheer systemen merkbaar zijn. Om dit te voorkomen zal er altijd intern

ontwikkeld worden, vervolgens intern getest en als laatste zal het systeem getest worden door een collega voordat er een wijziging definitief doorgevoerd wordt.

5.2 Operationaliseren stand-alone beheersysteem

Het systeem zal naast de koppeling met het Interpulse CMS ook als losstaand product worden aangeboden. Met de aanschaf van het product krijgt de klant een persoonlijke versie die volledig tot de beschikking staat.

Ook bij het operationeel maken van deze stand-alone versie geldt het gebruik van de virtuele mappen. De klant is namelijk verplicht het systeem te hosten op een Interpulse server. Hierdoor blijft de ASP code in eigen huis en daarmee wordt voorkomen dat derden de code misbruiken of 'hergebruiken'.

5.3 Acceptatietest van het systeem

In deze fase van het project is het gehele systeem operationeel. Daarom is het zaak om de laatste testen uit te voeren en vervolgens het project formeel af te ronden. Ik heb er voor gekozen deze laatste test met een externe klant van Interpulse uit te voeren. De klant krijgt het systeem zoals elke klant dit in de toekomst zal krijgen. De testpersoon kan vanaf dat moment zelfstandig aan de slag gaan. Doel van de workshop is het verzenden van een nieuwsbrief naar het emailadres van de testpersoon.

Doordat er al tijdens het project al vele testen uitgevoerd waren, met behulp van prototypes, waren de resultaten van deze laatste test positief. Noemenswaardige problemen kwamen niet meer naar boven. Het systeem bleek intuïtief te zijn, de gebruiker kon zelfstandig zijn weg vinden door het systeem. Na deze laatste testen heb ik in overleg met de opdrachtgever afgesproken dat er direct email contact is tussen de klant en de ontwikkelaar. Bij eventuele problemen of vragen kan er direct gereageerd worden. Door dit contact krijgt de klant optimale ondersteuning. Na deze fase is het systeem gereed om 'op de markt' gebracht te worden.

Bij een mogelijke uitbreiding van het systeem is er in deze fase geïnventariseerd welke onderdelen van het systeem nog gerealiseerd kunnen worden bij een mogelijk volgende iteratie. Dit geeft de ontwikkelaar op dat moment een helder beeld van waar het systeem nu staat en wat de plannen nog zijn.

Systeemonderdeel	Huidige prioritering	Status
Dubbele emailadres handlingmodule	Luxe	-
Template ontwikkel wizard	Luxe	-
Gebruikersondersteuning	Comfort	75%
Responsemeter	Luxe	

6. Evaluatie

Dit laatste hoofdstuk heeft als doel het gehele proces en de producten te evalueren, hiermee krijgen de examinatoren inzicht in wat ikzelf van de gerealiseerde producten vind. Als eerste zal ik het proces evalueren, vervolgens in paragraaf 2 de producten.

6.1 Procesevaluatie

Wanneer ik terugkijk op de afgelopen 18 weken mag ik best tevreden zijn. Niet alles verliep vlekkeloos, maar het uiteindelijke resultaat staat als een huis.

6.1.1 *Communicatie opdrachtgever en gebruikers*

Tijdens het project heb ik veelvuldig contact gehad met de opdrachtgever en de gebruikers. Dit veelvuldige contact heb ik als zeer prettig ervaren. De opdrachtgever nam veel tijd voor de bijeenkomsten waardoor alle zaken rustig besproken konden worden. Tijdens de bijeenkomsten hebben we flink kunnen brainstormen over het systeem en kwamen we al in een vroeg stadium op een systeemschets uit.

Het contact met de gebruikers was iets minder frequent, maar desalniettemin erg prettig. Tijdens de bijeenkomsten namen de gebruikers uitgebreid de tijd en gaven veel nuttige tips. Ook het enthousiasme was groot bij het uitvoeren van de workshops. Dit gaf de gebruikers vertrouwen in het systeem en de gehanteerde manier van werken hebben ze als erg prettig ervaren.

6.1.2 *Ontwikkelmethode*

Tijdens het project heb ik met succes gebruik gemaakt van de methode IAD. De methode biedt een duidelijke handvat en geeft daarbij een duidelijke lijn in het proces. Bij aanvang van het project was het lastig inschatten welke activiteiten bruikbaar zijn en welke onderdelen mogelijk niet van toepassing zijn. Ook het samenstellen van de technieken was een lastige klus. Uiteindelijk ben ik tevreden over de keuze van de activiteiten binnen de methode IAD. Ik zal hier later nog per fase op ingaan.

6.1.3 *Incrementeel ontwikkelen*

Over de keuze van de ontwikkelstrategie ben ik erg tevreden. Door eerst de systeemeisen volledig in kaart te brengen en deze af te beelden in een systeemconcept kreeg de opdrachtgever en de ontwikkelaar al snel een beeld van het systeem. Hierop kon snel gereageerd worden met feedback. Dit vond ik een voordeel van het eenmaal doorlopen van de definitiestudie. Enkele voordelen van de strategie binnen het project waren:

- Het systeem kon al snel goed worden gespecificeerd;
- De omvang van het systeem was te overzien.

Het in een keer invoeren van het systeem was een must voor dit project. De opdrachtgever zou aan losse onderdelen niets hebben, daarom was de keuze voor deze strategie van invoeren ('big bang') snel gemaakt. De keuze is mij achteraf ook zeer goed bevallen. Ik was nu in staat het gehele systeem op de lokale server te ontwikkelen en deze in één keer over te zetten op de webserver in Amsterdam. Deze manier van werken voorkwam mogelijke problemen zoals versieconflicten. Concluderend kan ik tevreden zijn over deze keuze van ontwikkelstrategie.

6.1.4 Fasen van het project

Tijdens het project zijn er een aantal fasen doorlopen. In deze paragraaf zal ik evalueren in hoeverre de verschillende fases geslaagd zijn geweest.

Oriëntatiefase

Doel van de oriëntatiefase was het voorbereiden van het project. Deze fase heeft zich voornamelijk vóór de projectperiode afgespeeld. Het opstellen van de opdrachtschrijving was de basis van deze fase en van het project. Na keuring van deze omschrijving was ik in staat om het plan van aanpak te realiseren. Het plan kon gezien worden als het contract waaraan ik me als ontwikkelaar moest houden, maar ook waar de opdrachtgever zich aan moest houden. Dit is uiteindelijk goed uitgekapt. De oriëntatiefase is achteraf goed bevallen, door het regelmatige contact met de betrokkenen was ik in staat om in deze fase reeds een eerste idee te schetsen van het uiteindelijke systeem. Hierdoor was ik in staat de activiteiten te plannen en de op te leveren producten vast te stellen. Een uitstekende basis om de volgende fase in te gaan.

Definitiestudiefase

Door de gehanteerde ontwikkelstrategie was het opstellen van de definitiestudie een tijdrovende klus met veel keuzes die bepalend waren voor het vervolg van het project. De betrokkenheid van de gebruikers en de opdrachtgever was in deze fase van groot belang. Daarom heb ik interviews gehouden, welke nuttige informatie opleverde. Daarbij heb ik een workshop gehouden over een belangrijk onderdeel van de definitiestudie, namelijk betreffende de systeemeisen. Wanneer ik terugkijk op de workshop was dit een erg nuttige bijeenkomst. De betrokkenen waren zeer enthousiast en kwamen met veel concrete en nuttige ideeën.

In de toekomst zal ik bij projecten zeker de workshopmomenten teruglaten komen, doordat deze veel informatie opleveren en daarbij het enthousiasme en de betrokkenheid van de deelnemers vergroot. Ook de verschillende onderzoeken zijn goed bevallen. Door eerst goed na te denken over bijvoorbeeld de gebruikers (persona's) en de te gebruiken scripttaal heb ik een bewuste stap gezet in het ontwikkeltraject. De onderzoeken waren daarnaast een belangrijke input voor het opstellen van de systeemeisen en het systeemconcept.

Pilotontwikkelingsfase

Het ontwerp, de bouw en het testen van de pilotdelen waren de fasen die binnen de pilotontwikkelingsfase doorlopen werden. Deze pilotontwikkelingsfase itereerde per pilot. Het opstellen van compacte, maar nuttige plannen heeft in de ontwerpfase per pilot bijgedragen aan een duidelijk gekaderd systeem. Door de systeemdelen duidelijk te plannen heb ik er voor kunnen zorgen dat de bouw voorspoedig verliep. De pilots werden afgesloten met testen. Tijdens deze testbijeenkomsten kwamen nuttige tips en een enkele foutmelding naar boven, welke snel hersteld konden worden. Het toepassen van de white -en blackbox methode heeft hier succesvol aan bijgedragen. Door de gebruikers hardop te laten denken was ik in staat om de gedachtegang van de gebruikers mee te beleven. Het faseren van de pilotontwikkelingsfase in ontwerp, bouw en test heb ik kortom als zeer prettig en nuttig ervaren.

Invoeringsfase

Dat het invoeren van het gehele systeem in één keer ('big-bang') zou plaats vinden stond voor mij en de gebruikers als een paal boven water. Een incompleet systeem was niet bruikbaar. Doordat de gebruikers al veelvuldig met het systeem in aanraking waren geweest konden zij, na invoering, zonder problemen met het systeem aan de slag. De keuze voor deze manier van invoeren was achteraf dan ook de verstandigste.

Het invoeren van het systeem was een relatief korte fase (1 week), welke succesvol volbracht is. Door het systeem te kopiëren naar de Amsterdamse webserver was een eerste stand-alone versie reeds operationeel. Het koppelen van het systeem aan het Interpulse CMS was ook in korte tijd te volbrengen. Ook hier ben ik geen noemenswaardige problemen tegengekomen. Laatste stap was het formeel accepteren van het systeem door de opdrachtgever en daarbij het project succesvol af te ronden. Achteraf mag ik terugkijken op een geslaagd project en daarbij was de opdrachtgever ook zeer tevreden. Ik mag dan ook met enige trots vermelden dat ik een baan heb aangeboden gekregen bij Interpulse.

6.1.5 Gehanteerde technieken

In deze paragraaf zal ik enkele technieken evalueren die ik tijdens de afstudeerperiode toegepast heb.

UML

Enkele onderdelen uit de uitgebreide set van UML heb ik tijdens het project gebruikt om een duidelijke weergave te geven van het systeem. Het klassendiagram heb ik reeds in de definitiestudie opgesteld. Achteraf was dit model te globaal waardoor er in het EER-model toch nog behoorlijk wat verschillen naar boven kwamen. Het gebruik van het klassendiagram heb ik overigens wél als positief ervaren. De opdrachtgever en ikzelf kregen al snel een gevisualiseerd beeld van het systeem. Toch zal ik het klassendiagram een volgende keer

proberen gedetailleerder op te stellen, waardoor mogelijk het EER-model overbodig zal worden.

Het gebruik van het use case diagram gaf een heldere weergave van de systeemgebruikers (actoren) ten opzichte van het systeem. De modellen waren veelzeggend, voor de opdrachtgever, maar ook voor mijzelf. Het gebruik van de modellen heb ik als nuttig ervaren.

GUIDE

Het gebruik van de technieken uit GUIDE waren voor mij redelijk vanzelfsprekend, doordat ze reeds in eerdere project hun nut bewezen hadden. Het analyseren van taken is met behulp van de taakmodellen van GUIDE goed gelukt. Daarbij hoorde ook de taakscenario's die kloppend zijn gemaakt met behulp van de gebruikers. Hierdoor was ik in staat me meer te verplaatsen in de gebruikers wat resulteerde in een op de gebruikers gebaseerd systeem.

Naast de taakanalytische modellen heb ik ook de style guide opgenomen zoals GUIDE deze beschrijft. De style guide gaf mij en mogelijk in de toekomst ook de collega's, inzicht in de gehanteerde stijlen. Hierdoor was ik in staat consistent om te gaan met kleuren, lettertypes en dergelijke. Ook in de toekomst zal ik het gebruik van een style guide toejuichen.

Alan Cooper

Van de technieken van Alan Cooper had ik, bij aanvang van het project, nog geen kennis. Voordat ik gebruik ging maken van deze technieken heb ik eerst onderzocht in hoeverre deze toepasbaar zijn binnen het project. Uiteindelijk bleek de persona techniek zeer bruikbaar. Ik kwam achter de technieken van Alan Cooper doordat ik op zoek was naar informatie over systeemontwikkeling met de nadruk bij de gebruiker. Alan Cooper heeft hier uitstekende technieken voor ontwikkeld. Tijdens het project heb ik met succes gebruik gemaakt van de persona techniek. Gedurende het gehele project heb ik bij keuzes de persona's fictief betrokken. In het verleden heb ik regelmatig keuzes gemaakt zonder bewust na te denken over de gevolgen bij een of meer type gebruikers. Persona's dwingen je na te denken over deze keuzes en dit heb ik als erg nuttig ervaren.

Jacob Nielsen

Jacob Nielsen is een goeroe wat betreft de 'usability' van systemen. Hij heeft in de loop der jaren vele boeken geschreven waarin hij uitvoerig inging op gebruikers en de manier van interactie met deze gebruikers. Tijdens het project heb ik onderzocht welke aspecten hij aanstipt en welke bruikbaar waren. Het onderzoek bleek erg nuttig en leverde een uitgebreide lijst van richtlijnen op. Tijdens de bouwfase heb ik continu gecontroleerd of het systeem nog aan de usability eisen kon voldoen. Deze manier van werken heb ik als erg prettig ervaren. Met behulp van de lijst moet je eerst nadenken wat te bouwen en daarna pas bouwen en dus niet andersom.

6.1.6 Planning

Over de werkwijze met betrekking tot de planning mag ik tevreden zijn. Door bij aanvang van het project een globale planning op te stellen kreeg ik al vroeg een totaal beeld. De planning was niet helemaal dekkend doordat ik in deze oriëntatiefase nog niet exact het aantal pilots wist.

Tijdens de definitiestudie kreeg ik al een gedetailleerder beeld van het project, waaruit ik een planning heb opgesteld op niveau van de pilots. Bij het opstellen van de pilotontwikkelplannen was ik in staat de basis, comfort en luxe pilotdelen in te plannen op zeer gedetailleerd niveau. Door deze werkwijze had ik een goed overzicht over het gehele project, maar zag ik ook in een oogopslag welke stap op welke dag gerealiseerd zou worden. Kortom een prettige manier van plannen. Overigens zijn uiteindelijk niet alle systeemeisen ontwikkeld, maar door de toepassing van de timebox en juicy bits first technieken was de opdrachtgever zeer tevreden met het eindresultaat. Alle basis en grotendeels alle comforteisen zijn binnen de planning gerealiseerd. Van de comforteisen is alleen de gebruikersondersteuning van de laatste pilot niet gerealiseerd.

6.2 Productevaluatie

In deze paragraaf zal ik de gemaakte producten evalueren welke tijdens het traject tot stand zijn gekomen. Het betreft de 'papieren' resultaten en het uiteindelijke nieuwsbrief beheersysteem.

6.2.1 Definitiestudie

Met de input van de interviews, plan van aanpak en workshop heb ik naar mijn mening een definitiestudie neer kunnen zetten die een goede weergave van het te bouwen systeem gaf en een duidelijke basis voor het vervolg van het project. In eerste instantie heb ik enkele onderzoeken uitgevoerd parallel aan het schrijven van de definitiestudie. Het opstellen van de systeemeisen was door de onderzoeken een tijdrovende klus, maar dit leverde vervolgens een, naar mijn mening, dekkend systeemconcept op.

6.2.2 Pilotontwikkelplannen

Door de uitgebreide beschrijving in de definitiestudie waren de ontwikkelplannen compact, maar naar mijn mening effectief. Het eerste ontwikkelplan, betreffende de pilot database, was wat betreft kwantiteit beperkt. Hierdoor heb ik de 'white noise' beperkt kunnen houden.

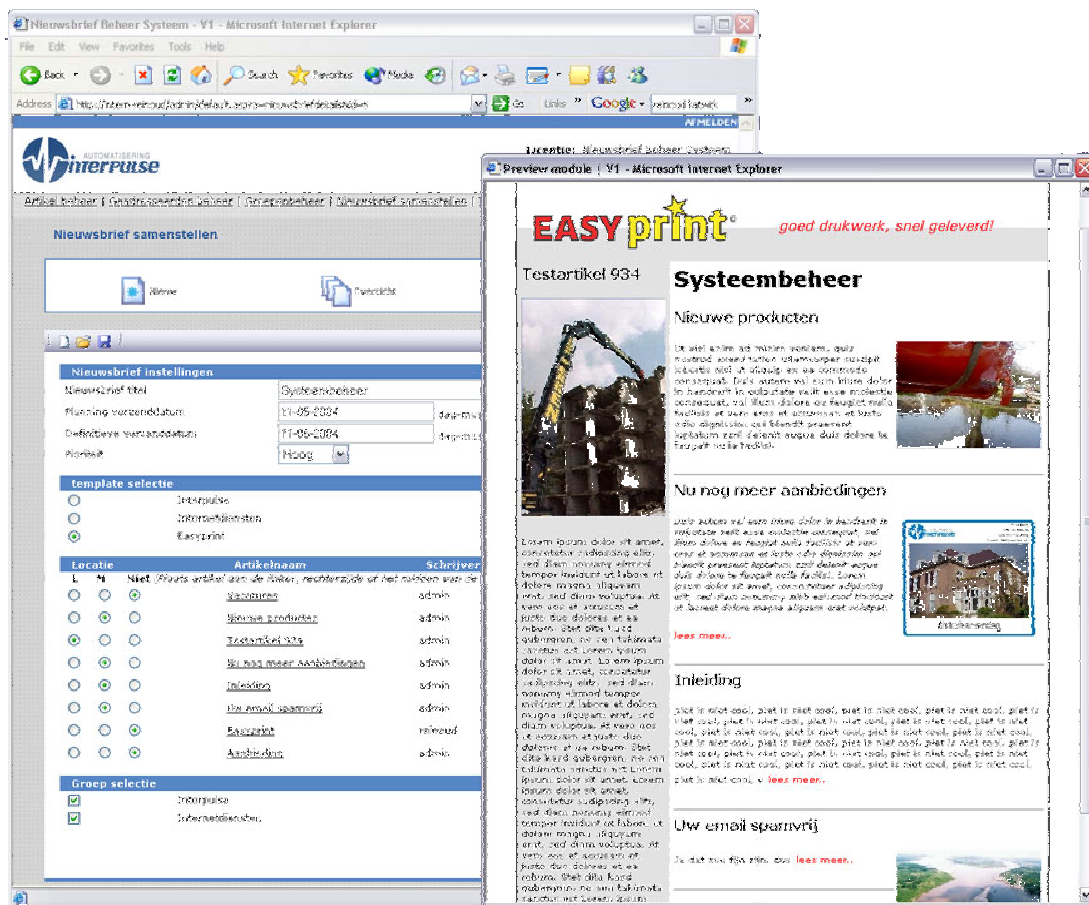
Het tweede plan kwam wat betreft inhoud meer overeen met de laatste twee pilots. Wat betekende dat dit tweede plan wat uitgebreider was en daarop volgende plannen minder uitgebreid. Het voordeel van het meerdere keren doorlopen van de fase pilotontwikkeling was vooral het specificeren van de informatie per pilot. Zo was de inhoud per

pilotontwikkelplan volledig afgestemd op de pilot en werd overbodige gegevens achterwege gelaten.

Het opstellen van de ontwikkelplannen heb ik als nuttig ervaren. Mede doordat ik niet alle activiteiten heb uitgevoerd, zoals IAD deze voorschrijft, heb ik compacte plannen neer kunnen leggen. De opdrachtgever was mede door de compactheid snel op de hoogte van de aanpak en daarop kreeg ik weer de nodige feedback. Wanneer ik alle activiteiten had uitgevoerd, zoals bijvoorbeeld een uitgebreide organisatorische inrichting, was ik veel meer tijd kwijt geweest en was de 'white noise' vele malen groter. De opdrachtgever en de ontwikkelaar hebben daarom de compacte plannen als prettig ervaren.

6.2.3 Nieuwsbrief beheersysteem

Waar het toch allemaal uiteindelijk om draaide was het nieuwsbrief beheersysteem. Daarbij mag ik uiteindelijk concluderen dat de opdracht geslaagd is geweest. Zeker na de uiteindelijke testen die ik heb uitgevoerd, waaruit bleek dat de gebruikers erg tevreden waren. Dat is altijd fijn om te horen.



Wanneer we kijken naar de systeemeisen kan ik zeggen dat 100 % van de basiseisen gehaald is. Ook zijn grotendeels de comforteisen gehaald. De luxe eisen zijn daarentegen niet gehaald. Wanneer deze alsnog ontwikkeld zullen worden in de toekomst kunnen de

pilotontwikkelplannen erbij gepakt worden. Met deze plannen is de ontwikkelaar in staat te achterhalen wat nog 'open' staat. Daarbij heb ik in de code overal commentaar vermeld. Voordeel hiervan is dat bij iedere functie staat wat deze exact doet. De opdrachtgever heeft aangegeven hiermee erg tevreden te zijn.

Bij aanvang van het project zijn er enkele doelstellingen opgesteld. In onderstaande toelichting zal ik aangeven in hoeverre de doelstellingen wel of niet behaald zijn na invoering van het systeem. De hoofddoelstelling, het ontwikkelen van een beheermodule waarmee bedrijven een nieuwsbrief kunnen genereren en verzenden, is behaald. Het totale systeem is hiervan het resultaat. De daarbij behorende componenten zijn hieronder opgesomd:

- Klanten, enerzijds van Interpulse, anderzijds "klanten van klanten" van Interpulse, de mogelijkheid bieden om zich op de website in te schrijven voor een nieuwsbrief, zodat ze altijd op de hoogte zijn van de laatste ontwikkelingen.

De eerste doelstelling is eenvoudig behaald. Een ieder kan zich, wanneer gewenst, inschrijven voor de nieuwsbrief middels een invulveld op de website.

- De nieuwsbrief zal in het oog moeten springen door de herkenbare huisstijl van de verzender.

Binnen het systeem kunnen huisstijlen en sjablonen aan elkaar gekoppeld worden. Door deze toepassing kan het systeem voldoen aan de doelstelling.

- Dit systeem moet ook los van het Interpulse CMS aangeboden kunnen worden.

Het systeem kan stand-alone 'draaien', maar is ook gekoppeld aan het Interpulse CMS. Ook aan deze doelstelling is voldaan.

- Dit systeem moet eenvoudig bij externe websites, die niet in het beheer van Interpulse zijn, te implementeren zijn.

Door een code in het systeem te genereren kan deze in de website geplaatst worden. Met deze code kan het systeem gekoppeld worden aan websites welke niet in het beheer van Interpulse zijn. De tweede hoofddoelstelling was het uitschakelen van de derde partij. Middels het systeem is de derde partij niet meer benodigd. Ook deze laatste doelstelling is behaald tijdens het project.

6.2.4 *Gebruikersdocumentatie*

De gebruikersdocumentatie stond niet heel hoog op de prioriteitenlijst van de opdrachtgever. Daarom zijn niet alle onderdelen van de helpfile uitgewerkt. Hierdoor is er uiteindelijk een incompleet helpsysteem neergezet. Hoofdoorzaak is dat binnen bijvoorbeeld pilot 1 de comforteis 'helpfile' wel binnen de timebox is gerealiseerd, maar bij pilot 4 is de helpfile weer niet ontwikkeld. In de toekomst zal ik de verschillende helponderdelen niet apart indelen en ontwikkelen, maar in een keer. Een andere mogelijkheid is de gebruikersondersteuning overal als basis systeemeis op te nemen.

Verklarende woordenlijst

IAD	Iterative Application Development
GUIDE	Graphical User Interface Design Evaluation
UML	Unified Modeling Language
ASP	Active Server Pages
FTP	File Transport protocol
HTML	Hyper Text Markup Language
WYSIWYG	What You See Is What You Get
SQL	Structured Query Language
EER	Enhanced Entity Relationship Model
IIS	Internet Information Services
CSS	Cascading Style Sheet
CMS	Content Management Systeem
DOM	Document Object Model

Literatuurlijst

Alan Cooper

About Face, The Essentials of User Interface Design

Alan Cooper

The Inmates Are Running the Asylum : Why High Tech Products Drive Us Crazy and How to Restore the Sanity (2nd Edition)

Jacob Nielsen

Designing Web Usability: The Practice of Simplicity

Joseph Schmuller

UML in 24 uur

Redmond-Pyle & Moore

Graphical User Interface Design and Evaluation (GUIDE)

R. Tolido

Iterative Application Development (IAD). Het evolutionair ontwikkelen van informatiesystemen

W. Horton

Designing and writing online documentation