



Ontwikkeling nieuwsbriefmodule voor On-Site BV

Afstudeerverslag

Student

Naam: Tony Kanter

Studentnummer: 20031228

Opleiding: Informatica

Bedrijf

Naam: On-Site BV

Bedrijfsmentor: Dhr. D. Bots

Haagse Hogeschool

Examinatoren:

Dhr. P.R.C. Breukel

Dhr. W. van Vliet

Referaat

Auteur : Tony Kanters

Titel : Ontwikkelen van een nieuwsbriefmodule voor On-Site BV

Onderwerp : Het ontwikkelen van een ASP.NET webapplicatie

Dit afstudeerverslag beschrijft mijn afstudeeropdracht bij On-Site BV. In dit verslag is er een scala aan activiteiten te vinden op het gebied van het ontwikkelen van een webapplicatie.

De webapplicatie is gemaakt met Microsoft Visual Studio 2005. Er is volgens de UP methode ontwikkeld. Als modelleertechniek is UML gebruikt en de broncode is geschreven in C#.Net.

Voor het lezen van dit verslag is er enige kennis vereist op het gebied van (R)UP, UML en ASP.NET.

De volgende descriptors zullen in dit verslag aan bod komen:

- Afstuderen aan de Haagse Hogeschool
- ASP.NET
- Beveiliging
- BoogiePOP
- Merak mailserver
- Microsoft SQL Server
- Microsoft Visual Studio
- MoSCoW methode
- Nieuwsbriefmodule
- Privacy
- UML
- UP
- Webapplicatie
- Web service

Voorwoord

Dit afstudeerverslag behandelt mijn werkzaamheden tijdens mijn afstudeerperiode bij On-Site BV. De afstudeerperiode is onderdeel van mijn opleiding Informatica aan de Haagse Hogeschool.

Ik heb van 13 november 2006 tot en met 23 maart 2007 aan deze opdracht gewerkt.

De opdracht bestond uit het ontwikkelen van de webapplicatie 'Nieuwsdesk'. Met deze webapplicatie kan een gebruiker een nieuwsbrief maken en deze versturen naar zijn contacten.

Graag wil ik een aantal mensen bedanken voor hun hulp, ondersteuning en begeleiding tijdens mijn afstudeerperiode:

On-Site BV:

- Dhr. D. Bots
- Dhr. M. de Jong
- Dhr. V. Bendeler

Haagse Hogeschool:

- Dhr. P.R.C Breukel
- Dhr. W. van Vliet

Tony Kanters

Rotterdam, 22 maart 2007

Inhoudsopgave

1. Inleiding	1
2. Organisatie	2
2.1. <i>On-Site BV</i>	2
2.1.1. Organigram	2
2.1.2. Afdeling Support	2
2.1.3. Afdeling IT	2
2.1.4. Afdeling Multimedia	3
2.1.5. Afdeling Sales	3
3. Opdrachtschrijving	4
3.1. <i>Probleemstelling</i>	4
3.2. <i>Opdrachtschrijving</i>	4
3.3. <i>Op te leveren producten</i>	5
3.4. <i>Gebruikte methoden en technieken</i>	5
3.5.1. UP	5
3.5.2. UML	6
4. Iteratie 1 van ontwikkeling Nieuwsdesk	7
4.1. <i>Inception Fase van ontwikkeling Nieuwsdesk</i>	7
4.1.1. Vision Document opstellen	7
4.1.2. Software Development Plan beschrijven	9
4.1.3. Risk Assessment opstellen	10
4.1.4. Use Case Model bepalen	11
4.1.5. Business Object Model uitwerken	12
4.2. <i>Elaboration fase van ontwikkeling Nieuwsdesk</i>	14
4.2.1. Use case specification maken	14
4.2.2. Use case realization maken	15
4.2.3. Class Model opstellen	16
4.2.4. Software Architecture Document opstellen	20
4.2.5. Navigation Map opstellen	21
4.2.6. Data Model opstellen	22
4.3. <i>Construction fase van ontwikkeling Nieuwsdesk</i>	24
4.3.1. Beta Release	24
4.3.1.1. Master Page gebruiken	24
4.3.1.2. Web service	26
4.3.1.3. Statistieken bijhouden	26
4.3.1.4. Importfunctionaliteit ontwikkelen	27
4.3.2. Testplan opstellen	30
4.4. <i>Transition fase van ontwikkeling Nieuwsdesk</i>	32
4.4.1. Testen	32
5. Iteratie 2 van ontwikkeling Nieuwsdesk	33
5.1. <i>Inception Fase van ontwikkeling Nieuwsdesk</i>	33
5.1.1. Software Development Plan aanpassen	33
5.1.2. Use Case Model bijwerken	33
5.1.3. Business Object Model bijwerken	35
5.2. <i>Elaboration fase van ontwikkeling Nieuwsdesk</i>	36
5.2.1. De use case specification maken	36
5.2.2. Use case realization maken	36
5.2.3. Class Model opstellen	36
5.2.1. Navigation Map opstellen	37
5.2.2. Data Model opstellen	38
5.3. <i>Construction fase van ontwikkeling Nieuwsdesk</i>	39
5.3.1. Beta Release	39
5.3.2. Testplan opstellen	39
5.4. <i>Transition fase van ontwikkeling Nieuwsdesk</i>	40
5.4.1. Testen	40
6. Evaluatie ontwikkeling Nieuwsdesk	41
6.1.1. Productevaluatie	41
6.1.2. Procesevaluatie	41
Literatuurlijst	43
Begrippenlijst	44
Figurenlijst	45
Tabellenlijst	46
Interne bijlagen	47
Bijlage I – Opdrachtschrijving	47
Bijlage II – Vision Document	49
Externe bijlagen	62

1. Inleiding

In dit verslag wordt beschreven welke werkzaamheden door mij zijn verricht tijdens mijn project voor On-Site BV.

Dit project is ontstaan doordat On-Site behoefte had aan een nieuwsbriefmodule voor de klanten van On-Site en haar dochterbedrijven.

Dit verslag is bedoeld voor iedereen die zich interesseert voor het ontwikkelen van een webapplicatie. Daarnaast is er voorkennis vereist van UML, (R)UP en het .NET Framework.

Middels dit rapport wil ik graag de lezer duidelijk maken wat de inhoud van het project was. Hoe het project is aangepakt en er wordt uitgelegd wat voor keuzes er zijn gemaakt. Bij deze keuzes wordt beschreven hoe en waarom deze zijn gemaakt.

In hoofdstuk twee wordt de organisatie waarin de werkzaamheden zijn verricht beschreven.

In hoofdstuk drie wordt de opdracht beschreven met daarbij de uitleg van de toegepaste methoden en technieken.

In hoofdstuk vier wordt het doorlopen proces van de eerste iteratie van dit project beschreven aan de hand van de gemaakte producten.

In hoofdstuk vijf wordt het doorlopen proces van de tweede iteratie van het project beschreven aan de hand van de gemaakte producten.

In hoofdstuk zes wordt het product en proces van het project geëvalueerd door mijzelf.

2. Organisatie

In dit hoofdstuk wordt de organisatie beschreven waar de afstudeeropdracht is uitgevoerd. Naast de organisatie wordt ook de afdeling waar ik werkzaam was, toegelicht.

2.1. On-Site BV

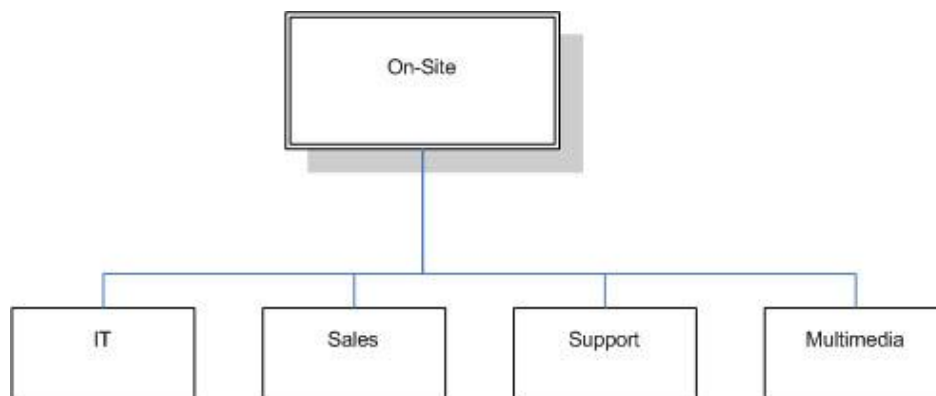
On-Site BV is een ICT bedrijf dat is gevestigd in het Braingate te Capelle a/d IJssel. Het is in 1999 opgericht door dhr. Dik Bots en dhr. Vincent Bendeler.

Het bedrijf is gericht op systeem- en netwerkbeheer. Daarnaast ontwikkelt, levert en beheert On-Site Internet- en Multimediatoeepassingen. On-Site is een implementatiepartner van Ektron. Als implementatiepartner leveren en ontwikkelen zij het Ektron CMS als Content Management System. Kort na de oprichting is het bedrijf zich gaan specialiseren in het maken en via het web ontsluiten van complex beeldmateriaal zoals luchtfoto's en 3D-modellen. De activiteiten die met luchtfotografie en 3D/ GIS (Geografische Informatie Systemen) visualisaties te maken hebben, zijn inmiddels ondergebracht in de aparte dochteronderneming TerraDesk.

On-Site is een groeiende onderneming en telt op dit ogenblik ongeveer 20 werknemers.

2.1.1. Organigram

In figuur 1 is de organisatie van On-Site BV schematisch weergegeven. Het bedrijf kent twee directeuren. Beide geven ze leiding over verschillende afdelingen.



Figuur 1: Organigram On-site

2.1.2. Afdeling Support

De afdeling Support is verantwoordelijk voor het ondersteunen van klanten. Het is het eerste aanspreekpunt voor problemen/vragen. Deze afdeling werkt nauw samen met de afdeling IT. Deze afdeling bestaat uit 2 werknemers.

2.1.3. Afdeling IT

De afdeling IT is verantwoordelijk voor het systeem- en netwerkbeheer bij On-Site en haar klanten. De werkzaamheden van deze afdeling zijn:

- § Het leveren van hard- en software

§ Het installeren en onderhouden van netwerken

De afdeling bestaat uit 2 werknemers.

2.1.4. Afdeling Multimedia

Deze afdeling is verantwoordelijk voor het ontwerpen en ontwikkelen van Intranet en e-commerce oplossingen voor ondernemingen die werkzaam zijn in de diensten- en IT sector. Dit is de afdeling waar ik werkzaam ben geweest tijdens de afstudeerperiode.

Op deze afdeling worden de volgende technieken gebruikt:

§ ASP.NET

§ ASP

§ VB.NET / C#.NET

§ Javascript

§ SQL

Verder worden er geen bekende ontwikkelmethoden gebruikt voor de producten die ontwikkeld worden. De reden hiervoor is dat On-Site over het algemeen opdrachten binnenkrijgt waarbij het voortraject (de ontwerpen) reeds heeft plaatsgevonden, voor deze opdrachten realiseren zij alleen de ontwerpen.

Bij opdrachten waarbij het voortraject nog niet heeft plaatsgevonden, maakt On-Site een plan van aanpak, sitemap(s), navigatiestructuur en schermontwerpen. Deze manier van ontwikkelen, kent geen uitgebreide ontwerpfase zoals UP dit wel heeft.

Deze afdeling is met 9 werknemers de grootste afdeling binnen On-Site. Dit is de afdeling waar ik mijn werkzaamheden uitgevoerd heb.

2.1.5. Afdeling Sales

De afdeling Sales verkoopt de producten en diensten die door On-Site BV aangeboden worden. Deze afdeling is met 5 werknemers de op één na grootste afdeling.

3. Opdrachtomschrijving

In dit hoofdstuk wordt de opdracht toegelicht. Zo wordt er beschreven wat de probleemstelling is met daarbij de uitgangssituatie. Vervolgens worden ook de opdrachtomschrijving en de daarbij behorende op te leveren producten beschreven.

3.1. Probleemstelling

On-Site heeft voor verschillende klanten een mogelijkheid ontwikkeld waarbij een klant een nieuwsbrief kan versturen naar verschillende emailadressen (dmv een distributielijst). Dit kan door een template van een nieuwsbrief in te vullen in Microsoft Outlook en te versturen naar een emailadres of meerdere emailadressen. Wanneer een klant een nieuwsbrief wil maken en versturen, zijn er veel handmatige configuratiestappen nodig om dit proces te voltooien. Deze stappen moeten zorgvuldig uitgevoerd worden, een kleine fout en de nieuwsbrief wordt niet verstuurd of naar de emailadressen van een andere klant gestuurd. Tevens is het door deze situatie niet mogelijk om de emailadressen te beheren.

Wanneer er een wijziging plaats moet vinden zal een klant dit doorgeven aan On-Site en zal er iemand bij On-Site de wijziging door moeten voeren. Ook is er via deze manier geen mogelijkheid om historie en statistieken bij te houden van een verstuurd nieuwsbrief. De klanten die gebruik maken van de nieuwsbriefmodule doen dit, omdat ze een nieuwsbrief naar honderden en/of duizenden emailadressen sturen. Er zijn klanten die een nieuwsbrief naar 60.000 emailadressen (per week) sturen.

3.2. Opdrachtomschrijving

Deze opdracht bestaat uit het ontwikkelen van de webapplicatie "Nieuwsdesk". Nieuwsdesk zal ontwikkeld worden in twee iteraties. Bij de ontwikkeling van de eerste iteratie zal rekening gehouden moeten worden met een mogelijke integratie met het Ektron CMS400.Net. Iteratie 1 moet bestaan uit:

- § Beheerdersgedeelte (voor On-Site); hiermee is het mogelijk om gebruikers van Nieuwsdesk te beheren, distributielijsten te beheren, templates van nieuwsbrieven voor klanten te beheren, historie weergeven van alle verstuurd nieuwsbrieven van de klanten en statistieken van de klanten te bekijken.
- § Adressenbeheer; dit gedeelte maakt het mogelijk om contacten van een klant (de gebruiker van Nieuwsdesk) te beheren.
- § Distributielijstbeheer; dit gedeelte maakt het mogelijk om distributielijsten van een klant te beheren.
- § Nieuwsbriefbeheer; dit gedeelte maakt het mogelijk om nieuwsbrieven aan te maken, te bekijken en te versturen, rapportage genereren van een verstuurd nieuwsbrief en historie bijhouden van verstuurd nieuwsbrieven.

- § Statistieken; dit gedeelte geeft een overzicht per klant van de statistieken van een contact of artikel en aan- en afmeldingen van de laatste 30 dagen. Zo kan er bijvoorbeeld ook worden gecontroleerd welk contact welk artikel heeft geopend.
- § Bouncemanagement; met bouncemanagement wordt het mogelijk om per nieuwsbriefmailing, een emailadres van een contact te voorzien van een status. Deze status geeft inzicht in de beschikbaarheid van een contact (bijv. emailadres bestaat niet meer).
- § Opt-in en opt-out; met deze functionaliteit is het mogelijk voor een contact om zichzelf in en uit te schrijven voor een nieuwsbrief van een klant via zijn eigen website. Voor deze functionaliteit zal er een API ontwikkeld moeten worden.
- § Importfunctie voor contacten; contacten vanuit Excel, CSV en/of Outlook moeten kunnen worden geïmporteerd naar het adresboek van een klant in Nieuwsdesk.

Na de implementatie van iteratie 1 kan er worden begonnen aan de ontwikkeling van iteratie 2 van Nieuwsdesk. Iteratie 2 van Nieuwsdesk zal meer als marketing tool gaan fungeren. Dit programma zal gebruikt gaan worden door werknemers van On-Site en haar dochterbedrijven. Het programma moet de volgende mogelijkheden bieden:

- § Ad hoc distributielijsten kunnen samenstellen. In iteratie 1 van Nieuwsdesk krijgt een gebruiker een aantal distributielijsten toegewezen. In iteratie 2 moet het mogelijk zijn om ad hoc contacten aan een distributielijst te koppelen.
- § Contactstatistieken; de mailing naar de gemaakte ad hoc distributielijst moet worden vastgelegd. Een contact kan zich uitschrijven dmv de opt-out functie uit iteratie 1. Doordat vastgelegd wordt dat een contact zich uitschrijft dmv een opt-out in een bepaalde nieuwsbrief kan actie worden ondernomen door de werknemers van On-Site om dit contact te benaderen.

3.3. Op te leveren producten

Aan het einde van dit project dienen de volgende producten opgeleverd te worden:

- § Rapport Inception fase
- § Rapport Elaboration fase
- § Rapport Construction fase
- § Rapport Transition fase
- § Nieuwsdesk
- § SQL Server database

3.4. Gebruikte methoden en technieken

3.5.1. UP

Een ontwikkelmethode draagt bij aan de kwaliteit van de op te leveren producten. Tijdens de opleiding zijn er twee ontwikkelmethoden behandeld. In het begin van de opleiding is de ontwikkelmethode (R)UP behandeld. Deze ontwikkelmethode is een aantal keren toegepast gedurende projecten. De andere ontwikkelmethode die aan bod is geweest, is de ontwikkelmethode MSF. Deze ontwikkelmethode is ontwikkeld door Microsoft. Deze methode is slechts theoretisch en gedurende

één blok van tien weken behandeld. Tijdens dit blok heb ik succesvol het examen van het certificaat 70-300 (Analyzing Requirements and Defining Microsoft .NET Solution Architectures) afgelegd. Het certificaat is onderdeel van het MCAD/MCSD traject en in dit onderdeel wordt het MSF uitvoerig behandeld en getoetst.

Door de opgedane kennis gedurende mijn opleiding, heb ik de twee ontwikkelmethoden kunnen vergelijken. Op basis van deze vergelijking ben ik tot een conclusie gekomen voor een te gebruiken ontwikkelmethode.

De overeenkomsten tussen UP en MSF zijn dat in beide methoden het project iteratief doorlopen wordt.

Voor dit project heb ik UP gebruikt. Dit vanwege de volgende redenen:

- Ik heb meer (praktijk)ervaring met UP dan MSF
- UP is geschikt voor projecten waar de eisen en wensen (grotendeels) vooraf vaststaan
- Bij het toepassen van UP wordt er volgens een strakke planning gewerkt en staat dus vast wanneer een product opgeleverd moet worden
- MSF is geschikt voor projecten waar de eisen en wensen niet vooraf vaststaan. Voor dit project stonden de eisen en wensen wel vast
- Bij het toepassen van MSF is een planning minder relevant, omdat er veranderende eisen en wensen zijn

Er is gebruik gemaakt van een iteratieve ontwikkelmethode, omdat door iteratief te ontwikkelen een opdrachtgever veel sneller een (eind)resultaat te zien krijgt dan bij het gebruik van een ontwikkelmethode die volgens het watervalmodel werkt. Daarnaast worden de risico's van het project door het iteratief ontwikkelen beperkt / uitgesloten.

3.5.2. UML

Voor dit project is er gebruik gemaakt van de modelleertechniek UML. De keuze voor UML is gemaakt, omdat het eindproduct van dit project, Nieuwsdesk, beheerd en onderhouden gaat worden door On-Site BV. Bij On-Site worden geen modelleertechnieken gebruikt, dus vond ik het verstandig om een gestandaardiseerde modelleertechniek, zoals UML, te gebruiken bij het ontwerpen.

Daarnaast is UML een modelleertechniek die gericht is op objectgeoriënteerde programmeertalen, voor dit project is gebruik gemaakt van de objectgeoriënteerde programmeertaal ASP.NET.

De opdrachtgever vond het geen probleem dat ik UML als modelleertechniek ging gebruiken. De opdrachtgever en medewerkers van On-Site kunnen de diagrammen overigens wel lezen en begrijpen, dit komt doordat zij een technische vooropleiding hebben gedaan.

4. Iteratie 1 van ontwikkeling Nieuwsdesk

Deze afstudeeropdracht is ontwikkeld volgens de UP methode. Aan de hand van de UP methode wordt er iteratief ontwikkeld. Bij iteratief ontwikkelen wordt het project opgedeeld in een aantal fasen en worden deze fasen per iteratie doorlopen, daarnaast kunnen er binnen een fase ook iteraties plaatsvinden. Bij elke fase hoort een milestone. Deze milestone bepaalt aan de hand van de gemaakte producten binnen één fase of er wel of niet wordt doorgegaan met het project.

Het voordeel van deze ontwikkelmethode is dat het risico, dat het eindproduct niet voldoet aan de wensen en eisen van de opdrachtgever, verkleind wordt.

Bij aanvang van het project zijn een aantal interviews met de opdrachtgever gehouden. Tijdens deze interviews zijn de eisen en wensen van de opdrachtgever besproken. In een interview gaf de opdrachtgever aan dat er twee versies van Nieuwsdesk moeten komen. Hij wilde dat de eerste versie eind januari 2007 in gebruik genomen kon worden.

De tweede versie van Nieuwsdesk zou eind maart 2007 klaar moeten zijn. Op basis van deze gegevens heb ik besloten om de ontwikkeling van Nieuwsdesk op te delen in twee iteraties. Daarnaast kunnen de functionaliteiten uit de tweede versie niet werken zonder de functionaliteiten uit de eerste versie. Ook heeft de tweede versie een andere doelgroep dan de eerste versie, door per iteratie met een andere doelgroep rekening te houden, heb ik mij beter kunnen concentreren op de te ontwikkelen functionaliteiten voor die iteratie.

4.1. Inception Fase van ontwikkeling Nieuwsdesk

Het doel van de Inception fase is het bepalen van de eisen en wensen, vaststellen van risico's en het bepalen van de aanpak van het project. Dit doel wordt behaald door het opstellen van het Vision document, Software Development Plan, Risk Assessment, Use Case Model en het Business Object Model.

4.1.1. Vision Document opstellen

Het Vision Document geeft inzicht in de probleemstelling en gewenste situatie met daarbij de [niet]functionele systeemeisen van de te ontwikkelen applicatie. Daarnaast geeft het een overzicht van alle belanghebbende en toekomstige gebruikers.

Voordat het Vision Document opgesteld kon worden, zijn er twee interviews met de opdrachtgever geweest. Tijdens deze interviews heeft de opdrachtgever verteld wat Nieuwsdesk zou moeten kunnen en waaraan het zou moeten voldoen. Ook is in deze gesprekken naar voren gekomen welke personen er betrokken zouden zijn bij dit project.

Voordat de interviews plaatsvonden heb ik een onderzoek gedaan naar bestaande nieuwsbriefmodules. Met de informatie uit dit onderzoek heb ik in kunnen spelen op de eisen en wensen van de opdrachtgever. In dit onderzoek is er informatie verzameld over de functionaliteiten van bestaande nieuwsbriefmodules. Er zijn in totaal negen nieuwsbriefmodules gebruikt voor dit onderzoek. De negen modules zijn met elkaar vergeleken en op basis van deze resultaten is mij duidelijk geworden wat de bestaande nieuwsbriefmodules wel en niet aanbieden. Ik ben zo ook te

weten gekomen waarmee ik Nieuwsdesk iets extra's kan bieden ten opzichte van de bestaande nieuwsbriefmodules.

Functionele systeemeisen

Na het tweede interview met de opdrachtgever is overeenstemming bereikt over de [niet]functionele systeemeisen van Nieuwsdesk.

Hieronder staat een beknopt overzicht van de functionele systeemeisen:

- Beheerdersgedeelte (klant-, template-, distributielijstbeheer en statistieken)
- Adressenbeheer (contact toevoegen, wijzigen, verwijderen, koppelen met distributielijst)
- Importfunctionaliteit voor contacten
- Distributielijstbeheer (distributielijst toevoegen, wijzigen, verwijderen)
- Nieuwsbriefbeheer (nieuwsbrief aanmaken, verzenden)
- Historie (historie verstuurd nieuwsbrieven)
- Statistieken (statistieken van een nieuwsbrief opvragen)
- Bouncemanagement (welke bounces en waarom)

Ik heb de importfunctionaliteit voorgesteld als aanvulling voor het adressenbeheer. Dit heb ik gedaan naar aanleiding van een opmerking van de opdrachtgever uit het eerste interview. De opdrachtgever gaf aan dat adressenbeheer een mooie oplossing zou zijn om alle contacten die in verschillende Excel bestanden zijn opgeslagen, centraal te beheren.

Doordat ik nu wist dat er allerlei verschillende Excel bestanden met contacten bestonden binnen On-Site (en deze situatie ook bij klanten van On-Site zou kunnen voorkomen), kon ik mij indenken dat het een vervelende en tijdrovende klus zou zijn om al deze contacten handmatig in te voeren. Om tijd te besparen, heb ik voorgesteld om een importfunctionaliteit te ontwikkelen die de mogelijkheid biedt om Excel, CSV en Outlook bestanden te importeren.

Niet functionele systeemeisen

Hieronder staat een beknopt overzicht van de [niet]functionele systeemeisen:

- De nieuwsbriefmodule moet werken op een Windows machine met Internet Explorer
- De functionaliteiten moeten aanspreekbaar zijn voor het Ektron CMS400.NET
- Gebruikers/klanten moeten onafhankelijk van elkaar kunnen werken met de nieuwsbriefmodule. Als de ene klant iets doet moet dit geen invloed hebben op de andere
- De nieuwsbriefmodule moet beveiligd zijn
- De nieuwsbriefmodule moet aansluiten op de reeds gebruikte software die ook voor andere applicaties gebruikt wordt (Microsoft SQL Server, Microsoft IIS)
- Op elk gewenst moment dient een nieuwsbrief aangemaakt/gelezen te worden

De niet functionele systeemeisen zijn in overleg met de opdrachtgever bepaald.

Belanghebbenden en toekomstige gebruikers

In het tweede interview zijn vooral de verfijning van de [niet]functionele systeemeisen en de toekomstige gebruikers en belanghebbenden van dit project besproken.

Zo is er duidelijk geworden dat bij dit project acht belanghebbenden en gebruikers betrokken zijn. Vanuit On-Site is er een aparte teamleider aangesteld als aanspreekpunt voor gebruikers, applicatiebeheerder, testers en opdrachtnemer. Beslissingen aangaande de functionaliteiten moeten altijd met de opdrachtgever besproken worden. Er is wel onderling contact tussen de opdrachtnemer, teamleider en applicatiebeheerder, maar beslissingen die van invloed zijn op de werking van Nieuwsdesk dienen door de opdrachtgever goedgekeurd te worden.

Rol	Beschrijving	Verantwoordelijkheden
Opdrachtgever		• Specificeren eisen en wensen • Accepteert het systeem • Voorziet in beschikbaarheid hard- en software • Neemt applicatie in beheer • Specificeren (gedetailleerde) eisen en wensen • Keurt Vision Document
Teamleider gebruikers	Begeleid het project vanuit de kant van On-Site BV	• Contactpersoon voor opdrachtnemer • Tussenpersoon voor klanten van On-Site BV • Beschikbaar stellen van gebruikers voor het project • Bewaken van voortgang van project
Opdrachtnemer		• Ontwerpen van Nieuwsdesk • Ontwikkelen van Nieuwsdesk • Implementeren van Nieuwsdesk • Contactpersoon voor de teamleider van gebruikers • Projectmanagement
Applicatie beheerder	Het onderhouden en beheren van Nieuwsdesk na implementatie	• Beheren van Nieuwsdesk • Aanspreekpunt voor gebruikers • Klantenbeheer
Klant	De eindgebruiker van Nieuwsdesk	• Het gebruiken van de eerste versie van Nieuwsdesk
Salesmedewerker	De eindgebruiker van Nieuwsdesk	• Het gebruiken van beide versies van Nieuwsdesk
Tester	Tester van Nieuwsdesk	• Testen van Nieuwsdesk aan de hand van de aangeleverde testsets

Tabel 1: Belanghebbenden en toekomstige gebruikers Nieuwsdesk

4.1.2. Software Development Plan beschrijven

Het Software Development Plan beschrijft de aanpak voor het project. In dit document wordt bepaald wat de milestones, activiteiten en werkzaamheden zijn van dit project. Daarnaast worden verantwoordelijkheden en de planning vastgelegd. Dit document is gemaakt om officieel de projectindeling vast te leggen.

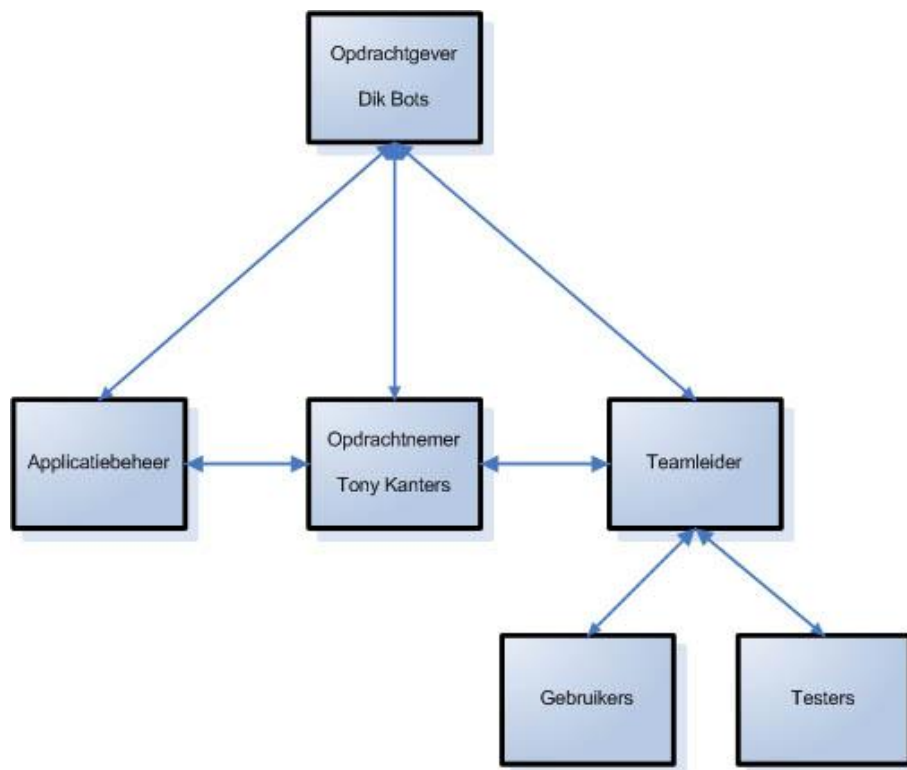
Milestones

Een milestone is een belangrijke gebeurtenis in het project. Bij het bereiken van een milestone wordt een fase afgesloten. Hieronder staat een beknopt overzicht van de milestones van dit project:

- LifeCycle Objective Milestone, dit wordt getoetst door het Rapport Inception Fase
- LifeCycle Architecture Milestone, dit wordt getoetst door het Rapport Elaboration Fase
- Initial Operational Capability Milestone, dit wordt getoetst door het Rapport Construction Fase
- Product Release Milestone, dit wordt getoetst door het Rapport Transition Fase
- De webapplicatie Nieuwsdesk
- SQL Server database

Organisatiestructuur

In het Software Development Plan beschrijf ik ook de organisatiestructuur van het project. In de organisatiestructuur worden schematisch de betrokkenen van het project weergegeven.



Figuur 2: Organisatiestructuur van het project

4.1.3. Risk Assessment opstellen

In het Risk Assessment document zijn de risico's van het project beschreven. De risico's zijn opgesteld op basis van mijn eigen ervaring. Het Risk Assessment is in het Vision Document opgenomen.

De risico's die van toepassing zijn op dit project, zijn:

1. De opdrachtnemer heeft geen kennis van nieuwsbriefmodules
2. De opdrachtnemer heeft geen kennis van bouncemanagement
3. De opdrachtnemer heeft nauwelijks kennis van mailservers en mailprotocollen

4. De opdrachtnemer heeft geen kennis van het Ektron CMS400.Net
5. Er kan niet zonder meer informatie opgeslagen worden van klanten/contacten ten behoeve van de statistieken

Om de risico's te verkleinen/vermijden worden de volgende maatregelen getroffen:

1. Er zal een onderzoek gestart moeten worden waar bestaande nieuwsbriefmodules onderzocht worden
2. Er zal een onderzoek gestart moeten worden waar het begrip bouncemanagement zodanig onderzocht wordt dat bouncemanagement voor dit project toepasbaar wordt
3. Er zal een onderzoek gepland moeten worden naar mailprotocollen en bestaande mailservers.
4. De opdrachtnemer zal een handleiding voor het gebruik van het Ektron CMS400.Net moeten doorlopen.
5. Er zal moeten worden geïnformeerd in hoeverre en op welke manier er informatie van klanten/personen geregistreerd mag worden

4.1.4. Use Case Model bepalen

In dit document wordt een samenvatting gegeven van alle te ontwikkelen functionaliteiten. Tevens worden aan alle use cases een gewicht en classificatie toegekend. Het gewicht van een use case wordt bepaald door het gewicht van de actor die gekoppeld is aan de use case. De gewichten zijn bepaald in overleg met de opdrachtgever. De actor die de meeste interactie met Nieuwsdesk heeft, heeft het hoogste gewicht gekregen.

De classificatie van een use case is bepaald op basis van de MoSCoW methode.

De MoSCoW methode is een methode die gebruikt wordt bij het bepalen van prioriteiten. Deze methode is gebruikt, omdat deze snel en makkelijk uit te leggen is aan personen die deze methode niet kennen. De letters in MoSCoW staan voor:

- **Must have** - deze eis moet in het eindresultaat terugkomen
- **Should have** - deze eis is zeer gewenst, maar een workaround is ook goed genoeg
- **Could have** - deze eis mag alleen aan bod komen als er tijd genoeg is
- **Would like to have but won't have this time around** - deze eis zal nu niet aan bod komen maar kan in de toekomst interessant zijn

Tabel 2 geeft de opgestelde actoren uit het Use Case model weer.

Code	Actor	Omschrijving	Gewicht
A1	Klant	De eindgebruiker van Nieuwsdesk	4
A2	CMS	Het Ektron CMS400.Net. Dit zal de functionaliteiten aanspreken van Nieuwsdesk	2
A3	Applicatiebeheerder	Beheerder van de Nieuwsdesk	3
A4	Contact	Contact van een klant	1
A5	Salesmedewerker	De eindgebruiker van Nieuwsdesk	4

Tabel 2: Actoren overzicht Nieuwsdesk

Tabel 3 geeft een aantal opgestelde use cases uit het Use Case Model. De classificaties aan de use cases zijn in overleg met de opdrachtgever vastgesteld. Dit is gedaan, omdat hij kan bepalen welke functionaliteiten er nodig zijn. De prioriteit wordt bepaald door het optellen van het gewicht en de classificatie. Een classificatie M is gelijk aan het cijfer 4, de S is gelijk aan het cijfer 3, de C is gelijk aan het cijfer 2 en de W is gelijk aan het cijfer 1.

De prioriteit aan een use case is bepalend bij het ontwikkelen van Nieuwsdesk. Een use case met een hoge prioriteit (per gedeelte, bijv. klantbeheer) heeft voorrang op use cases met een lagere prioriteit. Dus bij het ontwikkelen van klantbeheer heeft het ontwikkelen van UC3, voorrang op UC4.

Code	Use Case Naam	Omschrijving	Gewicht	Samenhang	Classificatie	Prioriteit
UC1	Aanmelden	Het aanmelden bij Nieuwsdesk. Een actor kan hiermee toegang krijgen tot Nieuwsdesk	4		M	8
UC2	Afmelden	Het afmelden bij Nieuwsdesk. Een actor verbreekt de aanmelding en zal geen toegang hebben tot Nieuwsdesk totdat er gebruik gemaakt van UC1 door dezelfde actor	4		S	7
UC3	Klant toevoegen	Het toevoegen van een klant, zodat deze gebruik kan maken van Nieuwsdesk	3		M	7
UC4	Klant zoeken	Het zoeken naar een klant	3		S	6
UC12	Distributielijst wijzigen	Het wijzigen van de eigenschappen van een lijst	3	UC13	C	5

Tabel 3: Use case Model Nieuwsdesk

Use case diagrammen

In het Use Case Model zijn ook alle use case diagrammen opgenomen. In principe horen alle use cases in één diagram geplaatst te worden met de daarbij behorende actoren. Door het redelijk grote aantal use cases (totaal 53) is het onmogelijk om alle use cases op te nemen in één diagram, daarom wordt er gesproken over use case diagrammen.

Om de use case diagrammen overzichtelijk te houden, zijn de use cases die onder één categorie vallen, opgenomen in een use case diagram. Dit betekent dat bijvoorbeeld alle use cases van de categorie Contact, in één diagram zijn opgenomen. Ik heb ervoor gekozen om de use cases van één categorie op te nemen in een use case diagram, omdat de opdrachtgever snel inzicht wilde hebben in de opgestelde use cases en een plaatje veel sneller inzicht geeft dan een tabel met tekst.

4.1.5. Business Object Model uitwerken

Na afronding van de eerste interviews en het opstellen van het Use Case Model kon het Business Object Model van Nieuwsdesk gemaakt worden. Dit model is gemaakt om in een vroeg stadium vast

te leggen uit welke entiteiten en objecten Nieuwsdesk zal bestaan. Ook wordt er in dit model de samenhang tussen de objecten getoond.

Het eerste Business Object Model bestond uit vijf objecten:

- Contact, dit object representeert de contacten van een klant
- Klant, dit object is een gebruiker van Nieuwsdesk
- Nieuwsbrief, dit object bestaat uit artikelen en wordt verstuurd naar een distributielijst
- Artikel, dit object bevat informatie die wordt toegevoegd aan een nieuwsbrief
- Distributielijst, dit object bevat contacten

Met deze objecten kon ik aan de meeste systeemeisen en Use Cases voldoen. Helaas kon ik niet voldoen aan de eis om de opmaak van een nieuwsbrief, een template, per nieuwsbrief te wijzigen. Met deze objecten zou per nieuwsbrief de opmaak opnieuw gemaakt moeten worden. Daarnaast mag een klant niet zelf een opmaak maken van een nieuwsbrief. Om dit probleem op lossen is er een object Template toegevoegd in de tweede versie van het Business Object Model. Een klant krijgt een aantal templates ter beschikking gesteld voor de opmaak van een nieuwsbrief.

Na het toevoegen van het object Template ontbrak er nog iets aan het Business Object Model. Om gebruik te kunnen maken van Nieuwsdesk moet er worden ingelogd. Nieuwsdesk zal nooit dezelfde gebruikers kennen, dus is het niet mogelijk om op basis van gebruiker te controleren welke rechten er toegekend moeten worden bij het inloggen. Voor dit probleem is het object Gebruikersgroep bedacht. Door het toevoegen van dit object is het mogelijk om op basis van gebruikersgroepen rechten in Nieuwsdesk toe te kennen aan gebruikers die inloggen.

De tweede versie van het Business Object Model bestond uiteindelijk uit zeven objecten. Het Business Object Model is gemaakt als basis voor het Class Model, dat gemaakt is in de Elaboration Fase.

4.2. Elaboration fase van ontwikkeling Nieuwsdesk

Het doel van de Elaboration fase is het bepalen van een softwarearchitectuur, het ontwerpen van het Class Model, use case realization, use case specification en data model van het project en ontwerpen van de Navigation map.

Deze fase is in twee iteraties doorlopen:

1. Beheerdersgedeelte
2. Klantgedeelte

Deze fase is doorlopen in twee iteraties, omdat dan per iteratie de aandacht volledig op één gedeelte gevestigd kon worden.

4.2.1. Use case specification maken

De use case specification bestaat uit alle eerder opgestelde use cases in het use case model. De opgestelde use cases uit het use case model zijn in de use case specification stap voor stap beschreven. In de use case specification wordt per use case een actor, preconditie, scenariobeschrijving, uitzonderingen en een postconditie beschreven. Een basisscenario van een use case specification ziet er als volgt uit:

Actor	
Preconditie	
Scenario beschrijving	
Uitzonderingen	
Postconditie	

Tabel 4: Basisscenario use case specification

Ook zijn er in een use case specification van een use case de bijbehorende foutscenario's beschreven. Een foutscenario ziet er volgt uit:

Actor	
Preconditie	
Scenario beschrijving	

Tabel 5: Foutscenario use case specification

Voor iteratie 1 zijn er 53 use case specifications opgesteld. Om een beeld te krijgen van de inhoud van een use case specification wordt de use case specification van de use case 'Nieuwsbrief verzenden' getoond:

Use case specification Nieuwsbrief verzenden:

Basisscenario - Nieuwsbrief verzenden

Actor	Klant, Salesmedewerker
Preconditie	De actor is aangemeld
Scenario beschrijving	1. Het systeem toont het scherm "Nieuwsbrief verzenden" 2. De actor selecteert een nieuwsbrief die verzonden moet worden

	3. Het systeem toont de nieuwsbrief 4. De actor klikt op 'verzenden' 5. Het systeem valideert de ingevulde gegevens [onjuiste invoer] 6. Het systeem toont alle distributielijsten van de gebruiker 7. De actor selecteert een distributielijst 8. De actor klikt op verzenden 9. Het systeem verzendt de nieuwsbrief naar het emailadres van de gekozen distributielijst
Uitzonderingen	[onjuiste invoer] De use case gaat verder bij (2)
Postconditie	Een nieuwsbrief is verzonden

Foutscenario 1 – Nieuwsbrief verzenden [onjuiste invoer]

Actor	Klant, Salesmedewerker
Preconditie	De actor is aangemeld
Scenario beschrijving	• Stap 1 is hetzelfde als in het basisscenario, en vervolgens: • Het systeem constateert dat het type van de ingevoerde gegevens niet juist is • De actor corrigeert de gegevens. • Het scenario gaat verder bij stap 2 van het basisscenario.

Voor alle use case specifications verwijs ik u naar de use case specifications in het rapport Elaboration fase. Dit rapport is als externe bijlage opgenomen bij dit verslag.

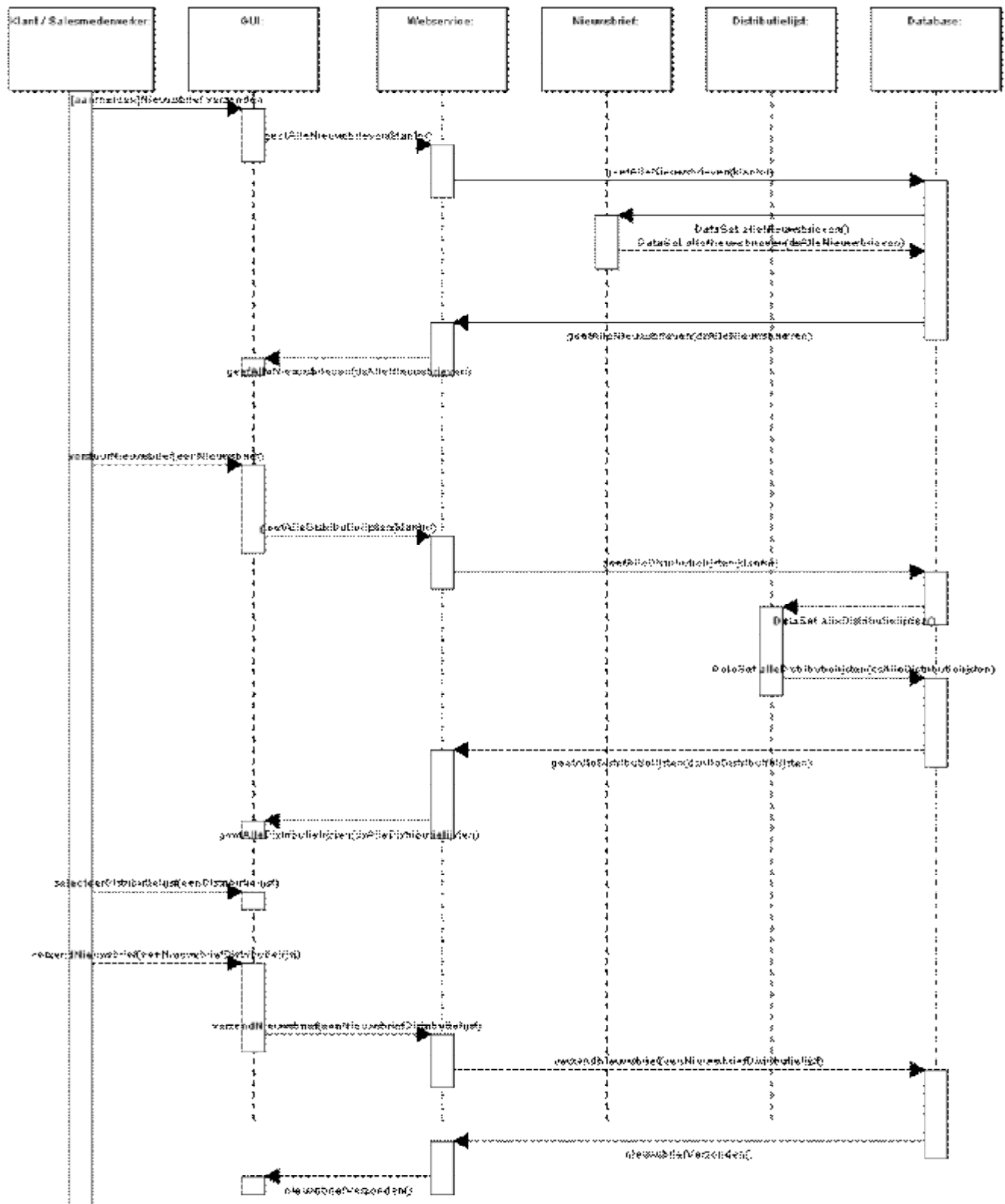
De Use Case Specification is gemaakt ter ondersteuning bij het ontwikkelen. Een functionaliteit van Nieuwsdesk is ontwikkeld volgens de stappen binnen een use case. De initiële versie van de Use Case specification is door mij opgesteld. Na het voltooien van de initiële versie zijn de use cases aan de opdrachtgever voorgelegd en door hem gekeurd.

4.2.2. Use case realization maken

De Use case realization bestaat uit het schematisch opstellen van de opgestelde use case specifications. Voor het schematisch weergeven van use case specifications zijn voor dit project sequentiediagrammen gebruikt.

De reden dat sequentiediagrammen zijn gebruikt tijdens dit project is, omdat een sequentiediagram de interactie tussen een gebruiker en de webapplicatie weergeeft. Door het gebruik van sequentiediagrammen werd tijdens het ontwikkelen van de functionaliteiten duidelijk welke klassen gebruikt moesten worden.

In het subhoofdstuk 'Use case specification maken' is de use case 'Nieuwsbrief verzenden' getoond. Om te laten zien dat de Use Case Realization gebaseerd is op de Use Case Specification wordt het sequentiediagram van de use case 'Nieuwsbrief Verzenden' getoond.



Figuur 3: Sequentiediagramm use case Nieuwsbrief verzenden

4.2.3. Class Model opstellen

Op basis van het Business Object Model kon het Class Model worden opgesteld. Het Class Model bevat alle klassen met attributen en operaties. De attributen worden gebruikt als variabelen van een

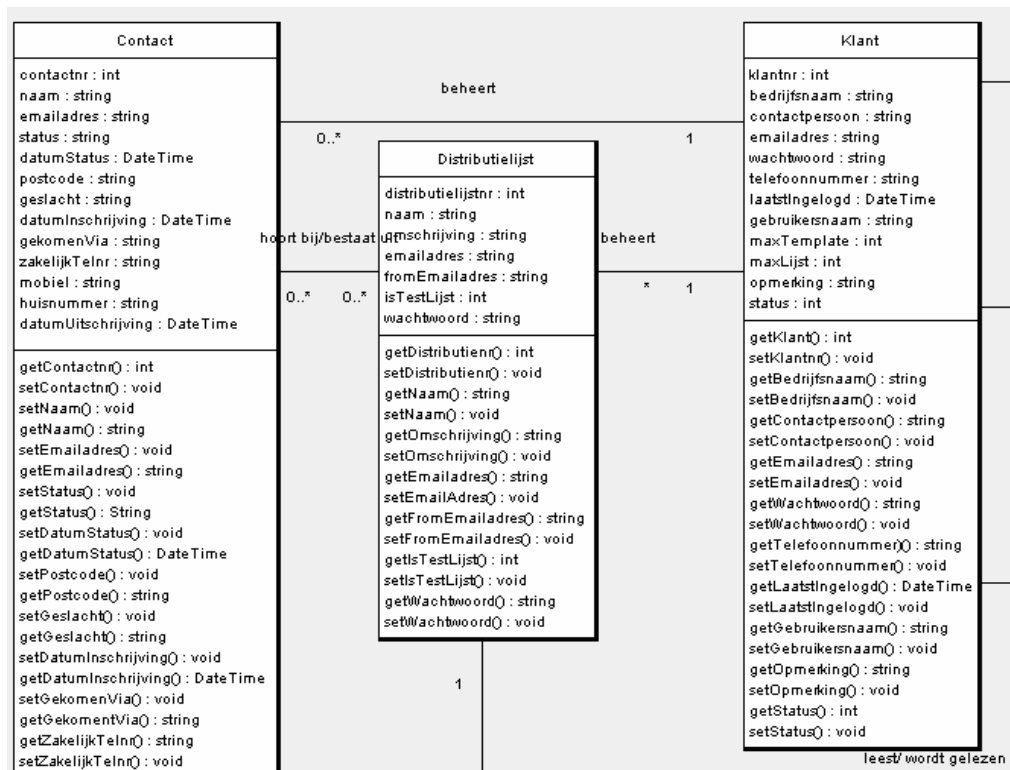
Class en representeren daarmee de eigenschappen van een Class. De operaties zijn de methods binnen een Class.

In het Business Object Model waren zeven objecten gedefinieerd. In het Class Model is er nog een Class bijgekomen, genaamd URL. Een URL hoort bij een artikel, omdat elk artikel één of meerdere URL's bevat. Deze URL Class is opgenomen om zo aan de eis van de statistieken te kunnen voldoen. Een contact opent namelijk een URL en zo kan er worden bijgehouden welk artikel door welk contact wordt gelezen en daarnaast kan er worden bijgehouden hoe vaak een artikel gelezen wordt.

Een ander verschil met het Business Object Model is dat een nieuwsbrief een associatie heeft gekregen met een distributielijst. Dit komt doordat een nieuwsbrief naar een distributielijst wordt verzonden en aan een distributielijst zitten contacten gekoppeld.

Hieronder is een deel van het Class Model opgenomen. Het gehele Class Model is opgenomen in het Class Model document van de externe bijlage 'Rapport Elaboration fase'.

Het Class model is na de toevoeging van de Class 'URL' altijd hetzelfde gebleven. Het is wel eens voorgekomen dat er een attribuut is aangepast/toegevoegd/verwijderd maar er zijn geen grote veranderingen geweest. (bijv. gekomenVia in de Class Contact, deze is toegevoegd om te kunnen zien hoe een contact is ingeschreven)



Figuur 4: Deel van het Class Model

Het Class model bestaat uit de volgende klassen:

- **Contact:**

Een contact heeft een associatie met een klant. Ik heb ervoor gekozen om een contact altijd bij één klant te laten horen, het kan nu dus voorkomen dat een contact meerdere keren voorkomt. Ik had

ervoor kunnen kiezen om een contact aan meerdere klanten te kunnen koppelen en hiermee het contact uniek te houden.

Ik heb ervoor gekozen om een contact altijd bij één klant te laten horen, omdat er informatie wordt opgeslagen van een contact die mogelijk alleen voor één bepaalde klant bedoeld is. In deze situatie is het mogelijk om informatie van een contact per klant bij te werken. Het kan zo zijn dat een klant privé informatie heeft van een contact en dit op wil slaan, bijvoorbeeld een mobiel nummer. Wanneer een klant nu informatie opslaat van een contact is dit niet zichtbaar voor andere klanten. Had ik ervoor gekozen om een contact uniek te maken, dan had dit mobiele nummer voor iedereen (die gekoppeld is met dit contact) zichtbaar geweest.

Deze keuze is ook genomen om performance redenen. Doordat een contact niet uniek is, wordt er bij toevoegen niet gecontroleerd of het contact al bestaat maar wordt het contact gelijk toegevoegd. Het toevoegen in de database bestaat dus maar uit één schrijfactie. Wanneer een contact uniek had geweest, had er bij het toevoegen eerst een leesactie gedaan moeten worden en vervolgens een schrijfactie. Bij toevoegen van één contact is dit geen probleem maar Nieuwsdesk beschikt ook over een importfunctionaliteit om meerdere contacten 'tegelijk' toe te voegen. Bij een groot aantal contacten zal de performance van het systeem dalen of in het ergste geval zal Nieuwsdesk 'plat' gaan. Nieuwsdesk dient op elk moment van de dag beschikbaar te zijn, omdat op elk moment van de dag artikelen gelezen kunnen worden, nieuwsbrieven gemaakt/verzonden worden.

Mocht een contact nu meerdere malen voorkomen, heeft dit geen invloed op een nieuwsbriefmailing. Op de mailserver Merak worden redundante emailadressen eruit gefilterd gedaan, dus een contact zal nooit twee keer dezelfde nieuwsbrief van een nieuwsbriefmailing ontvangen.

Ook voor het bouncemanagement is het niet erg dat een contact meerdere malen voorkomt.

BoogiePOP (de gebruikte bouncetool) update statussen op basis van emailadres, dus mocht een nieuwsbriefmailing naar een bepaald emailadres een bounce opleveren, wordt dit doorgevoerd op alle contacten met dat emailadres.

Een andere associatie van een contact is met artikel. Een contact kan namelijk meerdere artikelen lezen (en ook meerdere malen hetzelfde artikel) en een artikel kan door meerdere contacten gelezen worden.

Daarnaast heeft contact een associatie met distributielijst. Een contact kan aan meerdere distributielijsten gekoppeld zijn.

- *Distributielijst:*

Een distributielijst heeft drie associaties. Een distributielijst heeft een associatie met klant. Een distributielijst hoort namelijk bij één klant en een klant kan over meerdere distributielijsten beschikken. Daarnaast kan een distributielijst nul of meerdere contacten bevatten en een contact kan aan nul of meerdere distributielijsten gekoppeld zijn.

- *Klant:*

Een klant heeft vijf associaties. Een klant heeft een associatie met contact. Een klant kan meerdere contacten beheren en een contact hoort bij één klant.

Een klant heeft een associatie met template. Een klant kan over meerdere templates beschikken en een template hoort bij één klant, het maximaal aantal templates wordt bepaald door het attribuut maxTemplate. Het maximaal aantal templates voor een klant is afhankelijk van het afgesloten abonnement met On-Site.

Een klant heeft een associatie met distributielijst. Een klant kan meerdere distributielijsten hebben, het maximaal aantal distributielijsten wordt bepaald door het attribuut maxLijst. Het maximaal aantal distributielijsten voor een klant is afhankelijk van het afgesloten abonnement met On-Site.

Een klant heeft een associatie met nieuwsbrief. Een klant kan meerdere nieuwsbrieven maken maar een nieuwsbrief hoort altijd maar bij één klant.

Een klant heeft een associatie met gebruikersgroep. Een klant hoort altijd bij één gebruikersgroep en een gebruikersgroep kan meerdere klanten hebben.

- *Gebruikersgroep:*

Een gebruikersgroep heeft een associatie met klant. Een gebruikersgroep kan meerdere klanten hebben en een klant hoort altijd bij één gebruikersgroep.

- *Nieuwsbrief:*

Een nieuwsbrief heeft vier associaties. Een nieuwsbrief heeft een associatie met template. De lay-out van een nieuwsbrief wordt bepaald door één template en een template kan bij meerdere nieuwsbrieven horen.

Een nieuwsbrief heeft een associatie met klant. Een nieuwsbrief wordt gemaakt door een klant en een klant kan meerdere nieuwsbrieven hebben.

Een nieuwsbrief heeft een associatie met artikel. Een nieuwsbrief kan meerdere artikelen bevatten en een artikel kan bij meerdere nieuwsbrieven horen.

Een nieuwsbrief heeft een associatie met distributielijst. Een nieuwsbrief kan aan meerdere distributielijsten verzonden worden.

- *Template:*

Een template heeft twee associaties. Een template heeft een associatie met nieuwsbrief. Een template kan de lay-out bepalen van meerdere nieuwsbrieven en een nieuwsbrief heeft altijd één template.

Een template heeft een associatie met klant. Een template hoort bij één klant en een klant kan over meerdere templates beschikken.

- *Artikel:*

Een artikel heeft drie associaties. Een artikel heeft een associatie met een nieuwsbrief. Een artikel kan bij meerdere nieuwsbrieven horen en een nieuwsbrief kan uit meerdere artikelen bestaan.

Een artikel heeft een associatie met contact. Een artikel kan door meerdere contacten gelezen worden en een contact kan meerdere artikelen lezen.

Een artikel heeft een associatie met URL. Een artikel kan meerdere URL's hebben maar een URL hoort altijd bij één artikel.

- *URL:*

Een URL heeft één associatie. Een URL heeft een associatie met artikel. Bij het aanmaken van een artikel wordt altijd een link toegevoegd.

4.2.4. Software Architecture Document opstellen

Het Software Architecture Document beschrijft de architectuur van Nieuwsdesk. Voor het beschrijven van de architectuur is gebruik gemaakt van het 4+1 view model.

Het 4+1 view model

Dit model beschrijft de architectuur van Nieuwsdesk vanuit 5 views:

- § Logical view
- § Process view
- § Implementation view
- § Deployment view
- § Use case view

Het 4+1 view model stelt de verschillende belanghebbenden in staat vanuit hun eigen perspectief de invloed van de gekozen architectuur te bepalen. Het 4+1 view model deelt een systeem op in vier views: logical, process, implementation en deployment. Aan dit model wordt de use case view toegevoegd (dit is de +1) om ook de functionele aspecten van het systeem als onderdeel te zien. Het 4+1 view model is gebruikt, omdat het de architectuur van Nieuwsdesk vanuit verschillende perspectieven behandelt.

Het 4+1 view model beschrijft een applicatie vanuit verschillende perspectieven. Door deze perspectieven te combineren, wordt het duidelijk hoe een applicatie als geheel moet werken.

De Logical view bestaat uit het Class Model en de use case realizations (sequentiediagrammen). De logical view geeft dus inzicht in de structuur van Nieuwsdesk.

De Process view beschrijft de processen van Nieuwsdesk en de onderlinge communicatie tussen deze processen.

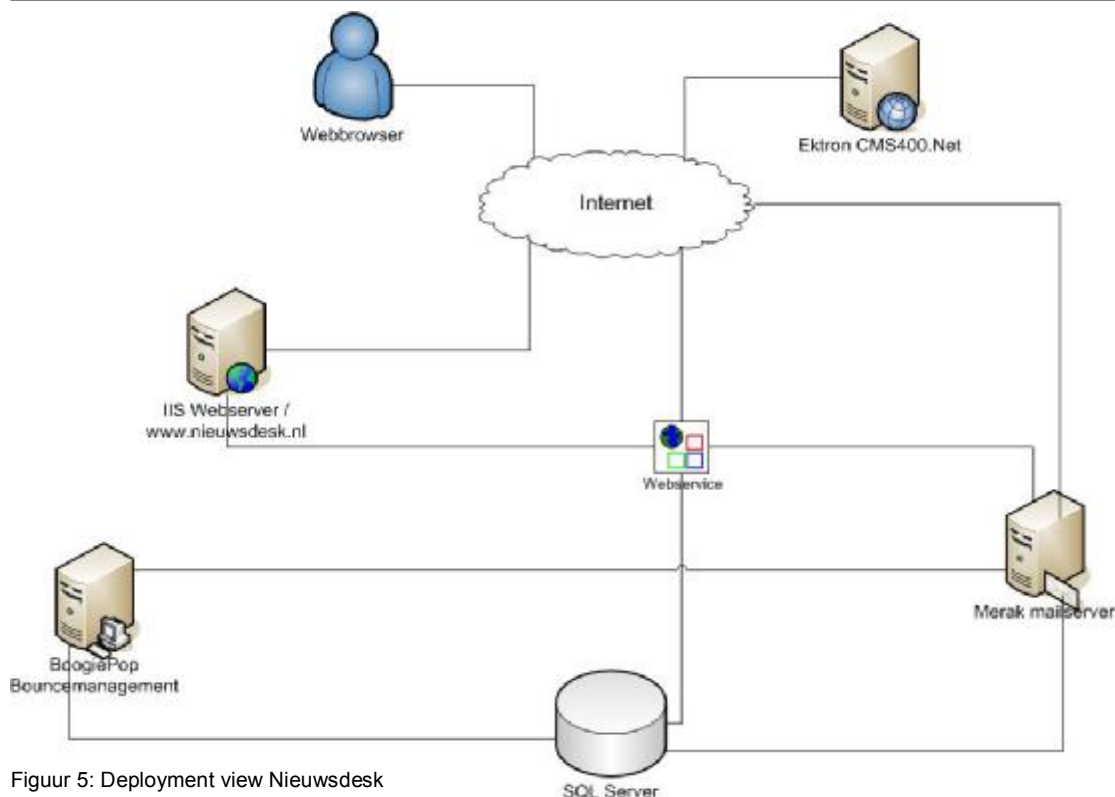
De Implementation view beschrijft de lagenstructuur van Nieuwsdesk. Nieuwsdesk bestaat uit vier verschillende lagen:

- Presentatie, deze laag zorgt voor interactie met de gebruiker. Dit zal gaan via de webbrowser Internet Explorer
- Service, deze laag zal webservices ter beschikking stellen die ervoor zorgen dat functionaliteiten van Nieuwsdesk aanspreekbaar zijn voor aanvragen binnen het systeem maar ook voor aanvragen van buiten het systeem
- Business, deze laag voorziet in de juiste afhandeling van de werking van het systeem
- Data, dit is de laag die voorziet in datacommunicatie

De Use Case view beschrijft alle use cases die relevant zijn voor het bepalen van een architectuur.

Een use case is relevant wanneer deze betrekking heeft op de niet-functionele systeemeisen.

De Deployment view beschrijft hoe Nieuwsdesk werkt op een netwerk. Onderstaand figuur komt uit het Software Architecture Document en geeft de Deployment view weer.

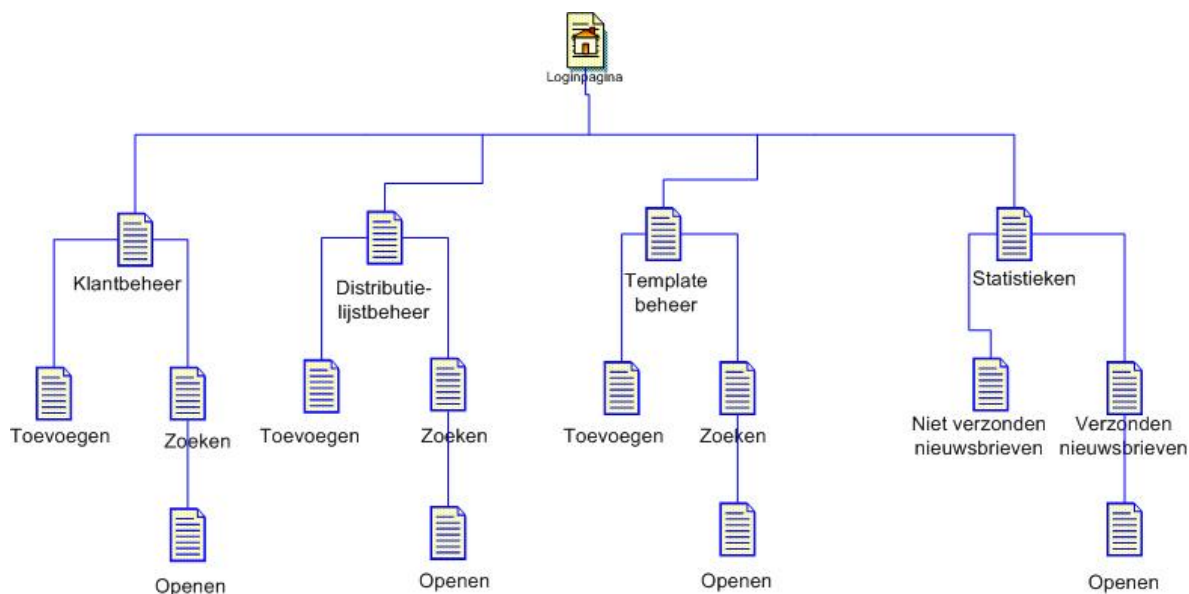


Figuur 5: Deployment view Nieuwsdesk

4.2.5. Navigation Map opstellen

De Navigation Map beschrijft de algemene navigatiestructuur van Nieuwsdesk. Voor Nieuwsdesk zijn er twee verschillende soorten Navigation Maps opgesteld. Eén voor het beheerdersgedeelte en de tweede voor het gedeelte voor klanten en salesmedewerkers.

De indeling van beide Navigation Maps zijn gemaakt op basis van het Use Case Model en de Use Case Specification. Het Use Case Model geeft inzicht in de verschillende soorten hoofdpagina's (Adresboek, Distributielijstbeheer, Artikelbeheer enz.) en de Use Case Specification geeft inzicht in de subpagina's. De Navigation Map voor het beheerdersgedeelte wordt hieronder weergegeven.



Figuur 6: Navigation Map beheerdersgedeelte

Een beheerder heeft vier hoofdpagina's waarmee Nieuwsdesk beheerd kan worden. Elk gedeelte heeft dezelfde indeling. Het statistiekengedeelte is afwijkend van de overige gedeeltes, dit komt doordat de beheerder geen nieuwsbrieven kan toevoegen en zijn de statistieken van nieuwsbrieven voor de beheerder niet nodig.

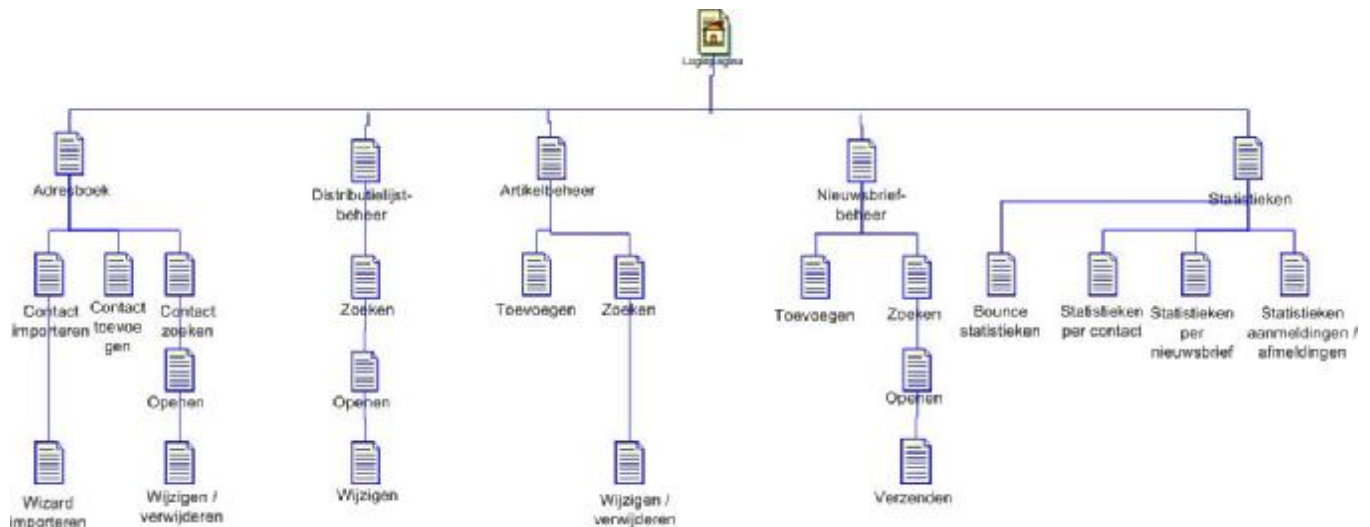
Het klantgedeelte is op dezelfde manier opgebouwd als het beheerdergedeelte om zo de consistentie van de GUI van Nieuwsdesk te bewaren. Het klantgedeelte wijkt wel af van het beheerdergedeelte. Zo heeft het klantgedeelte vijf hoofdpagina's. Een klant kan artikelen en nieuwsbrieven aanmaken, een beheerder kan dit niet. Een beheerder kan templates aanmaken, een klant kan dit niet.

Op de zoekpagina's (in beide gedeeltes) wordt bij het openen alle informatie getoond van die functionaliteit. Wanneer een klant de pagina Adresboek opent, worden alle contacten van de klant getoond, dit geldt ook voor Distributielijstbeheer, Artikelbeheer en Nieuwsbriefbeheer.

Er is voor deze manier gekozen, omdat dit aansluit op andere producten van On-Site BV.

Een andere reden om alle informatie te tonen van een functionaliteit is dat een klant zo gelijk kan zien waar hij/zij zich bevindt in Nieuwsdesk, dit bevordert dus de gebruiksvriendelijkheid.

De Navigation Map voor het klantgedeelte wordt weergegeven in onderstaand figuur.



Figuur 7: Navigation Map klantgedeelte

4.2.6. Data Model opstellen

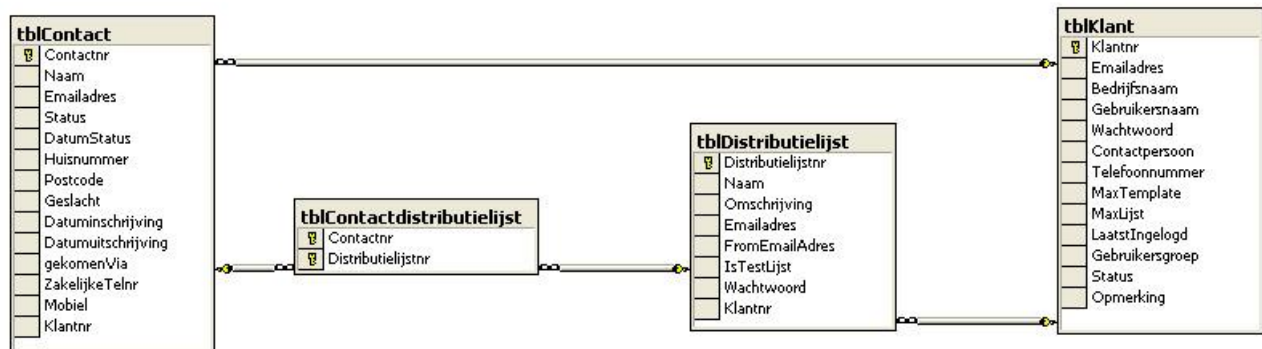
Het Data Model geeft schematisch weer wat de relaties en associaties zijn in de database van Nieuwsdesk.

Voor het ontwerpen van het Data Model is gebruik gemaakt van het Class Model. In het Class Model zijn alle attributen en associaties van de Classes bepaald. De Classes zijn in de database als relaties opgenomen en de associaties zijn doorgevoerd op de relaties.

Voor het ontwerpen van het Data Model is gebruikt gemaakt van de techniek Normaliseren. Deze techniek is gebruikt om redundantie en inconsistentie tegen te gaan. Normaliseren wordt toegepast bij het ontwerpen van relationele databases. De database van Nieuwsdesk is een relationele database. In een relationele database kunnen de tupels uit verschillende relaties met elkaar verbonden worden door het gebruik van vreemde sleutels.

In het Data Model is ook een implementatie van het gemodelleerde Data Model opgenomen. In deze implementatie wordt er per relatie aangegeven wat de primaire en vreemde sleutels (met daarbij de constraints) zijn. Dit implementatiemodel is gemaakt in overleg met de opdrachtgever. Het overleggen met de opdrachtgever is gedaan voor de lengtes van attributen.

In dit verslag is eerder een stuk van het Class Model getoond. In dit Class Model zijn drie Classes opgenomen; Klant, Contact en Distributielijst. Om het verband tussen het Class Model en het Data Model aan te tonen, zal ik de overeenkomstige relaties uit het Data Model tonen.



Figuur 8: Deel van het Data Model Nieuwsdesk database

De Classes Contact, Klant en Distributielijst uit het Class Model zijn in het Data Model opgenomen als relaties. De veel-op-veel associatie tussen Contact en Distributielijst heeft geresulteerd in een zogeheten tussentabel. In deze tussentabel zijn de primaire sleutels uit beide relaties opgenomen in de relatie ContactDistributielijst.

De één-op-veel associaties tussen Klant en Distributielijst en tussen Klant en Contact resulteren in het opnemen van de primaire sleutel van de één kant bij de veel kant (Tussen Klant en Distributielijst is dit dus klantnr uit Klant. Voor Klant en Contact is dit ook klantnr uit Klant).

Het Data Model bestaat in totaal uit dertien relaties. In dit verslag zijn er vier relaties opgenomen. Voor het gehele Data Model verwijs ik u naar het Data Model in de externe bijlage 'Rapport Elaboration Fase'.

4.3. Construction fase van ontwikkeling Nieuwsdesk

Het doel van de Construction fase is het ontwikkelen van de Beta Release van Nieuwsdesk. Naast de ontwikkeling van de Beta Release wordt er in deze fase een testplan opgesteld. Het testen aan de hand van het testplan is uitgevoerd in de Transition fase.

Binnen deze fase hebben er een aantal iteraties plaatsgevonden. Deze fase is doorlopen in 4 iteraties:

1. Lay-out beheerdersgedeelte
2. Code beheerdersgedeelte
3. Lay-out klantgedeelte
4. Code klantgedeelte

Er is voor deze manier van ontwikkelen gekozen, omdat het de ontwikkeling overzichtelijk maakt. Daarnaast worden de oplossingen van problemen die naar voren komen in een eerdere iteratie gelijk meegenomen naar de volgende iteratie.

4.3.1. Beta Release

De Beta Release bevat de technische realisatie van de opgestelde ontwerpen uit de Inception en Elaboration fase.

De Beta release zal worden getest op kwaliteit aan de hand van het testplan. Het testen zal plaatsvinden in de Transition fase.

Om de Beta Release wat nader toe te lichten worden er een aantal onderdelen toegelicht. Deze onderdelen worden toegelicht, omdat deze kenmerkend voor deze opdracht zijn of, omdat deze onderdelen bijgedragen hebben aan de diepgang en de omvang van het project. In de externe bijlage 'Rapport Construction Fase' is het document 'Functionaliteiten' opgenomen, in dit document zijn meer functionaliteiten beschreven die aan de diepgang en de omvang van het project hebben bijgedragen.

4.3.1.1. Master Page gebruiken

Een Master Page is een bestand dat het op een makkelijke manier mogelijk maakt om functionaliteiten op een centrale plaats binnen een applicatie, de master page, op te slaan en beschikbaar te maken voor andere pagina's. Naast de functionaliteiten zorgt een Master Page ook voor visuele overerving op de pagina's binnen een applicatie.

Bij aanvang van dit project had On-Site een grafisch ontwerp gemaakt voor Nieuwsdesk. Het grafisch ontwerp is niet als één geheel aangeleverd, de aanlevering bestond uit verschillende afbeeldingen. De afbeeldingen zijn van het bestandstype JPG en GIF.

Met de afbeeldingen moest de lay-out van Nieuwsdesk samengesteld worden. Het samenstellen van de lay-out kon op twee manieren gedaan worden:

1. In elke pagina code opnemen die de lay-out van de pagina bepaalt
2. Het gebruik van een Master Page

Met oplossing 1 kan de lay-out per pagina verschillen maar dient elke pagina bij een wijziging nagelopen te worden. Dit is inefficiënt. Tevens dient de lay-out van Nieuwsdesk op elke pagina

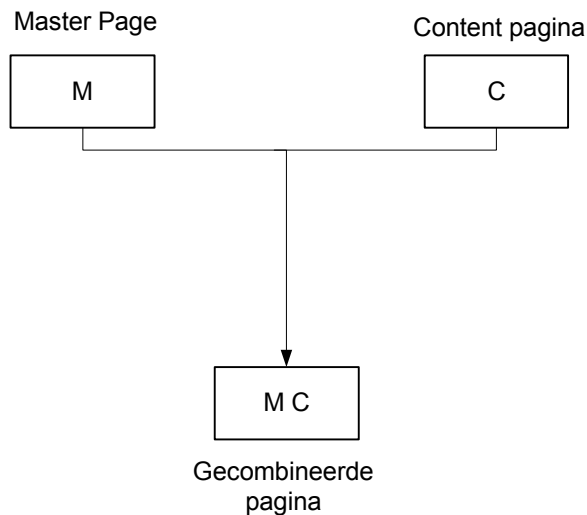
hetzelfde te zijn. Deze oplossing zou dus niet bijdragen aan de aanpasbaarheid en onderhoudbaarheid van Nieuwsdesk.

Met oplossing 2 wordt de lay-out op een centrale plaats opgeslagen. Hierdoor kan de lay-out van de Master Page door andere pagina's binnen Nieuwsdesk hergebruikt worden.

Wanneer er een wijziging in de lay-out plaats moet vinden kan dit bij het gebruik van een Master Page, op één plaats doorgevoerd te worden.

Op basis van aanpasbaarheid en onderhoudbaarheid is er de keuze gemaakt voor het gebruik van een Master Page.

Figuur 13 geeft de werking van een Master Page weer.



Figuur 9: Master Page werking

Alle pagina's binnen Nieuwsdesk gebruiken de Master Page als lay-out pagina. Bij het creëren van een nieuwe pagina in Visual Studio 2005 kan ervoor gekozen worden om een Master Page toe te kennen aan de nieuwe pagina.

Nadat de nieuwe pagina aangemaakt is met een Master Page, ontstaat een gecombineerde pagina.

Naast het gebruik van de Master Page is ervoor gekozen om de opmaak van de pagina's van Nieuwsdesk centraal te regelen via een Cascading Style Sheet. Een Cascading Style Sheet is een techniek waarmee de vormgeving van webpagina's vanuit één plaats beheerd kan worden. Het gebruik van een Cascading Style Sheet draagt dus bij aan de onderhoudbaarheid en aanpasbaarheid van Nieuwsdesk, en daarom is voor het gebruik van een Cascading Style Sheet gekozen.

Een nadeel aan het gebruik van een Cascading Style Sheet is dat elke pagina binnen Nieuwsdesk een verwijzing moet hebben naar de Cascading Style Sheet, maar door het gebruik van een Master Page is dit niet meer nodig. Er is in de Master Page een verwijzing naar de Cascading Style Sheet opgenomen en zoals eerder aangegeven zorgt een Master Page voor visuele overerving (inheritance) op de pagina's die een Master Page hebben.

4.3.1.2. Web service

Met de komst van het .NET Framework 1.0 werd het gebruik van een XML web service aangemoedigd door Microsoft. Een web service is een herbruikbare softwarecomponent die (platform onafhankelijk) gebruikt kan worden door verschillende soorten cliënten. Doordat een web service met heterogene systemen kan communiceren op een gestandaardiseerde manier, is deze technologie snel geaccepteerd voor datacommunicatie over het web.

Een web service bestaat uit één of meerdere methoden die een bepaalde functionaliteit kunnen uitvoeren. Een web service communiceert door middel van XML (eXtensible Markup Language) berichten via SOAP (Simple Object Access Protocol) over HTTP. Het voordeel van SOAP berichten is dat ze doorgelaten worden door een firewall op de HTTP port.

Een eis van de opdrachtgever voor dit project is dat Nieuwsdesk beschikbaar dient te zijn voor het Ektron CMS400.NET. Aan het begin van dit project is er een onderzoek gedaan naar de mogelijkheden voor het gebruik van web services door het Ektron CMS400.NET. Aangezien het Ektron CMS400.NET volledig gebaseerd is op het .NET Framework, is berichtuitwisseling door web services mogelijk.

In de web service van Nieuwsdesk zijn alle functionaliteiten opgenomen die door een klant uit te voeren zijn. Naast deze functionaliteiten zijn ook de aanmeldfunctie, opt-out en opt-in functionaliteiten opgenomen.

Het beveiligen van een web service vond ik minder gemakkelijk dan het aanmaken en gebruiken ervan. Nieuwsdesk is een webapplicatie die de authentication mode "Forms" toepast. Hierdoor is het beveiligen van een web service niet via de Web.config (dit is het configuratiebestand van Nieuwsdesk) te regelen. Wanneer er voor Nieuwsdesk de authentication mode "Windows" toegepast had kunnen worden, zou het beveiligen van een web service geen probleem zijn geweest.

De web service van Nieuwsdesk moet niet voor iemand die zich niet heeft ingelogd, te gebruiken zijn. Dit probleem is opgelost door bij elke method, waar nodig, eerst te controleren of degene die de method wil aanspreken ook aangemeld is.

4.3.1.3. Statistieken bijhouden

Een eis van de opdrachtgever was het bijhouden van statistieken. Na het analyseren van de genoemde eisen in een interview, zijn de eisen in een categorie geplaatst. Een categorie wordt in Nieuwsdesk gebruikt, om zo de statistiekpagina overzichtelijk te houden.

De volgende vier categorieën met bijbehorende eisen zijn opgesteld:

Statistieken per contact:

- § Welke artikelen een contact gelezen heeft
- § Hoe vaak een contact een artikel gelezen heeft

Statistieken per nieuwsbrief:

- § Uit welke artikelen een nieuwsbrief bestaat
- § Hoe vaak een artikel gelezen is
- § Aan welke distributielijsten een nieuwsbrief verstuurd is

- § Wanneer een nieuwsbrief verstuurd is
- § Aan hoeveel contacten een nieuwsbrief per mailing verstuurd is

Aan- en Afmeldingen statistieken:

- § Welke aanmeldingen er zijn geweest in de afgelopen 30 dagen
- § Welke afmeldingen er zijn geweest in de afgelopen 30 dagen

Bounce statistieken:

- § Welke contacten een bepaalde soort bounce status hebben
- § Sinds wanneer heeft een contact een bepaalde soort bounce status

De statistieken van contacten en artikelen worden bijgehouden door de URL's in een nieuwsbrief.

Wanneer een contact een URL aanklikt wordt er een pagina van Nieuwsdesk geopend. Deze pagina schrijft een contactnummer en een guid weg en verwijst door naar het originele URL.

Het contactnummer en guid worden opgeslagen in de relatie tblStatistiek. De statistieken die worden ingelezen via Nieuwsdesk worden niet uit de tblStatistiek uitgelezen. Deze worden uit de relaties tblContactArtikel en tblUrl uitgelezen. De statistieken in de relatie tblStatistiek worden door middel van een job op de SQL Server weggeschreven naar de relaties tblUrl en tblContactArtikel. De statistieken die getoond worden in Nieuwsdesk, lopen dus een dag achter. De keuze hiervoor is gemaakt op basis van performance.

Wanneer de statistieken gelijk weggeschreven zouden worden in de relaties tblURL en tblContactArtikel zou dit een negatief effect hebben op de performance. Nieuwsbrieven worden door veel klanten wekelijks gemaakt en verstuurd, dit heeft als gevolg dat contacten de hele week door artikelen kunnen bekijken.

Wanneer een contact een artikel bekijkt moet er eerst een leesactie en daarna een schrijfactie worden gedaan. Dit is niet bevorderend voor de performance, daarnaast zouden concurrency-problemen op kunnen treden en dit heeft mogelijk dataverlies als gevolg. Daarom is er gekozen om een aparte relatie op te nemen in de database waar alleen maar schrijfacties in worden gedaan. De leesacties worden gedaan op de relaties tblContactArtikel en tblURL.

De data in de relaties tblContactArtikel en tblURL worden bij het starten van de job geupdate. Deze job update eerst beide relaties en leegt vervolgens de relatie tblStatistiek.

4.3.1.4. Importfunctionaliteit ontwikkelen

Voor een gebruiker is er een importfunctionaliteit ontwikkeld om contacten uit Excel en CSV

bestanden te importeren. Deze functionaliteit is gevolgd van een functionele eis van Nieuwsdesk.

Voor het importeren van contacten in Nieuwsdesk is een eigen wizard ontworpen. De wizard bestaat uit drie stappen:

1. Het uploaden van een bestand
2. Het toewijzen van kolommen uit het geuploade bestand, aan waarden die gebruikt worden in Nieuwsdesk

3. Het importeren van de contacten en weergeven van mislukte imports

Er is voor een wizard gekozen, omdat een gebruiker hiermee zo weinig mogelijk inspanning moet doen om zijn doel te bereiken.

Voor het inlezen van Excel bestanden is gebruik gemaakt van het open source programma NExcel. NExcel is een programma dat Microsoft Excel bestanden in kan lezen zonder Microsoft Excel op te starten. Er was de mogelijkheid om een reference naar Microsoft Excel in Nieuwdesk toe te voegen, maar dit kost meer performance. Daarnaast is er een probleem met de versies van Microsoft Excel, een bepaalde versie ondersteunt geen eerdere versies. Bij NExcel worden alle versies ondersteund vanaf Excel 97.

In stap 1 wordt een keuze gemaakt voor het te importeren bestand. Dit bestand kan een CSV of XLS bestand zijn.

Figuur 10 geeft stap 1 van het importeren weer.

home | Adresboek | Distributielijsten | Artikelbeheer | Nieuwsbriefbeheer | Statistieken

Aanmelden
Gebruikersnaam:
Wachtwoord:
U bent aangemeld als tony

Selecteer een bestand om te uploaden en klik op 'Upload'.

FirstName	MiddleName	LastName	Company	Email	BusinessStreet	Zakelijknr	Mobielnu
Test		test	test	email@emial.com	test	0101234567	06123456789
Geen	Marques	Email	On-Site	h	beurs	0107896543	06987654321
Tony		Kanthers	On-Site	email@emial.com	beurs		06123789456
Tony		Kanthers	On-Site	email@emial.com	beurs		06321654789
Michael	Marques	de Almeida	On-Site	email@emial.com	beurs		
Geen	Marques	Email	On-Site	h	beurs	0109876547	

Wanneer u het juiste bestand geupload heeft, kunt op 'Ja' klikken.

Wanneer de inhoud overeenkomt met de inhoud in het geupload bestand, kunt u op de 'Ja' knop klikken.

Figuur 10: Stap 1: contacten importeren uitgevoerd

De gebruiker dient na uploaden op 'Ja' te klikken om verder te gaan met het importeren. Na het uploaden van een bestand, worden de eerste acht regels uit het bestand getoond. Op basis van de eerste acht regels moet een gebruiker de keuze kunnen maken of het juiste bestand is geupload. Wanneer dit niet het geval is, kan de gebruiker een ander bestand uploaden.

Nieuwsdesk

home

Adresboek Distributielijsten Artikelbeheer Nieuwsbriefbeheer Statistieken

Aanmelden
 Gebruikersnaam:

 Wachtwoord:

 U bent aangemeld als tony

Op deze pagina kunt u de kolommen van uw geuploade bestand toekennen aan bepaalde waarden. Op dit scherm is een dropdownlist te zien met daarachter de knop 'Toewijzen'. Daaronder is een kolom uit uw bestand te vinden. Door middel van de onderste knoppen kunt u door uw bestand navigeren. Wanneer u een kolom wilt opnemen in Nieuwsdesk kunt deze toewijzen aan een waarde uit de dropdownlist. Als u een waarde wilt wijzigen kunt u op klikken op 'Wijzig'. De betreffende waarde zal weer beschikbaar zijn in de dropdownlist.

Achternaam

MiddleName
Marques
Marques
Marques

Voornaam:
Achternaam:
E-mail:
Huisnummer:
Postcode:
Geslacht:
Zakelijke Tel:
Mobiel:

Figuur 11: Stap 2: Toewijzen van kolommen uit het geuploade bestand, aan waarden die gebruikt worden in Nieuwsdesk

In stap 2 van de wizard wordt het geuploade bestand per kolom doorlopen. Een gebruiker kan hier een kolom toewijzen aan een waarde uit de dropdownlist.

Er kan door het Excel bestand genavigeerd worden door middel van de knoppen Terug en Volgende. De knop Klaar wordt beschikbaar als alle waarden (behalve voornaam of achternaam) gevuld zijn en/of als een voornaam of achternaam en emailadres zijn toegewezen.

In stap 3 van de wizard worden alle contacten uit het Excel bestand getoond met als kolomkoppen de toegewezen waarden. Onder dit overzicht worden alle distributielijsten getoond, een selectievak van een distributielijst kan worden aangevinkt. Vervolgens kan een gebruiker op 'Ja' klikken en de contacten worden geïmporteerd met daarbij de koppeling(en) voor de gekozen distributielijst(en). Wanneer een gebruiker op 'Nee' klikt wordt er teruggegaan naar de upload pagina.

Na het aanklikken van de 'Ja' knop worden alle contacten toegevoegd. Er wordt per rij gecontroleerd of het emailadres een geldig format heeft. Dit gebeurt aan de hand van een regular expression. Alle contacten/rijen die niet kunnen worden toegevoegd, worden aan het einde van het importeren getoond in een GridView. Een klant kan met deze informatie de contacten aanpassen.

Figuur 12 geeft stap 3 van de wizard weer.

Nieuwsdesk

home
Adresboek
Distributielijsten
Artikelbeheer
Nieuwsbriefbeheer
Statistieken

Aanmelden

Gebruikersnaam:

Wachtwoord:

U bent aangemeld als tony

Hieronder ziet u de zojuist gekozen bestandsindeling. Als het voorbeeld klopt met de kolomnamen, kunt u op 'Ja' klikken. Tevens kunt u ervoor kiezen om alle contacten aan een distributielijst te koppelen. Dit kunt u doen door boven de 'Ja' knop één of meerdere distributielijsten aan te vinken.

Naam	Email	Huisnummer	Postcode	Geslacht	Zakelijke Telnr	Mobiel
Test test	email@emial.com				0101234567	06123456789
Geen Email	h				0107896543	06987654321
Tony Kanters	email@emial.com					06123789456
Tony Kanters	email@emial.com					
Michael de Almeida	email@emial.com					
Geen Email	h				0109876547	

☒ Corporate
 ☐ Test

Dit zijn de contacten die niet zijn toegevoegd. Dit komt doordat er een verkeerd emailadres of geen naam was opgegeven. Druk op de knop 'Klaar' om terug te gaan naar het adresboek.

Naam	Email	Huisnummer	Postcode	Geslacht	Zakelijke Telnr	Mobiel
Geen Email	h				0107896543	06987654321
Geen Email	h				0109876547	

Figuur 12: Stap 3: Het importeren van de contacten en weergeven van mislukte imports

4.3.2. Testplan opstellen

Na de ontwikkeling van Nieuwsdesk is het testplan opgesteld. De basis voor het testplan gemaakt met de softwaretestaanpak 'TMAP'. Deze testaanpak is gebruikt, omdat het een testaanpak is voor het gestructureerd testen van software.

Om Nieuwsdesk te testen op kwaliteit zijn er drie testtechnieken en twee checklists gebruikt:

- Syntactische test

De syntactische test heeft als doel fouten te ontdekken in de lay-out van schermen en overzichten. Daarnaast wordt er bij deze techniek ook getest op primaire invoercontroles van invoervelden. Deze testtechniek is gebruikt om consistentie en gebruiksvriendelijkheid van Nieuwsdesk te testen.

- Semantische test

De semantische test heeft als doel het vaststellen van kwaliteit van de consistentie tussen de ingevoerde gegevens en/of de al in het systeem aanwezige informatie.

- Dataflow test

De dataflow test heeft als doel de gegevensstroom en de verwerking van een applicatie te testen op kwaliteit. Deze testtechniek is gebruikt om te testen of de ontwikkelde functionaliteiten volgens de ontwerpen zijn ontwikkeld.

- Checklist gebruiksvriendelijkheid

Er is gebruik gemaakt van een checklist gebruiksvriendelijkheid. Deze checklist is gebruikt, omdat in deze checklist punten zijn opgenomen die afkomstig zijn van de website van Jakob Nielsen. Jakob Nielsen is een specialist op het gebied van usability.

- Checklist beveiliging

Er is gebruik gemaakt van een checklist beveiliging. In deze checklist zijn een aantal bekende issues (zoals SQL Injections) opgenomen. Door hierop te testen wordt duidelijk hoe veilig Nieuwsdesk is.

De testsets zijn opgesteld in overleg met de opdrachtgever. Met het percentage van de functionaliteit was ik het in het begin gelijk eens. Over de performance, gebruiksvriendelijkheid en beveiliging is er gediscussieerd. Aanvankelijk wilde de opdrachtgever 10% voor de beveiliging reserveren, 15% voor de performance en 30% voor de gebruiksvriendelijkheid.

Ik heb hem overtuigd om 15% minder belang te hechten aan de gebruiksvriendelijkheid en 10% meer aan beveiliging en 5% aan performance.

Ik heb dit voorgesteld, omdat Nieuwsdesk wel een zekere mate van beveiliging nodig heeft aangezien er allerlei emailadressen van contacten in zijn opgenomen. De 5% voor de performance vond ik nodig, omdat de functionaliteiten van Nieuwsdesk zware impact op performance kunnen hebben.

Kwaliteitsattributen	Relatief belang
Beveiliging	20
Functionaliteit	45
Gebruiksvriendelijkheid	15
Performance	20
Totaal in %	100

Tabel 6: Kwaliteitsattributen Nieuwsdesk

4.4. Transition fase van ontwikkeling Nieuwsdesk

In de Transition fase wordt Nieuwsdesk getest aan de hand van het opgestelde testplan uit de Construction fase.

Na het testen van Nieuwsdesk is er een herstelperiode ingepland. In deze herstelperiode zijn alle gevonden fouten verbeterd.

Deze fase is in één iteratie doorlopen. Er had gekozen kunnen worden om na het testen van elk deelsysteem een herstelperiode voor dat deelsysteem in te plannen. Er is niet voor deze manier gekozen, omdat ik de gevonden fouten niet van zo'n grote omvang vond om hier aparte herstelfasen voor in te plannen.

4.4.1. Testen

Het testen is uitgevoerd om te bepalen wat de kwaliteit is van het ontwikkelde Nieuwsdesk.

De resultaten van het testen zijn opgenomen in het testrapport. Het testrapport beschrijft alle resultaten van alle toegepaste testtechnieken en checklists. Naast een beschrijving bevat het testrapport ook een conclusie.

De conclusie van het testen van Nieuwsdesk is redelijk positief. De belangrijkste punten uit de conclusie zijn:

- In totaal zijn er 168 testgevallen gedefinieerd in het testplan
- Van de 168 testgevallen hebben er 42 een ander resultaat dan de verwachte uitvoer. Dit is 25% van alle testgevallen
- Alle 42 foutieve testgevallen komen voort uit de syntactische test, alle 42 testgevallen hebben betrekking op primaire invoercontroles
- De herstelperiode is kort, omdat het herstellen van de foutieve testgevallen weinig tijd in beslag neemt
- De gebruiksvriendelijkheid van de applicatie is goed te noemen. De opbouw van de schermen is volgens de testers acceptabel, het taalgebruik wordt als prettig ervaren (ook die van de foutboodschappen) en naamgeving van velden e.d. is voldoende
- De beveiliging is ook goed. Alle vooraf opgestelde SQL Injections testen hadden geen effect.

5. Iteratie 2 van ontwikkeling Nieuwsdesk

Gedurende iteratie 2 zijn er functionaliteiten ontwikkeld waardoor Nieuwsdesk als marketing tool kan fungeren. Deze functionaliteiten zijn (nog) niet beschikbaar voor klanten van On-Site maar voor haar salesmedewerkers en haar dochterbedrijven. De marketingfunctionaliteiten van Nieuwsdesk zijn ontwikkeld als onderdeel van Nieuwsdesk uit de eerste iteratie, omdat deze functionaliteiten niet zouden werken zonder de mogelijkheden van het reeds ontwikkelde Nieuwsdesk. Hierdoor is deze iteratie onderdeel van dit project en wordt deze iteratie niet als apart project beschouwd. Ook is er tijdens deze iteratie gewerkt aan dit afstudeerverslag.

5.1. Inception Fase van ontwikkeling Nieuwsdesk

Niet alle documenten die tijdens iteratie 1 zijn gemaakt, zijn in iteratie 2 aangepast. Het Risk Assessment is niet aangepast, omdat er geen andere risico's zijn bijgekomen voor het project die de voortgang van het project in gevaar kunnen brengen.

Het Vision Document (met daarin het Risk Assessment) is voor iteratie 2 herzien in samenwerking met de opdrachtgever. In het Vision Document zijn er nieuwe functionele systeemeisen opgenomen. Naast nieuwe functionele systeemeisen zijn er nieuwe gebruikers gedefinieerd.

5.1.1. Software Development Plan aanpassen

Voor de tweede iteratie is de planning uit het Software Development Plan aangepast. De planning is zonder de opdrachtgever opgesteld, omdat het einde van de afstudperiode, en hiermee ook iteratie 2, op 23 maart 2007 is. In het Software Development Plan uit de eerste iteratie is een planning voor het gehele project opgenomen. Doordat de eerste iteratie een week eerder dan gepland is afgerond en de eisen en wensen voor de tweede iteratie van het ontwikkelen van Nieuwsdesk in de Inception Fase van iteratie 2 gedetailleerd zijn beschreven, is de planning voor het resterende gedeelte van het project aangepast.

In de opgestelde planning uit de eerste iteratie, was het einde van de Transition Fase van iteratie 1 gepland voor 15 januari 2007. Het daadwerkelijke einde van deze fase, en daarmee ook iteratie 1, was op 9 januari 2007.

Afgezien van de planning uit het Software Development Plan uit de eerste iteratie is verder niets veranderd in het Software Development Plan. De organisatiestructuur is onveranderd gebleven voor de tweede iteratie.

5.1.2. Use Case Model bijwerken

Voor de tweede iteratie van Nieuwsdesk is er een aantal nieuwe use cases bijgekomen. Aan deze use cases zijn ook classificaties gegeven door middel van de methode MoSCoW, zoals dit gedaan is tijdens de eerste iteratie. De actoren in het Use Case Model voor iteratie 2 verschillen wel van de opgestelde actoren tijdens iteratie 1.

Deze use cases en actoren zijn in overleg met de opdrachtgever gemaakt. Ook de classificaties zijn in overleg gemaakt met de opdrachtgever, zoals dit gedaan is tijdens iteratie 1.

Onderstaande tabel geeft de actoren voor iteratie 2 weer. Er zijn twee actoren, CMS en Contact, niet gebruikt voor het ontwikkelen van Nieuwsdesk tijdens iteratie 2. De actor CMS is niet gebruikt, omdat het Ektron CMS400.NET geen gebruik zal maken van de marketingfunctionaliteiten en de actor Contact zal ook geen interactie hebben met de marketingfunctionaliteiten van Nieuwsdesk.

Code	Actor	Omschrijving	Gewicht
A1	Salesmedewerker	De eindgebruiker van Nieuwsdesk	4
A3	Applicatiebeheerder	Beheerder van de Nieuwsdesk	3

Tabel 7: Actoren overzicht Nieuwsdesk iteratie 2

Voor de ontwikkeling van iteratie 2 zijn er in totaal vier nieuwe use cases bijgekomen. Onderstaande tabel geeft de use cases uit iteratie 2 weer.

Code	Use Case Naam	Omschrijving	Gewicht	Samenhang	Classificatie	Prioriteit
UC54	Ad hoc distributielijst maken	Ad hoc een distributielijst aanmaken	4		M	8
UC55	Contact(en) koppelen aan een ad hoc distributielijst	Het koppelen van een contact aan een ad hoc aangemaakte distributielijst	4		M	8
UC56	Informatie ad hoc distributielijst weergeven	Het weergeven van statistieken van een ad hoc aangemaakte distributielijst	4		M	8
UC57	Statistieken opt-out weergeven	Het weergeven van alle afmeldingen via de opt-out functionaliteit	4		M	8

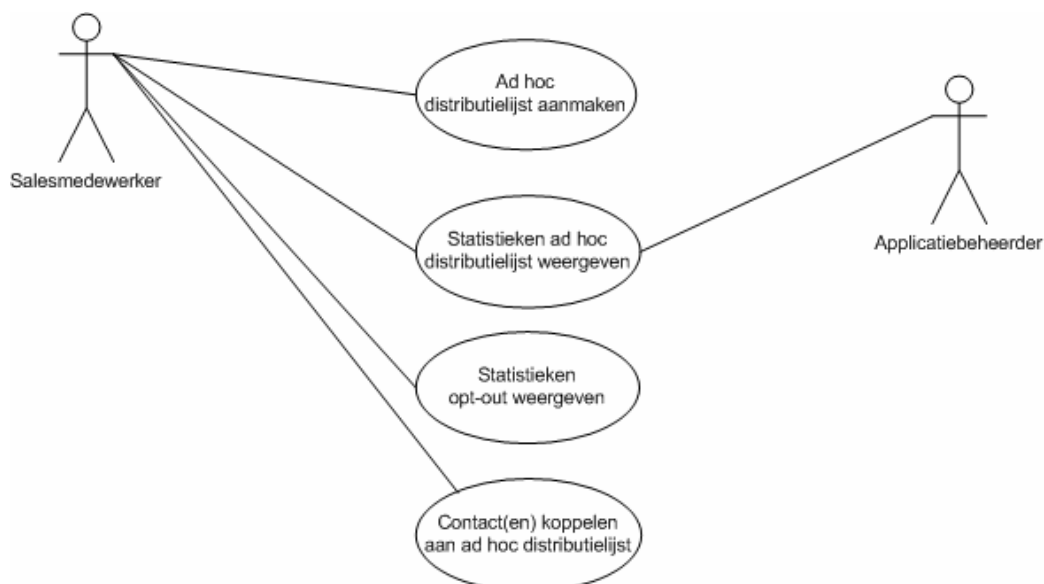
Tabel 8: Use Case Model iteratie 2

In de Inception Fase van iteratie 2 is er ook een use case diagram gemaakt van de nieuwe use cases.

De actor Salesmedewerker kan interactie hebben met alle vier de use cases en de actor

Applicatiebeheerder kan met één use case interactie hebben.

Figuur 13 geeft het use case diagram uit iteratie 2 weer.

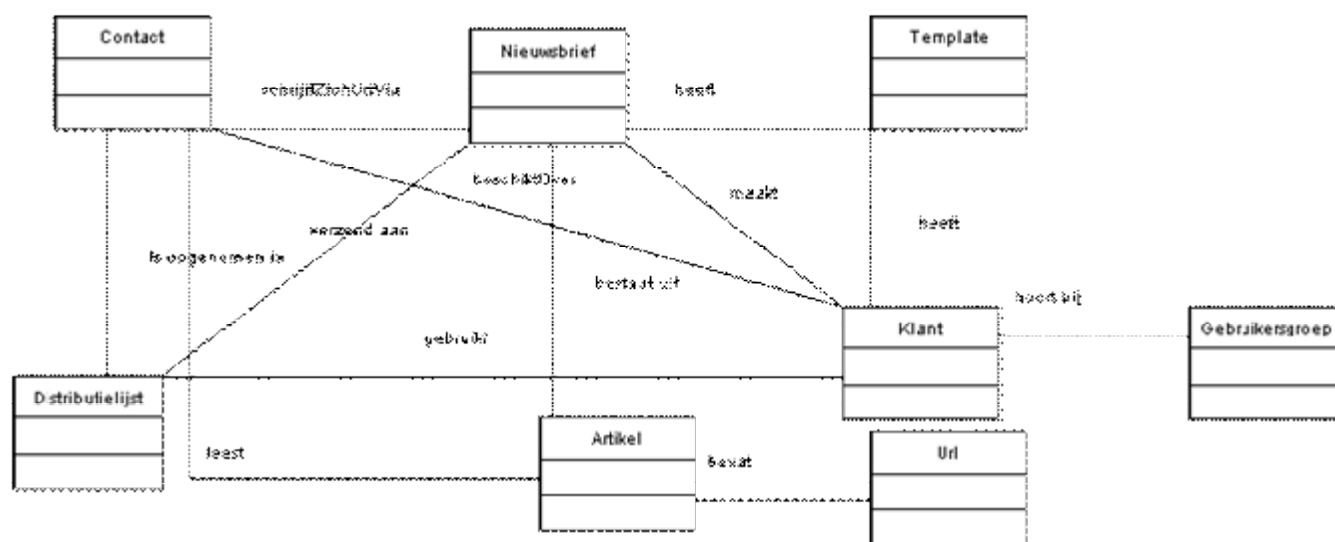


Figuur 13: Use case diagram iteratie 2

5.1.3. Business Object Model bijwerken

Het Business Object Model is tijdens iteratie 2 aangepast. De nieuwe functionele systeemeis 'Statistieken opt-out weergeven' kan niet worden opgenomen met de huidige situatie. In het Business Object Model uit iteratie 1 bestaat er geen associatie tussen Contact en Nieuwsbrief. Hierdoor is het onmogelijk om te achterhalen welk contact zich heeft uitgeschreven via de opt-out functionaliteit in een bepaalde nieuwsbrief. Voor deze functionaliteit is er een associatie toegevoegd tussen Contact en Nieuwsbrief.

Door deze wijziging komt het Business Object Model er als volgt uit te zien:



Figuur 14: Business Object Model iteratie 2

Dit Business Object Model is na het toevoegen van de associatie tussen Contact en Nieuwsbrief niet meer aangepast gedurende iteratie 2.

5.2. Elaboration fase van ontwikkeling Nieuwsdesk

In de Elaboration fase van iteratie 2 zijn alle documenten aangepast behalve het Software Architecture Document. Dit document had geen aanpassing nodig, omdat de nieuwe use cases geen invloed hebben op de gekozen architectuur van Nieuwsdesk.

Deze fase is in één iteratie doorlopen, dit in tegenstelling tot deze fase in iteratie 1. Er is hiervoor gekozen, omdat de nieuwe functionaliteiten (op één na) bij het klantgedeelte horen. In iteratie 1 waren er namelijk twee duidelijke gedeeltes te onderscheiden.

5.2.1. De use case specification maken

Voor iteratie 2 zijn er vier use cases opgesteld. Alle vier de use cases zijn na het voltooien van initiële versie aan de opdrachtgever voorgelegd en door hem gekeurd. De use case specification is uitgebreid met deze use cases. De nieuwe use cases zijn gemaakt op basis van een interview met de opdrachtgever. In dit interview werd duidelijk wat de opdrachtgever exact wilde met deze functionaliteiten.

5.2.2. Use case realization maken

Op basis van de nieuwe use cases in de use case specification, zijn de use cases aan de use case realization toegevoegd.

De sequentiediagrammen in de use case realization zijn toegevoegd op basis van de nieuwe use cases in de use case specification.

5.2.3. Class Model opstellen

De eis voor de statistieken van de opt-out heeft in het Business Object Model tot een nieuwe associatie geleid tussen contact en nieuwsbrief. In het Class Model heb ik de soort associatie bepaald. Er waren drie mogelijkheden:

1. Eén-op-één
2. Eén-op-veel
3. Veel-op-veel

De eerste optie heeft geen enkele zin, omdat dat dan een tuple uit de relatie contact een corresponderende tuple in de relatie nieuwsbrief zou hebben. De één-op-één associatie tussen twee relaties in een relationele database is een teken van een niet goed genormaliseerd datamodel. De enkele keer dat deze associatie toegepast wordt in een relationele database is om de tekortkomingen van een database te omzeilen door bijvoorbeeld een relatie op te delen om performancewinst te halen.

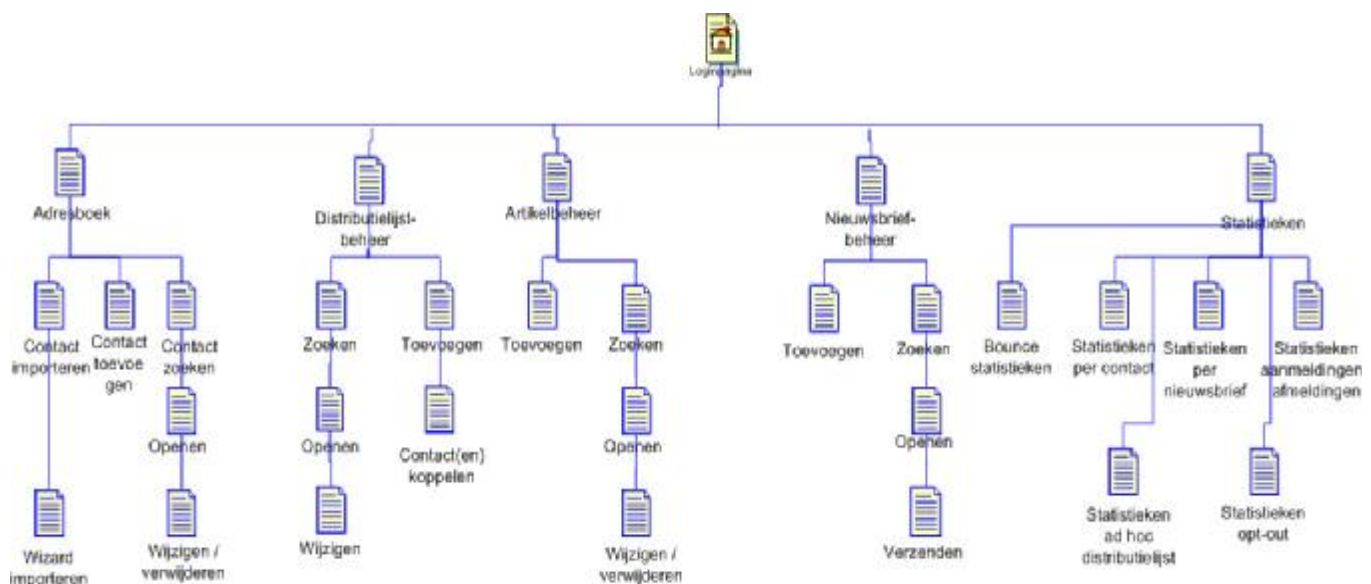
Bij het toepassen van de tweede optie (één-op-veel) zou nieuwsbrief de veel kant en contact de één kant worden. Dit zou betekenen dat een nieuwsbrief nul of meerdere contacten kan hebben die zichzelf uitgeschreven hebben en dat een contact zich via één nieuwsbrief uit zou kunnen schrijven. Bij het toepassen van deze associatie wordt de primaire sleutel uit nieuwsbrief opgenomen als

vreemde sleutel in contact. Met het opnemen van vreemde sleutels in een relationele database wordt de samenhang tussen de gegevens vastgelegd. Vreemde sleutels zijn gebonden aan de referentiële integriteitsregel. Deze regel bepaalt dat er een vreemde sleutel in een relatie nooit een NULL waarde mag bevatten. Bij het toepassen van deze associatie zou het betekenen dat een contact altijd een associatie met een nieuwsbrief zou hebben, dit heeft als resultaat dat elk contact zichzelf uitgeschreven moet hebben.

Bij het toepassen van de derde optie (veel-op-veel) wordt het mogelijk om een contact via meerdere nieuwsbrieven uit te laten schrijven. Voor deze associatie is een koppeltabel nodig tussen contact en nieuwsbrief. Door het gebruik van een koppeltabel zullen er geen vreemde sleutels een NULL waarde hebben, want als het contactnummer en een nieuwsbriefnummer in de koppeltabel voorkomen, betekent dit dat het contact zich heeft uitgeschreven. Als een contactnummer niet voorkomt in de koppeltabel betekent dat het contact zichzelf niet heeft uitgeschreven.

Op basis van bovenstaande beredenering heb ik gekozen voor de derde optie, de veel-op-veel associatie.

5.2.1. Navigation Map opstellen



Figuur 15: Navigation Map klantgedeelte iteratie 2

In overleg met de opdrachtgever is er besloten om de vier nieuwe functionaliteiten op te delen in vijf pagina's. De Statistieken ad hoc distributielijst is namelijk beschikbaar voor zowel het klantgedeelte als het beheerdergedeelte. In het beheerdergedeelte moeten alle ad hoc distributielijsten te zien zijn en in het klantgedeelte alleen de ad hoc distributielijsten van de ingelogde klant.

Deze functionaliteit telt dus twee pagina's, de andere drie functionaliteiten zijn opgenomen in het klantgedeelte. Voor elke functionaliteit is een aparte pagina gemaakt.

In het klantgedeelte zijn onder distributielijstbeheer de pagina's Toevoegen en Contact(en) koppelen toegevoegd. Onder Statistieken zijn de pagina's Statistieken ad hoc distributielijst en Statistieken opt-out toegevoegd.

De nieuwe pagina's onder Statistieken zijn daar toegevoegd, omdat zo alle functionaliteiten voor de Statistieken in één gedeelte te vinden zijn.

De pagina Toevoegen onder Distributielijstbeheer is gemaakt om ad hoc distributielijsten aan te maken. Wanneer een ad hoc distributielijst aan is gemaakt, wordt altijd de pagina contact(en) koppelen getoond. De ad hoc distributielijst kan dus één keer gevuld worden en vervolgens meerdere malen gebruikt worden.

Het beheerdergedeelte heeft, zoals eerder vermeld, er één pagina bij gekregen. Op deze pagina is het mogelijk om de statistieken van ad hoc distributielijsten van klanten te tonen. Deze Navigation Map is te zien in het document Navigation Map, deze is te vinden in het 'Rapport Elaboration Fase' van de externe bijlagen.

5.2.2. Data Model opstellen

Door het toevoegen van de veel-op-veel associatie tussen contact en nieuwsbrief is in het Data Model de relatie ContactNieuwsbrief toegevoegd. In deze relatie zijn de primaire sleutels uit contact en nieuwsbrief opgenomen. Wanneer een contact zichzelf uitschrijft moet er een uitschrijfdatum opgeslagen worden. Voor de uitschrijfdatum had ik drie opties:

1. Uitschrijfdatum opnemen in de koppeltabel
2. De bestaande uitschrijfdatum in contact gebruiken
3. De bestaande uitschrijfdatum in contact gebruiken en een uitschrijfdatum opnemen in de koppeltabel

De eerste optie bood de mogelijkheid om per tupel een uitschrijfdatum te bewaren. Bij het gebruik van deze optie zou de uitschrijfdatum in contact verwijderd moeten worden. Wanneer ik deze optie had doorgevoerd, zou een klant een contact niet meer kunnen uitschrijven, omdat een contact op deze manier alleen zichzelf uit kan schrijven via de opt-out functionaliteit in een nieuwsbrief.

Bij het gebruik van de tweede optie is het mogelijk dat een contact zichzelf uitschrijft voor een nieuwsbrief en een contact kan worden uitgeschreven door een klant. De uitschrijfdatum wordt op één plaats bewaard in de database. Deze optie kan gebruikt worden, omdat een contact ingeschreven óf uitgeschreven is.

De derde optie is niet gebruikt, omdat er dan voor een contact op tweede verschillende plaatsen een uitschrijfdatum bewaard kan worden. Bij het inlezen van een contact moet dan op twee plaatsen gecontroleerd worden of een contact uitgeschreven is. Wanneer een contact door een klant is uitgeschreven en een contact heeft zichzelf uitgeschreven, heeft hij twee verschillende uitschrijfdatum's.

Op basis van bovenstaande beredenering is er voor de tweede optie gekozen.

5.3. Construction fase van ontwikkeling Nieuwsdesk

5.3.1. Beta Release

Het realiseren van de Beta Release voor de tweede iteratie heeft veel minder tijd in beslag genomen dan bij de eerste iteratie. Dit komt omdat er voor de tweede iteratie vier verschillende functionaliteiten zijn ontwikkeld. In de eerste iteratie waren dit er 53.

Een ander groot verschil met de eerste iteratie is dat voor het realiseren van Beta Release er twee weken gepland waren, voor de eerste iteratie waren dit bijna vier weken.

5.3.2. Testplan opstellen

Voor het testplan in de tweede iteratie heb ik dezelfde testtechnieken en checklists als bij de eerste iteratie gebruikt. Dit heb ik gedaan, omdat de functionaliteiten voor iteratie 2 aan dezelfde kwaliteitseisen als die van iteratie 1 moeten voldoen.

De opdrachtgever vond dit geen probleem, omdat hij van mening is dat Nieuwsdesk in de eerste iteratie voldoende getest is.

De kwaliteitsattributen die bepaald zijn in iteratie 1, zijn in overleg met de opdrachtgever gehandhaafd.

5.4. Transition fase van ontwikkeling Nieuwsdesk

Deze fase is in één iteratie doorlopen. Er had gekozen kunnen worden om na het testen van elk deelsysteem een herstelperiode voor dat deelsysteem in te plannen. Er is niet voor deze manier gekozen, omdat voor de tweede iteratie niet alle deelsystemen zijn aangepast.

5.4.1. Testen

Tijdens het testen zijn er geen onverwachte problemen opgetreden. De organisatie rond het testen in de tweede iteratie was hetzelfde als in de eerste iteratie. Ik heb besloten om de organisatie hetzelfde te houden, omdat er tijdens het testen in de eerste iteratie geen onverwachte problemen zijn geweest. Het uitvoeren van de opgestelde testsets in het testplan is uitgevoerd door drie testpersonen, inclusief mijzelf. Dit zijn dezelfde testpersonen als bij de eerste iteratie.

Testpersoon 1: Wouter Veenboer

Testpersoon 2: Tony Kanters

Testpersoon 3: Michael Marques de Almeida

Testpersoon 1 is een salesmedewerker van Terradesk, dit is een dochteronderneming van On-Site BV. Deze testpersoon wordt een eindgebruiker van Nieuwsdesk. Dit was één van de redenen waarom hij aan deze test heeft deelgenomen. Doordat hij met Nieuwsdesk zal gaan werken, kan hij goed beschrijven wat er nog ontbreekt aan/onhandig is binnen Nieuwsdesk.

Testpersoon 2 is de opdrachtnemer.

Testpersoon 3 is een medewerker van de Multimedia afdeling van On-Site BV. Tijdens zijn werk ontwikkeld hij webapplicaties. Momenteel volgt hij de (duale) studie Grafimediotechnologie aan de Hogeschool van Rotterdam.

Testpersonen 1 en 3 hebben de semantische en syntactische test uitgevoerd. Daarnaast is er met beide testpersonen een interview afgenomen waarin de punten van de 'checklist gebruiksvriendelijkheid' behandeld worden. Testpersoon 3 kan door zijn studie heel goed beschrijven waar Nieuwsdesk nog verbeterd zou kunnen worden op het gebied van gebruiksvriendelijkheid. Dit is de reden waarom hij aan de hand van de checklist gebruiksvriendelijkheid heeft getest.

Testpersoon 2 heeft alle opgestelde testsets van verschillende testtechnieken en de checklist beveiliging uit het testplan uitgevoerd. Ik heb zelf alle testsets uitgevoerd, omdat ik twee testpersonen niet genoeg vond om een goed beeld te creëren van de kwaliteit van Nieuwsdesk.

6. Evaluatie ontwikkeling Nieuwsdesk

In dit hoofdstuk behandel ik de evaluatie van het product en mijn proces. Door deze evaluatie wordt duidelijk of ik de juiste keuzes heb gemaakt. Ook zal ik aangeven of ik een volgend project op dezelfde manier zal aanpakken als ik heb gedaan bij dit project.

6.1.1. Productevaluatie

Allereerst wil ik zeggen dat ik zeer trots ben op het feit dat ik Nieuwsdesk binnen de gestelde tijd en met de eisen en wensen van de opdrachtgever heb ontworpen en ontwikkeld. Dit komt doordat de opdrachtgever mij veel vrijheid en vertrouwen heeft gegeven voor de ontwikkeling van Nieuwsdesk. Ik vind het achteraf een verstandige keuze geweest om Nieuwsdesk op basis van het ASP.NET Framework te ontwikkelen. Steeds meer applicaties van On-Site BV worden op basis van het ASP.NET Framework ontwikkeld. Nieuwsdesk zal hierdoor beter onderhoudbaar en beheerbaar zijn voor On-Site BV, omdat het op basis van een bekende technologie gebouwd is.

Ik ben blij dat ik de kans heb gekregen om aan dit project te werken. Dit project heeft niet alleen bijgedragen aan mijn ervaring en kennis van ASP.NET maar mijn kennis op het gebied van Email Marketing en privacy is ook uitgebreid. Ik vind het jammer dat tijdens de opleiding geen aandacht is geschonken aan deze onderwerpen, omdat deze onderwerpen steeds belangrijker worden in de Informatica.

Ondanks dat ik mijn product zeer geslaagd vind, heb ik één punt van kritiek. Ik vind het grafisch design niet uitnodigend. Het is een zakelijk en strak ontwerp, dit had van mij wel wat speelser gemogen. Ik heb een eigen ontwerp aangeleverd maar dit sloot niet voldoende aan op de huisstijl van On-Site BV.

6.1.2. Procesevaluatie

Het proces is volgens mij goed verlopen. Ik heb eerst een oriënterend gesprek gehad met de opdrachtgever over wat hij globaal wilde en hoe het project georganiseerd kon worden. Aan de hand van dit interview heb ik een Vision Document en Software Development Plan geschreven en ben ik een onderzoek gestart naar bestaande nieuwsbriefmodules. Door dit onderzoek wist ik meer van het onderwerp af en ben ik te weten gekomen wat er allemaal bij komt kijken (bijvoorbeeld het onderwerp privacy).

Na de documenten geschreven te hebben en het onderzoek afgesloten te hebben, heb ik een tweetal interviews gehad met de opdrachtgever en de teamleider over wat de applicatie precies moest kunnen. Tijdens dit gesprek heb ik kunnen meedenken en discussiëren door de opgedane kennis uit het onderzoek.

Na deze interviews heb ik een detailplanning opgesteld voor het hele project. Ik heb mij gedurende het gehele project gehouden aan de planning. Na het opstellen van de planning ben ik verder gegaan met de analyse en ontwerp van Nieuwsdesk. De toegepaste technieken en gemaakte documenten hebben allen bijgedragen aan het succes van dit project.

Het gebruik van UP en UML is een goede keuze geweest, omdat ik hierdoor gestructureerd en overzichtelijk heb kunnen werken. UML is namelijk voor mij een vertrouwde techniek, door deze techniek te gebruiken heb ik mij niet uitgebreid hoeven te verdiepen in de toepassing ervan. Tijdens de Construction Fase heb ik mij wel moeten verdiepen in de mogelijkheden van ASP.NET en de toepassing ervan. De opgedane kennis gedurende de opleiding was niet voldoende om dit project tot een succesvol einde te brengen. Ik ben wel heel erg blij met het feit dat mijn kennis nu vele malen uitgebreider is dan bij aanvang van dit project.

Over de manier van testen ben ik niet geheel tevreden. Ik vind dat Nieuwsdesk door minstens vier testers getest had moeten worden. Door de drukte bij On-Site zijn er twee testers toegewezen. Doordat ik hier niet op had gerekend, heb ik zelf ook getest. Ik vind dat ik niet representatief ben voor de eindgebruikers. Ik heb wel de opgestelde testsets uitgevoerd maar ik ben van mening dat iemand die niets met de ontwikkeling van een programma te maken heeft gehad, meer representatief is dan dat een ontwikkelaar is. De uitslag van het testen is mij wel goed bevallen, ik had alleen twee extra testers Nieuwsdesk willen laten testen.

Op het testen na ben ik over het proces tevreden. Ik heb gedurende het gehele project zelfstandig kunnen werken. De opdrachtgever is gedurende het hele project altijd beschikbaar geweest voor vragen en opmerkingen van mij. Hij heeft mij de vrijheid en ruimte gegeven om zelfstandig aan de opdracht te werken. Daarnaast heb ik mee mogen denken over de functionaliteiten van Nieuwsdesk. Over de begeleiding ben ik zeer tevreden, omdat ik graag aan projecten/opdrachten werk waar ik niet alleen maar moet ontwikkelen wat een opdrachtgever wil, maar ook mijn input kan geven aan het project of de opdracht.

Literatuurlijst

Boeken

- Developing Web applications with Microsoft Visual C#.Net (70-315), Jeff Webb, Onderdeel van MCAD/MCSD Self-Paced training kit
- RUP op maat, een praktische handleiding voor IT-projecten, 1e druk, E. Dekker & Collaris, R. A., ISBN 9012114136
- Praktisch UML, 3e editie, J. Warmer & A. Kleppe, ISBN 9043008125
- Testen volgens TMap, 2e druk, M. Pol & R. Teunissen & E. van Veenendaal, ISBN 9072194586
- Juridische aspecten van de informatievoorziening, 3e druk, R.D. van Bruggen & H.A.A. van Dun & E. de Lange, ISBN 9039518874
- Mastering Regular Expressions, USA Edition, Jeffrey Reidl ISBN10 0596002890 ISBN13 9780596002893
- Professional ASP.NET 2.0 - Bill Evjen, Devin, Rader & Srinivasa Sivakumar ISBN10: 0764576100 ISBN13: 9780764576102

Internet

- <http://www.wikipedia.org/>
- <http://www.google.nl/>
- <http://www.devtips.net/>
- <http://msdn.microsoft.com/>
- http://www.justitie.nl/onderwerpen/opsporing_en_handhaving/wbp/index.aspx
- <http://www.cbpweb.nl/>
- <http://www.aspnl.com/>
- <http://www.iso.org/>
- <http://www.fckeditor.net/>
- <http://www.useit.com/>

Begrippenlijst

Term / afkorting	Omschrijving
Actor	Een rol die interactie heeft met de nieuwsbriefmodule, dit kan een persoon maar ook een ander systeem zijn
Adresboek	Een digitale verzameling van alle contacten van een gebruiker
ASP.NET	De techniek waarmee Nieuwsdesk is ontwikkeld
Bounce	Een bounce treedt op wanneer een nieuwsbrief verstuurd wordt naar een e-mailadres, waardoor een melding gegenereerd wordt dat de nieuwsbrief niet of met een uitzondering kan worden afgeleverd
Bouncemanagement	Het proces dat bounces regelt
Compositie(relatie)	Een associatie tussen twee klassen waarbij de ene klasse een onderdeel is van de andere klasse
C#.NET	De programmeertaal waarmee de nieuwsbriefmodule ontwikkeld is
Contact	Een persoon dat voorkomt in een adresboek van een gebruiker van de nieuwsbriefmodule
Database	Een digitaal bestand dat gegevens bevat of waar gegevens kunnen worden toegevoegd/gewijzigd
Distributielijst	Een digitale lijst waar verscheidene emailadressen aan gekoppeld zitten
GUID	Global Unique IDentifier, een unieke waarde
Klasse	Verzameling van objecten met overeenkomstige eigenschappen
Mailing List	Zie distributielijst
MoSCoW	Methode die gebruikt wordt voor het vaststellen prioriteiten bij softwareontwikkeling
Nieuwsbrief	Een digitale informatiebrief
UP	Een methodiek die gebruikt wordt bij softwareontwikkeling
Sequentiediagram	Visuele weergave van een use case waarbij duidelijk wordt welke klassen gebruikt worden
SQL	Structured Query Language, programmeertaal die gebruikt wordt voor het aanspreken van databases
Statistieken	Gebruiksoverzicht van een nieuwsbrief, klant of contact
Template	Een standaardontwerp dat gebruikt wordt bij het aanmaken van een nieuwsbrief
Use case	Opsomming van de te doorlopen stappen van een functionaliteit

Figurenlijst

Figuur	Omschrijving
1	Organigram On-Site
2	Organisatiestructuur van het project
3	Sequentiediagram use case Nieuwsbrief verzenden
4	Deel van het Class Model
5	Deployment view Nieuwsdesk
6	Navigation Map beheerdersgedeelte
7	Navigation Map klantgedeelte
8	Deel van het Data Model Nieuwsdesk database
9	Master Page werking
10	Stap 1: contacten importeren uitgevoerd
11	Stap 2: Toewijzen van kolommen uit het geupload bestand, aan waarden die gebruikt worden in Nieuwsdesk
12	Stap 3: Het importeren van de contacten en weergeven van mislukte imports
13	Use case diagram iteratie 2
14	Business Object Model iteratie 2
15	Navigation Map klantgedeelte iteratie 2

Tabellenlijst

Tabel	Omschrijving
1	Belanghebbenden en toekomstige gebruikers Nieuwsdesk
2	Actoren overzicht Nieuwsdesk
3	Use case Model Nieuwsdesk
4	Basisscenario use case specification
5	Foutscenario use case specification
6	Kwaliteitsattributen Nieuwsdesk
7	Actoren overzicht Nieuwsdesk iteratie 2
8	Use Case Model iteratie 2

Interne bijlagen

Bijlage I – Opdrachtoomschrijving

Titel afstudeeropdracht:

Ontwikkeling van een nieuwsbriefmodule voor On-Site BV

Inleiding (organisatorische omgeving, kader, historie):

On-Site BV is een bedrijf dat gericht is op systeem- en netwerkbeheer en het ontwikkelen, leveren en beheren van Internet- en Multimedia-toepassingen. Als implementatiepartner van Ektron levert en ontwikkelt On-Site het Ektron CMS400.Net als Content Management System. Het bedrijf telt ongeveer 20 werknemers en is gevestigd in Rotterdam. Het heeft verscheidene klanten in verschillende branches.

Probleemstelling:

On-Site heeft voor verschillende klanten een mogelijkheid ontwikkeld waarbij een klant een nieuwsbrief kan versturen naar verschillende emailadressen (dmv een distributielijst). Dit kan door een template van een nieuwsbrief in te vullen in Microsoft Outlook en te versturen naar een emailadres. Wanneer een klant een nieuwsbrief wil maken en versturen, zijn er veel handmatige configuratiestappen nodig om dit proces te voltooien. Deze stappen moeten zorgvuldig uitgevoerd worden, een kleine fout en de nieuwsbrief wordt niet verstuurd of naar de emailadressen van een andere klant gestuurd. Tevens is het door deze situatie niet mogelijk om de emailadressen te beheren. Wanneer er een wijziging plaats moet vinden zal een klant dit doorgeven aan On-Site en zal er iemand bij On-Site de wijziging door moeten voeren. Ook is er via deze manier geen mogelijkheid om historie en statistieken bij te houden van een verstuurd nieuwsbrief. De klanten die gebruik maken van de nieuwsbriefmodule doen dit omdat ze een nieuwsbrief naar honderden/duizenden emailadressen sturen. Er zijn klanten bij die de nieuwsbrief naar 60.000 emailadressen (per week) sturen.

Doelstelling van de opdracht:

De doelstelling van deze opdracht is het ontwikkelen van de webapplicatie "Nieuwsdesk". Nieuwsdesk zal ontwikkeld worden in twee iteraties. Bij de ontwikkeling van de eerste iteratie zal rekening gehouden moeten worden met een mogelijke integratie met het Ektron CMS400.Net. Iteratie 1 moet bestaan uit:

- Beheerdersgedeelte (voor On-Site); Hiermee is het mogelijk om gebruikers van nieuwsdesk te beheren, distributielijsten te beheren, templates van nieuwsbrieven voor klanten te beheren, historie weergeven van alle verstuurd nieuwsbrieven van de klanten en statistieken van de klanten te bekijken.
- Adressenbeheer; Dit gedeelte maakt het mogelijk om contacten van een klant (de gebruiker van nieuwsdesk) te beheren.
- Distributielijstbeheer; Dit gedeelte maakt het mogelijk om distributielijsten van een klant te beheren.
- Nieuwsbriefbeheer; Dit gedeelte maakt het mogelijk om nieuwsbrieven aan te maken, te bekijken en te versturen, rapportage genereren van een verstuurd nieuwsbrief en historie bijhouden van verstuurd nieuwsbrieven.
- Statistieken; Dit gedeelte geeft een overzicht per klant van de statistieken van een contact of artikel en aan- en afmeldingen van de laatste 30 dagen. Zo kan er bijvoorbeeld worden gecontroleerd welk contact welk artikel heeft geopend.
- Bouncemanagement; Met bounce management wordt het mogelijk om per nieuwsbriefmailing, een emailadres van een contact te voorzien van een status. Een contact voorzien van een status zal geautomatiseerd worden door het gebruik van een applicatie op de mailserver. Deze status geeft inzicht in de beschikbaarheid van een contact (bijv. emailadres bestaat niet meer).
- Opt-in en opt-out; Met deze functionaliteit is het mogelijk voor een contact om zichzelf in en uit te schrijven voor een nieuwsbrief van een klant via zijn eigen website. Voor deze functionaliteit zal er een API ontwikkeld moeten worden.

- Importfunctie voor contacten; Contacten vanuit Excel, CSV en/of Outlook moeten kunnen worden geïmporteerd naar het adresboek van een klant in Nieuwsdesk.

Na de implementatie van iteratie 1 kan er worden begonnen aan de ontwikkeling van iteratie 2 van Nieuwsdesk. Iteratie 2 van Nieuwsdesk zal meer als marketing tool gaan fungeren. Dit programma zal gebruikt gaan worden door werknemers van On-Site en haar dochterbedrijven. Het programma moet de volgende mogelijkheden bieden:

- Ad hoc distributielijsten kunnen samenstellen. In iteratie 1 van Nieuwsdesk krijgt een gebruiker een aantal distributielijsten toegewezen. In iteratie 2 moet het mogelijk zijn om ad hoc contacten aan een distributielijst te koppelen.
- Contactstatistieken; De mailing naar de gemaakte ad hoc distributielijst moet worden vastgelegd. Een contact kan zich uitschrijven dmv de opt-out functie uit iteratie 1. Doordat vastgelegd wordt dat het contact zich uitschrijft dmv een opt-out in een bepaalde nieuwsbrief kan actie worden ondernomen door de werknemers van On-Site om dit contact te benaderen.

Benodigde software

- Microsoft Visual Studio 2003 of 200
- Microsoft SQL Server 2000 of 2005
- ArgoUML 0.22 (Modelleertool)
- Microsoft IIS (Webserver)

Benodigde hardware

Een computer met minimaal 1,8 Ghz processor en 512 MB werkgeheugen, dit voor het draaien van de ontwikkelsoftware. Daarnaast is het noodzakelijk dat er een ontwikkelomgeving aanwezig is die losstaat van de productieomgeving.

Beschikbare rapporten

Het Ektron CMS400.Net is gedocumenteerd

Aanwezige ideeën

Het grafische ontwerp voor www.nieuwsdesk.nl dat gebruikt zal worden voor www.nieuwsdesk.nl

Concrete werkzaamheden:

- Orientatie Merak mailserver
- Onderzoek Bounce management
- Onderzoek bestaande nieuwsbriefmodules
- Onderzoek Emailmarketing/Privacy/SPAM
- Rapport Inception fase
- Rapport Elaboration fase
- Rapport Construction fase
- Rapport Transition fase
- Nieuwsdesk.
- SQL Server database

Te hanteren methodieken

UP

Te gebruiken technieken

ASP.NET, ADO.NET, UML, TMAP

Te vermelden nadrukken

De nieuwsbriefmodule zal moeten aansluiten op de andere gemaakte en reeds gebruikte programmatuur van On-Site qua huisstijl, gebruikte software (Microsoft SQL Server, IIS etc.). Daarnaast zal de nieuwsbriefmodule beheerbaar, beveiligd en gebruiksvriendelijk moeten zijn. Na afsluiting van het project zal de nieuwsbriefmodule beheerd en onderhouden worden door On-Site BV.

Bijlage II – Vision Document

Nieuwsdesk

Vision Document

Ontwikkeling Nieuwsdesk

Versie 2.0
Rotterdam, 18 januari 2007
Auteur:
Tony Kanters

Inhoudsopgave

1. Inleiding	51
1.1. Doel van dit document	51
1.2. Reikwijdte	51
1.3. Definities en afkortingen	51
2. Positionering van de applicatie	52
2.1. Het huidige bedrijfsproces	52
2.2. Probleemstelling	52
3. Belanghebbenden en gebruikers	53
3.1. Rollenoverzicht belanghebbenden	53
3.2. Rollenoverzicht gebruikers	53
3.3. Profiel van de belanghebbenden	54
3.3.1. Opdrachtgever	54
3.3.2. Teamleider gebruikers	54
3.3.3. Opdrachtnemer	54
3.4. Profiel van de gebruikers	54
3.4.1. Applicatiebeheerder	54
3.4.2. Klant	54
3.4.3. Salesmedewerker	55
3.4.4. CMS	55
3.4.5. Tester	55
4. Risico's	56
5. Productperspectief	57
6. Functionele systeemeisen	58
7. Niet-functionele systeemeisen	60
8. Documentatie	61
8.1. Gebruikershandleiding/Helpfunctie	61

1. Inleiding

Het doel van dit document is om de eisen en wensen van de On-Site BV voor Nieuwsdesk te verzamelen, analyseren en te definiëren.

1.1. Doel van dit document

Dit document geeft de visie van On-Site BV en de opdrachtnemer Tony Kanters van het project "Ontwikkeling van Nieuwsdesk".

Dit document dient als basis voor het project omdat het een overzicht geeft van de eisen en wensen van de te ontwikkelen applicatie. Vanuit het Vision document zal worden gewerkt aan de realisatie van de Nieuwsdesk.

1.2. Reikwijdte

Dit Vision document hoort bij het project "Ontwikkeling van Nieuwsdesk". Dit project is onderdeel van de afstudeeropdracht van Tony Kanters, student aan de Haagse Hogeschool.

1.3. Definities en afkortingen

Zie de Glossary.

2. Positionering van de applicatie

2.1. Het huidige bedrijfsproces

On-Site heeft voor verschillende klanten (ook voor eigen gebruik) een mogelijkheid ontwikkeld waarbij een klant een nieuwsbrief kan versturen naar verschillende emailadressen (dmv een distributielijst). Dit kan door een template van een nieuwsbrief in te vullen in Microsoft Outlook en te versturen naar een emailadres. Wanneer een klant een nieuwsbrief wil maken en versturen, zijn er veel handmatige configuratiestappen nodig om dit proces te voltooien. Deze stappen moeten zorgvuldig uitgevoerd worden, een kleine fout en de nieuwsbrief wordt niet verstuurd of naar de emailadressen van een andere klant gestuurd.

Tevens is het door deze situatie niet mogelijk om de emailadressen te beheren. Wanneer er een wijziging plaats moet vinden zal een klant dit doorgeven aan On-Site en zal er iemand bij On-Site de wijziging door moeten voeren. Ook is er via deze manier geen mogelijkheid om historie en statistieken bij te houden van een verstuurd nieuwsbrief.

De klanten die gebruik maken van de Nieuwsdesk doen dit omdat ze een nieuwsbrief naar honderden/duizenden emailadressen sturen. Er zijn klanten bij die de nieuwsbrief naar 60.000 emailadressen (per week) sturen.

2.2. Probleemstelling

Het probleem is:	Het huidige bedrijfsproces bestaat uit teveel handmatige configuratiestappen. Doordat het proces niet dagelijks doorlopen wordt, is het telkens weer een moeilijk klus.
Dit beïnvloedt:	Medewerkers en klanten van On-Site BV die gebruik maken van de huidige oplossing.
De impact hiervan is:	Dat een nieuwsbrief niet verstuurd wordt of dat een klant zijn nieuwsbrief naar de klanten van een andere klant sturen. Daarnaast is er geen mogelijkheid tot statistieken en rapportages. Hierdoor is de werking van een nieuwsbrief niet te achterhalen.
Een succesvolle oplossing zou zijn:	On-Site wil graag twee versies van Nieuwsdesk. Versie één zal moeten bestaan uit de volgende functionaliteiten; adressenbeheer, distributielijstbeheer, nieuwsbriefbeheer, statistieken, bouncemanagement, opt-in/opt-out en een importfunctionaliteit voor contacten. De tweede versie zal de gebruiker de mogelijkheid geven om ad hoc distributielijsten aan te maken en daarnaast moet het inzicht geven in uitgebreidere contactstatistieken. Bij de ontwikkeling van Nieuwsdesk dient rekening gehouden te worden met een mogelijke integratie met het Ektron CMS400.Net (dit geldt alleen voor de eerste versie). Om de hele situatie te beheren is er behoefte aan een beheermogelijkheid. Door deze mogelijkheid is de applicatiebeheerder in staat om een klant toe te voegen en aan te passen. Ook moet er per klant statistieken bekeken worden. De eerste versie zal gebruikt gaan worden salesmedewerkers en klanten van On-Site BV. De tweede versie zal meer als CRM pakket gaan fungeren en zal daarom alleen toegankelijk zijn voor salesmedewerkers van On-Site BV.

3. Belanghebbenden en gebruikers

Opdrachtgever: Dik Bots
 Teamleider gebruikers: Vincent Bendeler
 Opdrachtnemer: Tony Kanter
 Applicatiebeheerder: Marcel de Jong
 Gebruikers: Sales staff van On-Site BV en klanten die gebruik gaan maken van Nieuwsdesk (deze worden vertegenwoordigd door Vincent Bendeler).

3.1. Rollenoverzicht belanghebbenden

Rol	Beschrijving	Verantwoordelijkheden
Opdrachtgever		• Specificeren eisen en wensen • Accepteert het systeem • Voorziet in beschikbaarheid hard- en software • Neemt applicatie in beheer • Specificeren (gedetailleerde) eisen en wensen • Keurt Vision Document
Teamleider gebruikers	Begeleid het project vanuit de kant van On-Site BV	• Contactpersoon voor opdrachtnemer • Tussenpersoon voor klanten van On-Site BV • Beschikbaar stellen van gebruikers voor het project • Bewaken van voortgang van project
Opdrachtnemer		• Ontwerpen van Nieuwsdesk • Ontwikkelen van Nieuwsdesk • Implementeren van Nieuwsdesk • Contactpersoon voor de teamleider van gebruikers • Projectmanagement

3.2. Rollenoverzicht gebruikers

Rol	Beschrijving	Verantwoordelijkheden
Applicatie beheerder	Het onderhouden en beheren van de oplossing na implementatie	• Beheren van de applicatie • Aanspreekpunt voor gebruikers • Klantenbeheer
Klant	De eindgebruiker van Nieuwsdesk	• Het gebruiken van de eerste versie van Nieuwsdesk
Salesmedewerker	De eindgebruiker van Nieuwsdesk	• Het gebruiken van beide versies van Nieuwsdesk
CMS	Aanspreken van functionaliteiten	• Het gebruiken van sommige functionaliteiten
Tester	Tester van de Oplossing	• Testen van de oplossing aan de hand van de aangeleverde testsets

3.3. Profiel van de belanghebbenden

3.3.1. Opdrachtgever

Vertegenwoordiger	Dik Bots
Beschrijving	
Expertisegebied	Systeem- en applicatiebeheer, applicatieontwikkeling
Verantwoordelijkheden	• Specificeren eisen en wensen • Accepteert het systeem • Voorziet in beschikbaarheid hard- en software • Neemt applicatie in beheer
Succes Criteria	Het project is succesvol verlopen wanneer het doel van het project gehaald is, dus het opleveren van Nieuwsdesk

3.3.2. Teamleider gebruikers

Vertegenwoordiger	Vincent Bendeler
Beschrijving	Begeleid het project vanuit de kant van On-Site BV
Expertisegebied	Commercie
Verantwoordelijkheden	• Keurt plan van aanpak • Specificeren (gedetailleerde) eisen en wensen • Contactpersoon voor opdrachtnemer • Tussenpersoon voor klanten van On-Site BV • Beschikbaar stellen van gebruikers voor het project • Bewaken van voortgang van project
Succes Criteria	Het project is succesvol verlopen wanneer het doel van het project gehaald is, dus het opleveren van Nieuwsdesk

3.3.3. Opdrachtnemer

Vertegenwoordiger	Tony Kanters
Beschrijving	De opdrachtnemer van het project
Expertisegebied	Applicatieontwikkeling
Verantwoordelijkheden	• Ontwerpen van de oplossing • Ontwikkelen van de oplossing • Implementeren van de oplossing • Contactpersoon voor de teamleider van gebruikers • Projectmanagement
Succes Criteria	Het project is succesvol verlopen wanneer het product voldoet aan de eisen en wensen van de opdrachtgever

3.4. Profiel van de gebruikers

3.3.4. Applicatiebeheerder

Vertegenwoordiger	Marcel de Jong
Beschrijving	Het onderhouden en beheren van de oplossing na implementatie
Expertisegebied	Applicatieontwikkeling, applicatiebeheer
Verantwoordelijkheden	• Beheren van de applicatie • Aanspreekpunt voor gebruikers • Klantenbeheer
Succes Criteria	Het project is succesvol verlopen wanneer er een onderhoudbaar en beheerbaar product wordt opgeleverd

3.3.5. Klant

Vertegenwoordiger	Klant van On-Site BV
--------------------------	----------------------

Beschrijving	De eindgebruiker van de oplossing
Expertisegebied	Uitvoeren van bedrijfsprocessen
Verantwoordelijkheden	Het gebruiken van de oplossing
Succes Criteria	Het project is succesvol verlopen wanneer er een product wordt opgeleverd waarmee de klant het bedrijfsproces kan voltooien

3.3.6. Salesmedewerker

Vertegenwoordiger	Salesmedewerkers van On-Site BV
Beschrijving	De eindgebruiker van de oplossing
Expertisegebied	Uitvoeren van bedrijfsprocessen
Verantwoordelijkheden	Het gebruiken van de oplossing
Succes Criteria	Het project is succesvol verlopen wanneer er een product wordt opgeleverd waarmee de salesmedewerker het bedrijfsproces kan voltooien

3.3.7. CMS

Vertegenwoordiger	Het Ektron CMS400.NET
Beschrijving	Dit Content Management Systeem zal gebruik maken van de functionaliteiten gemaakt in iteratie 1
Expertisegebied	
Verantwoordelijkheden	
Succes Criteria	Een product waarvan de functionaliteiten gebruikt kunnen worden door het Ektron CMS400.NET

3.3.8. Tester

Vertegenwoordiger	Aangewezen testers door de teamleider
Beschrijving	De tester van de oplossing
Expertisegebied	
Verantwoordelijkheden	Het testen van de oplossing
Succes Criteria	Een product waarvan de kwaliteit hoog van is

4. Risico's

Onderstaande risico's zijn van toepassing zijn op dit project:

1. De opdrachtnemer heeft geen kennis van nieuwsbriefmodules
2. De opdrachtnemer heeft geen kennis van bounce management
3. De opdrachtnemer heeft nauwelijks kennis van mailservers en mailprotocollen
4. De opdrachtnemer heeft geen kennis van het te gebruiken Ektron CMS400.Net
5. Wegens privacy redenen mag er niet zomaar informatie van klanten/personen die de nieuwsbrief lezen geregistreerd worden

Om de risico's te verkleinen/vermijden zullen de volgende maatregelen getroffen worden:

1. Er zal een onderzoek gestart moeten worden waar bestaande nieuwsbriefmodules onderzocht worden
2. Er zal een onderzoek gestart moeten worden waar het begrip bounce management zodanig onderzocht wordt dat bounce management voor dit project toepasbaar wordt
3. Er zal een onderzoek gepland moeten worden naar mailprotocollen en bestaande mailservers. Aan de hand van dit onderzoek zal er voor een bepaalde mailserver gekozen gaan worden
4. De opdrachtnemer zal een handleiding voor het gebruik van het Ektron CMS400.Net moeten doorlopen. Daarnaast zal er onderzocht moeten worden, welk(e) onderdeel(e)n van het Ektron CMS400.Net voor dit project van belang is(zijn)
5. Er zal moeten worden geïnformeerd in hoeverre en op welke manier er informatie van klanten/personen geregistreerd mag worden

5. Productperspectief

Nieuwsdesk zal een webapplicatie zijn. Nieuwsdesk zal aanspreekbaar zijn via een internetbrowser en/of het Ektron CMS400.NET.

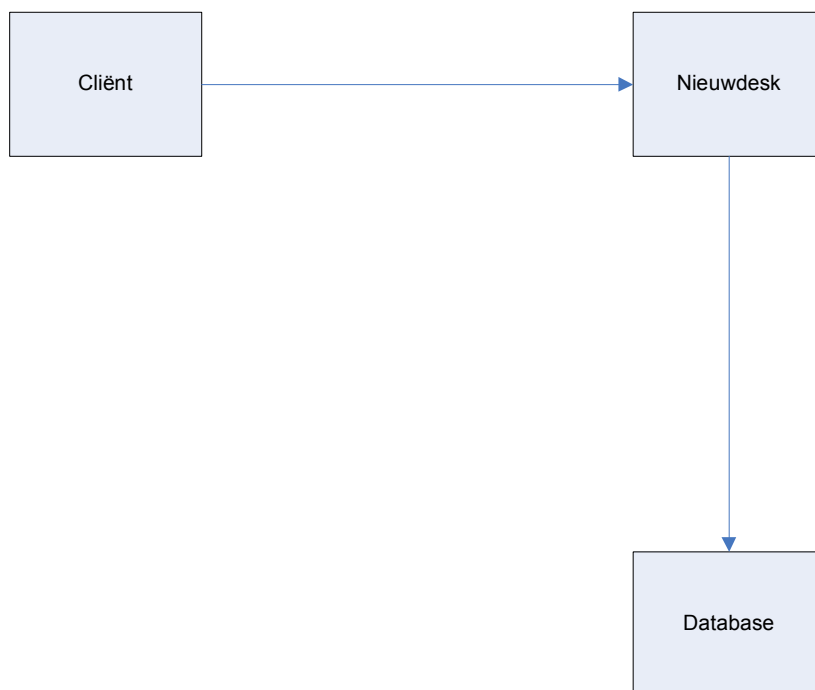
Nieuwsdesk zal hierdoor uit drie verschillende onderdelen bestaan:

- § Cliënt
- § Nieuwsdesk (webapplicatie)
- § Database

De cliënt zal in directe verbinding staan met Nieuwsdesk. Nieuwsdesk zal naast de directe verbinding met cliënt ook een directe verbinding hebben met de database.

De database zal in directe verbinding staan met Nieuwsdesk.

Dit levert het volgende schema op:



6. Functionele systeemeisen

De eerste versie van Nieuwsdesk moet aan de volgende functionele systeemeisen voldoen:

- Beheerder, bestaande uit:
 - Klantbeheer, bestaande uit:
 - Klant toevoegen
 - Klant zoeken
 - Klant verwijderen
 - Klant bekijken
 - Klant wijzigen
 - Status klant wijzigen
 - Groep klant wijzigen
 - Distributielijstbeheer, bestaande uit:
 - Distributielijst toevoegen
 - Distributielijst zoeken
 - Distributielijst verwijderen
 - Distributielijst wijzigen
 - Distributielijst bekijken
 - Templatebeheer, bestaande uit:
 - Template toevoegen
 - Template bekijken
 - Template zoeken
 - Template verwijderen
 - Template voorbeeld bekijken
 - Template wijzigen
 - Statistieken
 - Statistieken verzonden nieuwsbrieven bekijken (per klant)
 - Statistieken nog niet verzonden nieuwsbrieven bekijken (per klant)
 - Nieuwsbrief met toegevoegde artikelen bekijken
 - Bekijken aan welke distributielijst en hoeveel contacten een nieuwsbrief verstuurd is
- Adressenbeheer, bestaande uit:
 - Contact zoeken
 - Contact toevoegen
 - Contact bekijken
 - Contact wijzigen
 - Contact verwijderen
 - Contact uitschrijven
 - Contact koppelen met distributielijst(en)
 - Contact(en) importeren
- Distributielijstbeheer, bestaande uit:
 - Distributielijst bekijken
 - Distributielijst wijzigen
 - Distributielijst zoeken
 - Distributielijst koppeling(en) met contacten bekijken
 - Contacten zonder distributielijst bekijken
 - Contact uit distributielijst selecteren en bekijken
- Artikelbeheer
 - Artikel maken
 - Artikel zoeken
 - Artikel bekijken
 - Artikel wijzigen
 - Artikel verwijderen
- Nieuwsbriefbeheer
 - Nieuwsbrief maken
 - Nog niet verzonden nieuwsbrieven bekijken
 - Verzonden nieuwsbrieven bekijken
 - Nieuwsbrief bekijken
 - Nieuwsbrief verzenden

- Statistieken
 - Statistieken per contact bekijken
 - Statistieken per nieuwsbrief bekijken
 - Statistieken aan- en afmeldingen bekijken
 - Bounce statistieken bekijken

De tweede versie zal extra functionele systeemeisen kennen:

- Distributielijstbeheer:
 - Ad hoc distributielijsten kunnen samenstellen en daar contacten aan koppelen
- Statistieken:
 - De mailing naar de ad hoc distributielijst (met contacten) moet worden opgeslagen
 - Contacten kunnen zich uitschrijven dmv de opt-out functie. Wanneer een contact zich uitschrijft dmv de opt-out moet worden vastgelegd welk contact en van welke nieuwsbrief de opt-out gebruikt is

7. Niet-functionele systeemeisen

De Nieuwsdesk zal naast de functionele systeemeisen ook moeten voldoen aan een aantal niet-functionele systeemeisen:

- Nieuwsdesk moet werken op een Windows machine met Internet Explorer (voor de Regular versie)
- De functionaliteiten moeten aanspreekbaar zijn voor de CMS versie
- Nieuwsdesk moet een GUI hebben die aansluit op de huisstijl van On-Site BV
- Gebruikers/klanten moeten onafhankelijk van elkaar kunnen werken met de Nieuwsdesk. Als de ene klant iets doet moet dit geen invloed hebben op de andere
- Nieuwsdesk moet beheer- en onderhoudbaar zijn
- Nieuwsdesk moet beveiligd zijn
- Nieuwsdesk moet gebruiksvriendelijk zijn
- Nieuwsdesk moet aansluiten op de reeds gebruikte software die ook voor andere applicaties gebruikt wordt (Microsoft SQL Server, Microsoft IIS)
- Ektron CMS400.Net moet bepaalde functionaliteiten van Nieuwsdesk aan kunnen spreken
- Op elk gewenst moment dient een nieuwsbrief aangemaakt/gelezen te worden

8. Documentatie

8.1. Gebruikershandleiding/Helpfunctie

Voor het gebruik van Nieuwsdesk dient een handleiding/helpfunctie beschikbaar te zijn. De initiële versie zal worden gemaakt door de opdrachtnemer. Na afsluiting zal de handleiding/helpfunctie onder het beheer vallen van de applicatiebeheerder.

De handleiding zal online beschikbaar moeten zijn.

Externe bijlagen

- Externe bijlage I - Rapport Inception Fase
- Externe bijlage II - Rapport Elaboration Fase
- Externe bijlage III - Rapport Construction Fase
- Externe bijlage IV - Rapport Transition Fase