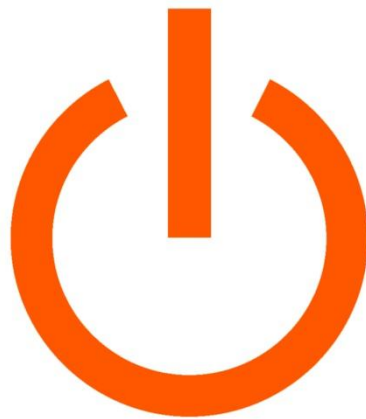


Afstudeerverslag

Ontwikkelen van een enquêtesysteem bij Webtechniek.



WEBTECHNIEK

Student : Mark Suurland

Studentnummer : 09071547

Opleiding : Informatica

Onderwijsinstelling : Haagse Hogeschool

Locatie : Den Haag

Begeleidend examiner: A.M.J.J. Lousberg-Orbons

Tweede examiner: P.R.C. Breukel.

Afstudeerperiode : 6 februari 2012 – 1 juni 2012

Bedrijf : Webtechniek

Adres : Sadatweg 18, Delft

Begeleider : ing. M.M.R Jongmans

Datum : 29 mei 2012

Versie : 1.0

Referaat

Suurland, M.V., Ontwikkelen van een enquêtesysteem bij Webtechniek

Dit verslag geeft een beschrijving van het proces dat tijdens de afstudeerperiode bij Webtechniek door de afstudeerder is doorlopen in het kader van zijn Informatica studie aan de Haagse Hogeschool. De afstudeerperiode heeft plaatsgevonden in de periode februari 2012 – juni 2012.

Descriptoren

- RUP
- ASP.NET
- Enquêtetechniek

Voorwoord

Na zeventien weken hard werken is mijn afstudeeropdracht afgerond. Ik weet nog goed toen ik voor het eerst naar binnen liep bij Webtechniek. Een klein bedrijf, nieuwe mensen, nieuwe namen maar vooral een plek waar ik mij gelijk thuis voelde. Inmiddels is het bijna zomer en is het werken bij Webtechniek een gewoonte geworden.

Tijdens deze opdracht heb ik veel geleerd. Door te werken aan een totaal nieuw systeem ontstonden er veel uitdagingen zoals het omgaan met een grote groep stakeholders en requirements. Omdat het een nieuw systeem betreft, kon ik veel keuze zelf maken die uiteindelijk tot een totaalproduct hebben geleid.

Bij deze wil ik van de gelegenheid gebruik maken om Maurice te bedanken. Maurice was mijn bedrijfsmentor en heeft mij begeleid tijdens de afstudeeropdracht. Ook wil ik Wouter en Robin bedanken. Hoewel zij mij niet hebben begeleid op procesmatig niveau kon ik wel bij hun terecht met technische vragen. Daarnaast wil ik alle overige medewerkers van Webtechniek bedanken voor de geweldige tijd bij Webtechniek. Tot slot wil ik Alwine Lousberg-Orbons en Paul Breukel bedanken voor hun begeleiding vanuit school.

Mark Suurland
Delft, 14 mei 2012

Inhoud

1	Inleiding	1
2	Webtechniek en zijn producten	2
2.1	Webtechniek	2
2.2	Producten van Webtechniek	2
3	De afstudeeropdracht.....	3
3.1	Aanleiding.....	3
3.2	Probleemstelling	3
3.3	Doelstelling.....	3
3.4	Resultaat	3
4	Aanpak.....	4
4.1	Projectmethodiek.....	4
4.2	Vaststellen Requirements	5
4.3	Ontwerpen software	6
4.4	Bouwen software	6
4.5	Testen software.....	7
4.6	Beroepstaken	9
5	Opstart van het project	10
5.1	Plan van aanpak	10
5.2	Vooronderzoek.....	12
5.3	Interviews Webtechniek stakeholders & klanten stakeholders	14
5.4	Aanpassing planning: globale requirements	17
5.5	OTAP	17
5.6	Inception rapport	18
6	Requirements en ontwerp van enquêteteknik	19
6.1	Gedetailleerde requirements	19
6.2	Use cases	25
6.3	Architectuur enquêteteknik	27
6.4	Ontwerp Enquêteteknik	30
6.5	Elaboration rapport.....	38
7	Bouw onderdeel 1: Creëren enquête	39
8	Bouw onderdeel 2: beheer klanten en beheer Webtechniek	45
9	Bouw onderdeel 3: Distribueren enquête	47
10	Bouw onderdeel 4: Resultaten enquête	49
10.1	Klanten interviews	50
11	Testen	53
11.1	Mastertestplan.....	53
11.2	Testrisicoanalyse.....	53
11.3	Planning testtraject.....	54
11.4	Testontwerp.....	54

11.5	Systeemtest	58
11.6	Testrapport	58
12	Evaluatie	60
12.1	Aanpak	60
12.2	Producten	61
12.3	Beroepstaken	62
	Verklarende woordenlijst	64
	Bronnenlijst.....	66

1 Inleiding

Het verslag dat hier voor u ligt is geschreven als eindverslag van mijn afstudeeropdracht bij het bedrijf Webtechniek. In dit eindverslag zal ik beschrijven wat mijn opdracht was en hoe ik deze heb uitgevoerd. Bij deze beschrijving zal ik de keuzes toelichten die ik heb gemaakt.

Allereerst zal ik in hoofdstuk 2 een beschrijving geven van het bedrijf Webtechniek en de producten die zij ontwikkelen en verkopen. Tevens zal ik toelichten wat mijn plek binnen de organisatie was.

In hoofdstuk 3 zal ik de situatie beschrijven bij aanvang van het afstuderen, hieruit is tevens mijn afstudeeropdracht ontstaan.

In hoofdstuk 4 zal ik mijn aanpak van dit project toelichten. Hierbij licht ik toe welke methodieken en technieken ik heb gebruikt en waarom ik deze heb toegepast.

In hoofdstuk 5 zal ik de opstart van het project beschrijven. Tijdens de opstart van dit project verdiep ik mijzelf in de opdracht, een onderdeel hiervan is het vooronderzoek.

In hoofdstuk 6 beschrijf ik hoe de requirements en ontwerpen van enquêteteknik zijn opgesteld.

In hoofdstuk 7 tot en met hoofdstuk 10 wordt beschreven hoe de bouw van de verschillende onderdelen van enquêteteknik zijn verlopen.

In hoofdstuk 11 wordt beschreven hoe het testen van enquêteteknik is verlopen.

In hoofdstuk 12 evalueer ik tot slot het project waarbij ik terug kijk naar mijn aanpak, opgeleverde producten en de beroepstaken.

2 Webtechniek en zijn producten

Voordat mijn afstudeeropdracht zal worden besproken, beschrijf ik eerst het bedrijf Webtechniek en de producten die zij ontwikkelen.

2.1 Webtechniek

Webtechniek is een commercieel ICT bedrijf gevestigd in Delft. Webtechniek biedt een totaalpakket voor webomgevingen. Van eenvoudige websites tot ingewikkelde managementsystemen, van hosting tot technische ondersteuning. Het bedrijf is vooral gericht op het ontwikkelen van webapplicaties en probeert zich te onderscheiden door maatwerk te leveren aan haar klanten. Webtechniek werkt voor individuele bedrijven, maar ook veel voor vormgevers, marketeers en adviesbureaus.

Het bedrijf is opgericht in het jaar 2000. Het personeel dat bij Webtechniek werkt is vrij jong en varieert van 20 tot 35 jaar. Momenteel werken er rond de tien medewerkers waarvan een deel parttime. Mede doordat het personeel vrij jong is, hangt er een informele werk sfeer en bestaan hierdoor geen aparte afdelingen. Er zijn echter wel aparte rollen/functies binnen het bedrijf. Zo is er een commercieel directeur die alle contacten/klanten beheert. Er is een technische directeur die voor alle technische ondersteuning zorgt zowel binnen het bedrijf als naar de klanten toe. Hiernaast zijn er medewerkers (programmeurs) die het werk uitvoeren, deze worden ondersteund door twee project managers. Mijn plaats binnen de organisatie is als junior programmeur.

De applicaties en websites die Webtechniek maakt, worden met name in ASP.NET ontwikkeld. De twee frameworks die als standaard binnen het bedrijf worden beschouwd zijn MVC en Web-forms.

2.2 Producten van Webtechniek

Webtechniek is vooral actief bezig met het ontwikkelen van webapplicaties. Om twee voorbeelden te geven zal ik het meubel order systeem (MOS) en mailtechniek kort uitleggen.

2.2.1 MOS

Voor de meubelbranche heeft Webtechniek een online order magazijnsysteem ontwikkeld (MOS). Op dit moment werkt onder meer een grote keten van twintig winkels met het softwaresysteem. Alle processen van een meubelorder worden in het systeem ondervangen; van order, bestelling tot transport, opslag in magazijn en levering van de meubels. Daarnaast is het systeem te koppelen aan hun online boekhouding systeem, de SMS module en een webshop met iDeal integratie.

2.2.2 Mailtechniek

Via de nieuwsbriefmodule (mailtechniek) van Webtechniek kan eenvoudig door de klant een nieuwsbrief worden opgesteld en snel nieuwsbrieven worden verstuurd aan een groep mensen. De module is zo opgebouwd dat aan de laatste anti-spam wetgevingseisen wordt voldaan. De module geeft ook inzicht in het lees- en klikgedrag van de ontvangers van de verstuurd nieuwsbrief.

3 De afstudeeropdracht

In dit hoofdstuk zal de afstudeeropdracht worden toegelicht.

3.1 Aanleiding

Webtechniek is bezig een aantal eigen producten te ontwikkelen met als doel deze te verkopen aan hun bestaande klantenkring. Deze producten worden als Software as service (SAAS) aangeboden, dat wil zeggen dat de klant middels een abonnementsvorm toegang krijgt tot de online software. Webtechniek wil deze “nieuwe” manier van diensten uitbreiden om zo de klanten meer aan zich te binden. Zo is er een applicatie ontwikkeld om nieuwsbrieven te ontwerpen, in te richten en uiteindelijk te versturen. Tevens wordt er gewerkt aan een systeem om makkelijk een webshop in te richten (door de klanten zelf).

3.2 Probleemstelling

Er is veel vraag vanuit de klanten van Webtechniek naar een enquêtesysteem, maar aan deze vraag wordt momenteel niet voldaan. Hierdoor verliest Webtechniek een mooie kans om klanten meer aan zich te binden en in te spelen op hun vraag. Webtechniek ziet dan ook de mogelijkheid tot het ontwikkelen van een eigen online enquêtesysteem dat aan hun klanten kan worden verkocht en dat zich onderscheidt ten opzichte van de bestaande enquêtesystemen.

3.3 Doelstelling

Het doel van deze afstudeeropdracht is de vraag van de klanten van Webtechniek naar een enquêtesysteem in te vullen. Wanneer het doel is behaald zal dit product meer binding geven tussen Webtechniek en hun bestaande klanten (er is immers ingespeeld op hun vraag). Daarnaast is een nieuw/extra systeem natuurlijk ook geschikt voor het aantrekken van nieuwe klanten voor Webtechniek.

Om dit doel te bereiken zal er een enquêtesysteem worden ontwikkeld. Er zal rekening moeten worden gehouden met niet-functionele requirements zoals beveiliging en gebruiksvriendelijkheid. Tevens moet integratie met andere modules (van Webtechniek) in de toekomst mogelijk zijn. Het enquêtesysteem zal uit een aantal hoofdonderdelen bestaan, namelijk de volgende:

Klant:

1. Het creëren van enquêtes.
 - a. Een inleiding, vragen en bijhorende antwoorden, afsluiting etc.
 - b. Het creëren van lay-outs (dit is een vormgeving die kan worden gekoppeld aan een enquête).
 - c. Het creëren van groepen (deze groepen kunnen worden gekoppeld aan een enquête om zo een onderscheid te maken tussen verschillende doelgroepen binnen een enquête).
2. Een beheer onderdeel voor de klanten, waarbinnen de bestaande enquêtes, lay-outs, groepen kunnen worden aangepast.
3. Het tonen en verspreiden van de enquête waardoor deze kan worden ingevuld door de respondenten (groepen).
4. Het tonen en exporteren van statistieken.

Webtechniek:

5. Een beheer onderdeel voor Webtechniek zelf, waarbinnen alle klanten (die gebruik maken van het enquêtesysteem) kunnen worden beheerd.

3.4 Resultaat

Wanneer de opdracht met succes is uitgevoerd, is er een nieuw product gerealiseerd, namelijk het enquête systeem. Dit product zal voldoen aan de gestelde eisen die voortvloeien uit het gebruikerdoel van de verschillende stakeholders (klanten van Webtechniek en het interne personeel). Het enquêtesysteem is klaar om verkocht te worden aan de klanten van Webtechniek. Met dit product zal Webtechniek de bestaande klanten meer aan zich binden. Uiteraard zal de verkoop van het enquêtesysteem resulteren in een hogere omzet.

4 Aanpak

In dit hoofdstuk zal ik toelichten welke aanpak ik heb toegepast tijdens dit project. Hiernaast zal ik toelichten welke technieken ik heb gebruikt tijdens het ontwikkelen van de applicatie.

4.1 Projectmethodiek

Ik heb gekozen om tijdens dit project met RUP als projectmethodiek te werken (Rational Unified Process). Dit is een iteratieve en incrementele softwareontwikkelingmethode die gebaseerd is op een aantal principes en best practices. RUP is gekozen omdat ik er zelf ervaring mee heb en Webtechniek er ook bekend mee is.

Verder bleek dit project geschikt voor RUP om de volgende redenen:

- Er wordt een nieuw systeem ontwikkeld van aanpak tot testen (eventueel implementatie), de fases en disciplines van RUP helpen om structuur te brengen in de ontwikkelfase.
- Er zijn duidelijke mijlpalen, zoals de einddatum en oplevering van producten binnen het bedrijf.
- Er wordt software ontwikkeld.
- De opdrachtgever is bekend met RUP, hierdoor is hij bekend met de methode die ik gebruik waardoor er makkelijker kan worden gecommuniceerd over de aanpak binnen het project.

RUP maakt gebruik van de volgende fases:

1. Inception

In deze fase ligt de nadruk op het opstarten van het project, en om het project verder te verkennen.

2. Elaboration

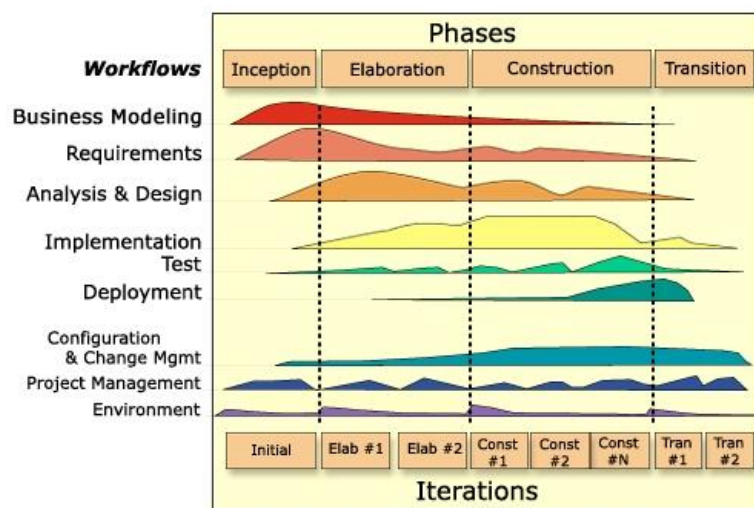
In deze fase ligt de nadruk op het ontlocken en specificeren van de requirements die voortvloeien vanuit betrokken stakeholders, als resultaat hiervan zal een gedetailleerd ontwerp ontstaan.

3. Construction

Tijdens deze fase ligt de nadruk op het ontwikkeling van het systeem.

4. Transition

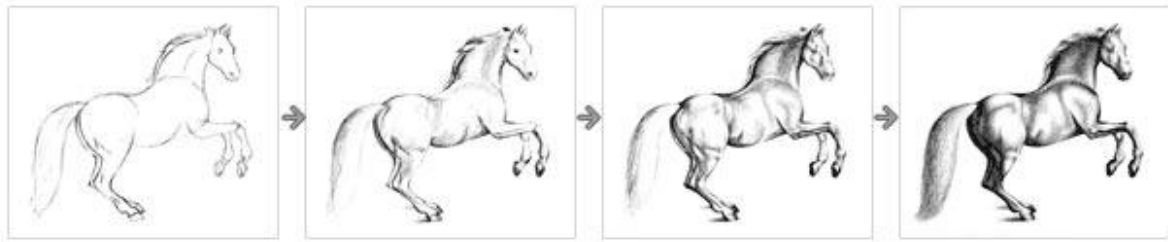
In deze fase ligt de nadruk op de afronding van het project waarin de kwaliteit van het systeem wordt getest.



Figuur 1 RUP fases en disciplines

Iteratieve en incrementele werkwijze

Een belangrijk aspect van RUP is de iteratieve en incrementele werkwijze. Met iteratief werken wordt bedoeld dat er eerst een voorlopige versie wordt ontwikkeld, waarna je feedback kunt vragen door middel van testen en/of reviews van de betrokken stakeholders. Aan de hand van deze test en review momenten wordt het systeem aangepast indien nodig.



Figuur 2 Iteratieve werkwijze, de applicatie krijgt langzaam vorm

Met incrementeel werken wordt bedoeld dat het systeem wordt opgesplitst in kleinere delen, deze zijn eerder benoemd in §3.3. Een dergelijke opsplitsing, een increment genoemd, wordt vervolgens uitgewerkt (ontwerp, bouw, test en implementatie). Nadat een increment is gebouwd zal deze worden getest om te kijken of nog alles goed werkt. Is dit niet het geval zal de increment nogmaals worden doorlopen om zo eventueel requirements, ontwerpen en/of onderdelen aan te passen. Tevens zal het programma hierdoor in aparte onderdelen kunnen worden opgeleverd (iteratieve werkwijze), hierin verschilt RUP dan ook van de waterval methode waarbij alles in één keer wordt ontwikkeld.

Van de disciplines die te zien zijn in figuur 1, zijn een aantal disciplines niet of beperkt gebruikt namelijk de volgende:

- Business modeling, het doel van business modeling is om kennis op te doen van de business, de structuur en doelen van de betreffende organisatie, de problemen binnen de organisatie en mogelijke verbeteringen. Omdat de organisatie klein is en de aanleiding/probleemstelling vooral vanuit de klanten van Webtechniek komt zal deze discipline niet aan bod komen.
- Configuration & change management, wanneer de applicatie eenmaal is ontwikkeld zullen de verschillende wijzigingen en changes via een structurele aanpak worden uitgevoerd. Omdat de applicatie niet in gebruik zal worden genomen tijdens dit project komt deze discipline niet aan bod.
- Project management, omdat het project door één persoon wordt uitgevoerd komt deze discipline beperkt naar voren. Wanneer een grote groep mensen (10-20 man) moest worden aangestuurd was deze discipline zeer geschikt geweest, dit is echter niet het geval.
- Environment, dit is het voorbereiden, aanpassen en onderhouden van de ontwikkelomgeving tijdens het project. Omdat maar één persoon van deze omgeving gebruik maakt is deze discipline zo beperkt gebleven dat hij niet uitgebreid aan bod komt tijdens dit project.

Binnen RUP zijn nog meer mogelijkheden, het is bijvoorbeeld mogelijk om rollen en verantwoordelijkheden binnen een ontwikkelteam toe te wijzen. Omdat ik het project in mijn eentje zal uitvoeren is dit overbodig. Hierdoor heb ik alleen gebruik gemaakt van de fasering en de iteratieve/incrementele werkwijze van RUP.

4.2 Vaststellen Requirements

Tijdens de inception fase zijn de globale requirements onderzocht en achterhaald. In de elaboration fase zijn de meer gedetailleerde requirements vastgesteld en gecategoriseerd. Omdat al tijdens het vooronderzoek bleek dat ik met een groot aantal requirements te maken kreeg, heb ik via een structurele aanpak de requirements opgesteld. Hierbij heb ik gebruik gemaakt van de volgende methodes of technieken:

De Swart

Om een duidelijke verdeling binnen de requirements te creëren is er gebruik gemaakt van het boek [Handboek requirements door Nicole de Swart]. Hierin worden verschillende categorieën requirements genoemd. Zoals: functionele requirements, niet functionele requirements, bedrijfsregels en technische beperkingen. Door gebruik te maken van deze indeling worden de requirements opgedeeld in duidelijke categorieën die elk voor verschillende doeleinden worden gebruikt. Zo hebben bijvoorbeeld de technische beperkingen vooral invloed op de architectuur van dit systeem. Dit komt omdat ze de ruimte beperken waarbinnen de functionele requirements kunnen worden geïmplementeerd.

ISO 9126

Om de niet functionele requirements op te stellen is gebruik gemaakt van de ISO 9126 normen. Binnen de ISO 9126 normen zijn verschillende categorieën die samen de kwaliteit van een applicatie omschrijven, zoals de beveiliging of gebruiksvriendelijkheid van een applicatie. Er is gekozen voor de ISO 9126 norm omdat deze tijdens mijn studie is gebruikt en een standaard is.

MoSCoW

Nadat de requirements waren onderverdeeld in verschillende categorieën heb ik de requirements geprioriteerd door middel van een MoSCoW analyse. Het resultaat van de MoSCoW analyse was dat de bestaande requirements werden verdeeld in een viertal categorieën namelijk de must, should, could en won't have. Door deze verdeling kan worden gekeken welke requirements als eerst worden ontworpen, gebouwd en getest.

4.3 Ontwerpen software

Tijdens de elaboration fase heb ik verschillende technische ontwerpen gemaakt. Deze ontwerpen zijn opgesteld op basis van de ondervonden requirements.

UML

Om de ontwerpen te maken is er gebruik gemaakt van onder meer UML. UML staat voor Unified Modeling Language en is een modelleertaal om ontwerpen van informatiesystemen te kunnen maken. Binnen RUP wordt gebruik gemaakt van UML en zelf heb ik er veel ervaring mee. Binnen Webtechniek is dit ook een gebruikelijke manier van modelleren. Ik heb ervoor gekozen om drie soorten diagrammen te modelleren die gebruik maken van UML. Dit zijn de klassen, sequentie en activiteiten diagrammen. Er zijn meerdere soorten diagrammen binnen UML beschikbaar echter worden deze niet binnen Webtechniek gebruikt, hierdoor zag ik er geen meerwaarde in om deze toe te passen.

- Klassendiagrammen worden gebruikt om de statische structuur van de applicatie weer te geven. In een klassendiagram komen de verschillende verantwoordelijkheden naar voren, hun gedrag en onderlinge relaties.
- Om de interactie tussen verschillende objecten te modelleren is er gebruik gemaakt van sequencediagrammen.
- Tijdens het ontwikkelen van de applicatie is er gebruik gemaakt van activiteitendiagrammen om bijvoorbeeld de verloop van een algoritme weer te geven.

Tijdens het ontwerpen van de architectuur is er vooral gebruik gemaakt van diagrammen die geen officiële manier van modelleren hanteren. Deze diagrammen geven via een abstracte en overzichtelijke manier de architectuur weer, ze worden ook wel "boxes-and-lines" diagrammen genoemd omdat ze via enkel boxen en lijnen de functionele elementen weergeven en hun onderlinge associaties. De reden waarom ik voor deze manier van modelleren gebruik heb gemaakt is omdat bij het opstellen van de architectuur stakeholders binnen Webtechniek betrokken zijn. Om de diagrammen makkelijk te kunnen bespreken is deze manier van modelleren zeer geschikt.

4.4 Bouwen software

Tijdens de construction fase heb ik via de incrementele werkwijze van RUP de verschillende onderdelen van het totale systeem ontwikkeld. Voor het bouwen van deze onderdelen is er gebruik gemaakt van een aantal methodes en technieken:

ASP.NET

Tijdens het bouwen van de software onderdelen is er gebruik gemaakt van ASP.NET, dit is een framework ontwikkeld door Microsoft. Binnen ASP.NET is er gebruik gemaakt van de programmeertaal C#. Bij Webtechniek wordt er ook gebruik gemaakt van dit framework dus was het logisch om ook hiermee te werken.

OTAP

Tijdens het bouwen van de software is gebruik gemaakt van een OTAP omgeving, OTAP staat voor ontwikkel, test, acceptatie en productie omgeving. Deze verschillen alle van elkaar qua toegankelijkheid; wat betreft architectuur lijken zij wel op elkaar. Er is gekozen om met OTAP te werken om via deze manier omgevingen op te stellen die gebruikt kunnen worden voor één doeleind zoals testen. Hierdoor wordt er voorkomen dat alle

activiteiten zoals ontwikkelen en testen plaats vinden op één omgeving. Omdat een deel van de OTAP omgeving er nog niet was, is tijdens het opstarten van dit project de OTAP omgeving opgezet.

Versiebeheertool

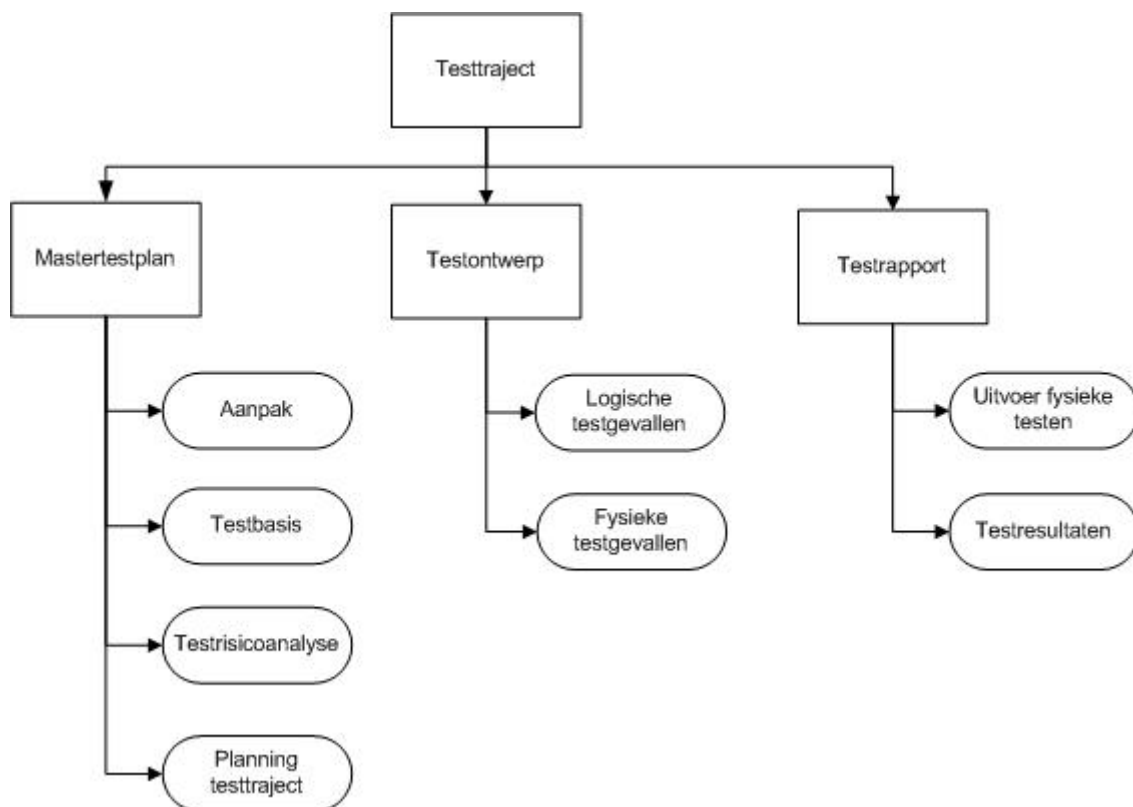
Tijdens het bouwen van de software is gebruik gemaakt van een versiebeheertool, te weten TortoiseSVN. Hierdoor werden er verschillende versies van de applicatie en documentatie bij gehouden. Binnen Webtechniek wordt er ook gebruik gemaakt van deze tool.

4.5 Testen software

Om de kwaliteit te waarborgen van de documenten en de applicatie die ik tijdens het project opstel heb ik gedurende het project verschillende tests uitvoert. Deze testen zijn afhankelijk per product. Voor het opstellen van de verschillende testen is gebruik gemaakt van Testgoal. Testgoal is een testfilosofie die resultaatgedreven testen als aanpak heeft en is hierdoor dus meer een richtlijn in plaats van een methodiek. In eerste instantie wou ik gebruik maken van TMap maar dit bleek een te uitgebreide testmethodiek voor mijn project.

4.5.1 Testtraject

Een schematische weergave van het testtraject is te zien in figuur 3.



Figuur 3 testtraject

Zoals te zien in figuur 3 zijn er een beperkt aantal onderdelen gekozen uit het algemene stappenplan (zoals die wordt beschreven in testgoal). Er zijn een aantal redenen waarom binnen het testtraject een beperkt aantal onderdelen terugkomen:

- Het project wordt ontwikkeld door één persoon, hierdoor is het opstellen van een organisatie qua rollen en verantwoordelijkheden niet nodig.
- Omdat normaliter het testtraject per testsoort wordt gehouden (detailtestplan) is er in dit geval voor gekozen om dit op een meer strategisch niveau te behandelen oftewel over meer dan één testsoort. Dit plan wordt een mastertestplan genoemd.
- Het inrichten van een testomgeving is niet nodig omdat er al gebruik wordt gemaakt van OTAP. Hierin wordt al een testomgeving ingericht.

Mastertestplan

Voordat de testen kunnen worden uitgevoerd zal er een mastertestplan worden opgezet. In dit plan komen een aantal onderdelen aan bod die er voor zorgen dat het testen op een correcte manier verloopt. Als eerst zal de aanpak kort worden beschreven. Vervolgens zal de testbasis in kaart worden gebracht voor de verschillende testsoorten. Om de functionaliteiten te prioriteren wordt er een TRA (testrisicoanalyse) opgesteld (wat wordt als eerst getest en hoe uitgebreid?). Als laatste zal er een planning worden toegevoegd van het gehele test traject.

Testontwerp

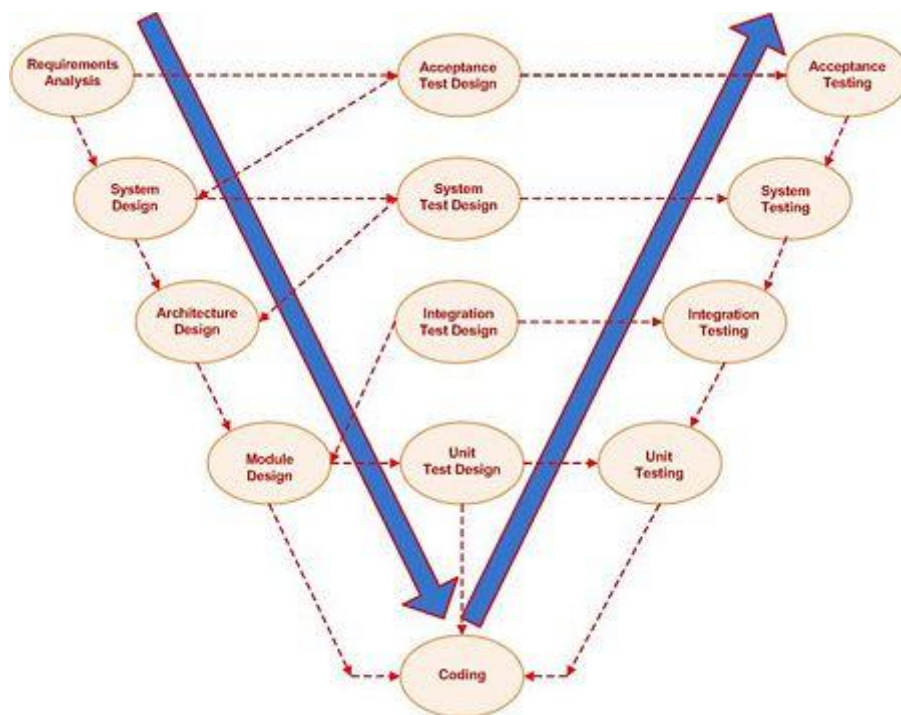
Het testontwerp zal bestaan uit logische en fysieke testgevallen, die verschillen uiteraard per testgeval. Een logisch testgeval wordt gebaseerd op de testbasis, het beoogde resultaat en de TRA. De fysieke testgevallen worden opgesteld op basis van de logische testgevallen, het doel van de fysieke testgevallen is om de testen concreet uit te werken, dit wil zeggen dat er daadwerkelijke invoerwaardes worden opgesteld.

Testrapport

Wanneer de testen zijn uitgevoerd zal een testrapport ontstaan waarin de resultaten en eventuele vervolg acties worden beschreven.

4.5.2 V-model

Om te bekijken welke producten per testsoort als testbasis dienen is er gebruik gemaakt van het V-model. In een V-model worden de producten die zijn gemaakt tijdens een project opgesomd om vervolgens per product de bijhorende testsoort te definiëren. Omdat ik binnen mijn project maar een beperkt aantal testsoorten zal gebruiken zal ik als eerst de testsoorten beschrijven uit het V-model die ik ga gebruiken. Om vervolgens kort uit te leggen welke testsoorten ik niet ga toepassen en waarom.



Figuur 4 V-model

Module (Unit testing)

Bij moduletesten worden individuele componenten getest. Als basis voor de moduletesten wordt de code gebruikt die is gemaakt op basis van verschillende ontwerpen. Tijdens het programmeren van de code zullen de moduletesten continue worden uitgevoerd. De testen worden op een informele manier uitgevoerd dit wil zeggen dat de programmeur zelf de code test op veel voorkomende fouten.

Systeem test (System testing)

Wanneer het volledige systeem is gebouwd is het tijd om de systeemtest uit te voeren. Dit is niet alleen een test om de verschillende componenten te testen maar ook het volledige systeem. In deze test kunnen ook de niet functionele requirements aan bod komen zoals beveiliging en gebruiksvriendelijkheid.

Reviews & feedback momenten

Hoewel dit niet een testsoort is en het niet in het V-model staat, vormen de review & feedback momenten een belangrijk onderdeel tijdens het project. De documenten die worden opgesteld tijdens het project worden continue gecontroleerd door middel van reviews. De review momenten zullen alleen worden gehouden met de stakeholders binnen Webtechniek om via deze manier de kwaliteit van de documenten te testen en eventuele op en aanmerkingen te kunnen verwerken. Hierdoor zal de kwaliteit van de documenten verbeterd worden.

Niet gebruikte testsoorten

De twee overige testsoorten de integratietest en de gebruikersacceptatietest, worden niet uitgevoerd tijdens dit project. Ik zal per testsoort beargumenteren waarom:

- Tijdens de integratietest worden de verschillende modules binnen het systeem getest qua samenwerking en of deze correct verloopt. Omdat er niet veel tijd beschikbaar is voor het testen moest er een keuze worden gemaakt tussen de systeemtest of integratietest. Het leek mij verstandiger om de systeemtest uit te voeren, omdat hierbij de nadruk op de functionaliteiten komt te liggen, tevens zullen de moduletesten ook de samenwerking tussen bepaalde modules testen.
- De gebruikersacceptatietest wordt niet gehouden omdat de applicatie tijdens de afstudeerperiode nog niet wordt geïmplementeerd, hierdoor is het eindresultaat een prototype dat nog niet in gebruik is genomen.

4.6 Beroepstaken

Tijdens de afstudeeropdracht heb ik op een aantal beroepstaken de nadruk gelegd. Via deze manier kan worden aangetoond dat ik in staat ben om een complexe opdracht zelfstandig uit te voeren. Ik heb er voor gekozen om een viertal beroepstaken te kiezen die het meest aansluiten bij de opdracht.

1.4 Uitvoeren van analyse door definitie requirements, niveau 4

Het enquêtesysteem moet aansluiten bij de eisen en wensen van een grote groep stakeholders. Deze groep stakeholders varieert van de klanten van Webtechniek tot de verschillende medewerkers (functies) binnen het bedrijf. Hieruit zullen een groot aantal requirements naar voren komen, waarbij tegengestelde requirements kunnen voorkomen.

3.2 Ontwerpen systeemdeel, niveau 3

Het enquêtesysteem zal verschillende functionaliteiten bevatten waarbij complexe algoritme en design patterns zullen worden toegepast. Verder zullen niet functionele requirements ook een rol spelen zoals: beveiliging, gebruiksvriendelijk en compatibiliteit.

3.3 Bouwen applicatie, niveau 3

De applicatie zal gebruik maken van een bestaand framework namelijk MVC of Web-forms (de twee standaarden die gebruikt worden binnen Webtechniek voor ASP.NET). De applicatie zal op een objectgeoriënteerde manier worden ontwikkeld. Verder wordt er rekening gehouden met toekomstige wijzigingen, testbaarheid en hergebruik. Het bouwen gebeurt in een geavanceerde ontwikkelomgeving inclusief versiebeheertool.

3.5 Uitvoeren van en rapporteren over het testproces, niveau 3

Omdat Webtechniek hoge eisen stelt aan de kwaliteit van hun producten zal testen een belangrijk onderdeel vormen binnen het project. De testen zullen worden uitgevoerd door middel van verschillende testsoorten zoals: een systeemtest en een moduletest.

5 Opstart van het project

Het project is gestart met de eerste fase van RUP, de inception fase. In deze fase wilde ik vooral helder krijgen wat de opdracht inhield en wat er moest gebeuren. Door dit in een vroeg stadium uit te voeren kan er een overzichtelijk planning worden gemaakt. In de volgende paragrafen zal ik toelichten wat ik precies heb gedaan tijdens de inception fase.

5.1 Plan van aanpak

Binnen de inception fase ben ik begonnen om eerst het plan van aanpak op te stellen. Het plan van aanpak was vooral bedoeld voor mijn opdrachtgever bij Webtechniek, Maurice Jongmans, en voor mijzelf als opdrachtnemer. Het plan van aanpak dient als richtlijn en als planning bij mijn opdracht; ik leg hierin zowel de opdracht als de aanpak vast.

De basis voor het plan van aanpak was al gelegd in het afstudeerplan, dit is vooral terug te zien in de opdrachtoomschrijving binnen het plan van aanpak. Echter zijn er een aantal nieuwe onderdelen toegevoegd of uitgebreid zoals:

- Een omschrijving van de doelgroep en de stakeholders
In dit hoofdstuk is de doelgroep omschreven, dit zal de groep zijn waar het project op gericht is en die uiteindelijk het resultaat zal gebruiken. Naast de doelgroep zijn ook de stakeholders omschreven die zijn betrokken bij het project.

Doelgroep

De doelgroep van deze opdracht komt voort uit de opdrachtoomschrijving namelijk “de klanten van Webtechniek”. Om dit begrip specifiekere te maken heb ik de volgende citaat gebruikt [Webtechniek2012]: *“Wij, Maurice en Wouter, eigenaren en oprichters, willen het MKB een comfortabel en betaalbaar webgebruik bieden. Want als harde werkers en gedreven ondernemers weten we dat u zich uitsluitend met uw eigen vakgebied bezig moet houden.”*

In de hier bovenstaande citaat, komt het woord MKB naar voren wat een afkorting is voor [Economische begrippen2012]:

“Midden- en kleinbedrijf”

Om het begrip Midden – en kleinbedrijf verder te definiëren is gebruik gemaakt van de volgende citaat [ru.nl2012]:

“Tot het MKB behoren de kleine en middelgrote bedrijven. Het 'kleinbedrijf' bestaat uit de ondernemingen met nul tot vijf werknemers; circa 550.000 bedrijven behoren daartoe (85% van alle bedrijven). Het 'middenbedrijf' bestaat uit ondernemingen met zes tot tweehonderd werknemers; circa 100.000 bedrijven (ongeveer 14%). Het 'grootbedrijf' bestaat uit ondernemingen met meer dan tweehonderd werknemers; circa 2000 bedrijven (minder dan 1%).”

Webtechniek is gevestigd in Delft en heeft hierdoor vooral klanten in de omgeving van Delft (regio Zuid-Holland), uiteraard zijn er ook klanten in de rest van Nederland maar hier ligt de focus niet op. Je kan dus stellen dat Webtechniek gefocust is op MKB'ers in de provincie Zuid-Holland.

Stakeholders

Persoon	Rol
Maurice Jongmans	Opdrachtgever en commercieel directeur
Wouter Olde Weghuis	Technische directeur
Robin Brusse	Project manager
Overige medewerkers Webtechniek	-
Klanten Webtechniek	Doelgroep
Mark Suurland	Opdrachtnemer
Lousberg-Orbons, A.M.J.J.	Begeleidend examinerator
Paul Breukel	Tweede examinerator

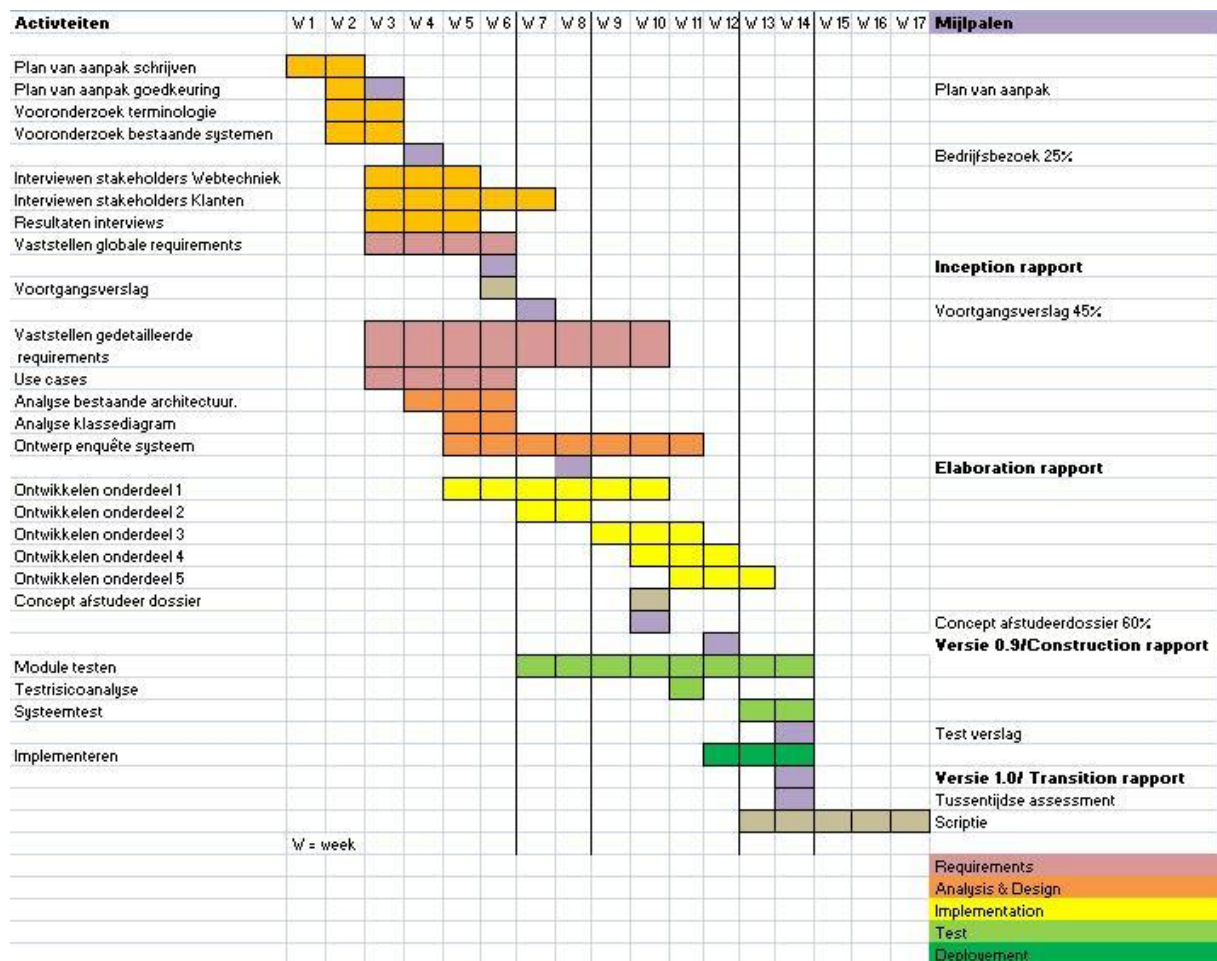
Hoewel de overige medewerkers van Webtechniek niet een specifieke rol hebben binnen het project. Zullen zij toch indirect betrokken zijn bij mijn project. Wanneer ik bijvoorbeeld technische vragen heb, kan ik bij hun terecht.

- Projectorganisatie

In dit hoofdstuk is de projectorganisatie in kaart gebracht, om dit op een juiste manier te doen is beschreven welke ontwikkelmethode er wordt toegepast binnen dit project en op welke manier. Vervolgens zijn alle betrokken personen beschreven en hun verantwoordelijkheden. Als laatst zijn er duidelijke afspraken gemaakt op welke manier er wordt gerapporteerd en naar wie.

- Aanpak en planning

In dit hoofdstuk is de aanpak besproken door middel van een planning, de bijhorende mijlpalen, de op te leveren producten en als laatst een procesdiagram waarin de producten en tussenliggende processen schematisch worden weergegeven. In figuur 5 is de eerste versie van de globale planning te zien.



Figuur 5 globale planning project

Zoals te zien is in figuur 5 is voor opstellen van de planning gebruik gemaakt van de disciplines en fases binnen RUP. Elke fase eindigt met een rapport. Binnen de fases komen verschillende disciplines terug van RUP waarbij bepaalde activiteiten horen. De disciplines zijn met verschillende kleuren weergegeven, de disciplines die terug komen in de planning zijn: requirements, analysis & design, implementation, test en deployment.

Tevens is de incrementele en iteratieve werkwijze terug te zien in de planning, de gehele applicatie is opgedeeld in verschillende onderdelen (onderdeel 1 tot en met 5) en worden in delen toegevoegd aan de applicatie. Als tijdens het project blijkt dat aanpassingen aan de planning nodig zijn, zal dit worden gedaan.

Het plan van aanpak is vervolgens opgeleverd aan de opdrachtgever om zo tot een overeenkomst te komen over de opdracht.

Ter inzage is het plan van aanpak te zien in bijlage B.

5.2 Vooronderzoek

Tijdens en na het opstellen van het plan van aanpak heb ik een vooronderzoek gehouden. Het doel van het vooronderzoek was om mijzelf in te lezen in het hoofdonderwerp van deze opdracht (enquête) en te bekijken wat voor functionaliteiten bestaande enquêtesystemen hebben. Het vooronderzoek bestaat hierdoor uit twee onderdelen namelijk uit een literatuur en uit een marktonderzoek.

Onderdeel 1: Terminologie

Om een duidelijke lijst met terminologie op te stellen die ondersteuning kon bieden tijdens het marktonderzoek en het ontwikkelen van het enquêtesysteem, is er een literatuuronderzoek gehouden met betrekking tot terminologie. Dit wordt beschreven in het eerste onderdeel van het vooronderzoek.

Dit onderdeel van het vooronderzoek heeft één week geduurd, dit is de helft van de totale tijd die ik heb ingepland voor het volledige vooronderzoek. Om gebruik te maken van betrouwbare bronnen heb ik twee boeken geraadpleegd in de bibliotheek van de Haagse hogeschool die als onderwerp "enquête" hadden. Het eerste boek (Enquête research, Ontwikkelen van vragenlijsten en steekproeven, 2007) heb ik gebruikt om alle termen te verzamelen en het tweede boek (Enquêteren, Het opstellen en gebruiken van vragenlijsten, 2004) ter controle (met de vraag: misten er nog termen?). Door deze aanpak was ik er zeker van dat de lijst compleet was.

Om een voorbeeld te geven van de in totaal zesentwintig begrippen is figuur 6 toegevoegd.

Steekproef: "Meestal moet je genoeg nemen met een deel van de mensen uit de populatie die je wilt onderzoeken. De groep mensen die je geselecteerd hebt om te enquêteren, noem je steekproef"

Representativiteit: "In je populatieomschrijving heb je zo precies mogelijk aangegeven welke groep mensen of objecten je gaat onderzoeken. Wanneer je een steekproef trek, kies je een (klein) gedeelte van de populatie en alleen bij dat gedeelte doe je onderzoek. Binnen de populatie kunnen er deelpopulaties, *strata*, zijn die elkaar verschillen. Je moet ervoor zorgen dat alle *strata* in de populatie in dezelfde verhouding aanwezig zijn in de steekproef zodat de samenstelling van de steekproef een precieze afspiegeling vormt van de samenstelling van de populatie"

Aselect: "Elk element van de populatie heeft eenzelfde kans om in de steekproef te worden getrokken"

Niet-aselecte steekproeven: "Je kunt dan niet zeggen dat alle eenheden in de populatie dezelfde kans hebben gehad om in de steekproef terecht te komen." (Oftewel wanneer niet alle eenheden gelijk zijn of gelijk aan de populatie).

Respons: "Het aantal elementen van de steekproef waarvan volledige informatie is verkregen"

Non-respons: "Ook al heb je een mooi steekproefkader, dan nog is het mogelijk dat bepaalde groepen in de populatie geen antwoord willen of kunnen geven. Dit heten zogenaamde *non-respons*"

Figuur 6 Voorbeeld terminologie

Onderdeel 2: Marktonderzoek

In het tweede deel van het vooronderzoek is een marktonderzoek gehouden naar bestaande enquêtesystemen om te bekijken welke functionaliteiten zij aanbieden. Omdat het aantal enquêtesysteem groot is zal ik een aantal selectiecriteria opstellen die er voor zorgen dat de juiste enquêtesystemen worden onderzocht. Om de objectiviteit te waarborgen zullen zowel nationale als internationale enquêtesystemen worden onderzocht. Tijdens het marktonderzoek zijn de volgende selectiecriteria gebruikt:

- Alleen web-based enquêtesystemen.
- Alleen systemen met een beschrijving van hun functionaliteiten.

Om de enquêtesystemen te kunnen vinden is er gezocht binnen een "searchengine" (google in dit geval) met een aantal termen die de zelfde betekenis hebben als "enquêtesysteem", dit zijn de volgende termen:

- Enquête tool
- Enquête systeem
- Survey tool

Om structuur te houden binnen de verschillende functionaliteiten zijn de categorieën uit de opdrachtomschrijving gebruikt, deze bleken echter niet voldoende. Hierdoor zijn de volgende categorieën ontstaan.

- Creëren, alle functionaliteiten die betrekking hebben op het creëren van enquêtes.
- Lay-out, alle functionaliteiten die betrekking hebben op de opmaak van de enquêtes.
- Media bibliotheek, alle functionaliteiten die betrekking hebben op het beheren van media.
- Distribueren, alle functionaliteiten die betrekking hebben op het distribueren van enquêtes.
- Resultaten, alle functionaliteiten die betrekking hebben op de resultaten van enquêtes.
- Account, alle functionaliteiten die betrekking hebben op accounts binnen een enquêtesysteem.

Omdat niet alle enquêtesystemen kunnen worden onderzocht die momenteel op de markt beschikbaar zijn, is er twee weken de tijd genomen om hierin een deel van het totale marktaanbod te onderzoeken. Uiteindelijk bleek binnen de geplande tijd dertien verschillende enquêtesystemen te zijn onderzocht. Omdat de kans bestaat dat er functionaliteiten niet zijn ondervonden zal ik interviews met de betrokken stakeholders houden om eventueel nieuwe functionaliteiten toe te voegen. Het doel van dit deel van het vooronderzoek is dan ook om een goede basis te leggen voor de requirements.

Het resultaat van het marktonderzoek is een lijst met functionaliteiten. Deze lijst diende als basis om de requirements op te stellen. Omdat de lijst een groot aantal functionaliteiten bevat zal ik alleen een aantal functionaliteiten van het onderdeel creëren van enquêtes tonen:

#	Function aliteit	Categorie	Toelichting
1	Editor	Creëren	De tekst die wordt ingevoerd in de enquête kan worden bewerkt, images kunnen worden toegevoegd.
2	Aanmake n enquête	Creëren	Bij het aanmaken van een enquête zijn twee opties mogelijk namelijk de volgende: -Vanaf nul beginnen (niks) -Een aantal voorgemaakte templates (met vragen etc.)
3	Talen	Creëren	Bij het aanmaken kan een taal worden gekozen waarin de enquête wordt gemaakt, meerdere talen is ook mogelijk.
4	Titel	Creëren	Bij het aanmaken moet er een titel worden gegeven aan de enquête, wanneer er meerdere talen zijn geselecteerd word er per taal een titel gegeven.
5	Start page	Creëren	Dit is de eerste pagina de respondent te zien krijgt, door middel van de editor (zie functionaliteit 1) kan de tekst worden aangepast.
6	Vragen aanmake n	Creëren	Vraagtypes, een overzicht met alle vraagtypes word getoond (zie functionaliteiten 8 t/m 12). Wanneer een vraag word aangemaakt worden de volgende onderdelen gevraagd: - Verplicht (ja/nee) - Hide question (ja/nee) - Nieuwe pagina (ja/nee)
7	Aanpasse n van de vragen	Creëren	De aangemaakte vragen kunnen worden bewerkt, zo is er een overzicht die vertakking van alle vragen laat zien.
8	Vraagsoo rten		Meerkeuze: - Keuzerondjes (radiobutton) - Selectievakjes (checkbox) - Vervolgkeuzelijst (dropdownlist) - Keuzelijst (listbox)
9	""		Tabellen: - Matrix (radiobutton) - Matrix (checkbox) - Matrix (textbox) - Matrix (eindpunten)

10	""		Waardering & rangschikking <ul style="list-style-type: none"> - Waarderingschaal - Constante som (bijvoorbeeld 100 punten te verdelen over 3 antwoorden) - Rangorde schaal - Net promoter score (1 – 10)
11	""		Opentekst velden <ul style="list-style-type: none"> - één tekstveld - Meerdere tekstvelden - één tekstveld met meerdere regels
15	Conditio nele vertakkin g	Creëren	Verschillende antwoorden van een vraag kunnen naar verschillende vragen wijzen.
13	Afsluiting	Creëren	De afsluiting van de enquête, door middel van een dank woordje naar de respondent. Deze tekst kan worden aangepast door de editor.

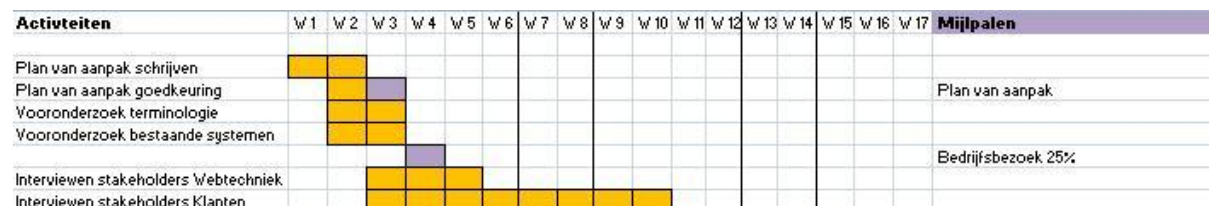
5.3 Interviews Webtechniek stakeholders & klanten stakeholders

Webtechniek stakeholders

In het plan van aanpak waren de betrokken personen en stakeholders gedefinieerd. Omdat de groep stakeholders binnen Webtechniek snel beschikbaar konden zijn, heb ik een tweetal interviews gehouden. Het eerste interview had als doel om de geïnterviewde met "eigen" functionaliteiten te laten komen, het interview heeft een aantal topics als stuurpunt maar is verder vooral een open gesprek. Het tweede interview had als doel om de onderzochte functionaliteiten uit het vooronderzoek te prioriteren van hoog naar laag, de geïnterviewde kon per functionaliteit een punt van 1 tot 10 geven. In totaal zijn drie personen binnen Webtechniek geïnterviewd namelijk: Maurice Jongmans, Wouter Olde Weghuis en Robin Brusse. De keuze voor deze personen is genomen omdat zij verantwoordelijk zijn voor projectbegeleiding of hiermee ervaring hebben. De overige medewerkers binnen Webtechniek hebben dit niet of waren niet beschikbaar voor interviews. De resultaten van beiden interviews zijn te zien in §5.3.1 en §5.3.2.

Klanten stakeholders

Mijn streven was om de interviews met de klanten in de inception fase te houden echter bleek al snel dat hier veel meer tijd voor nodig was. Er moest namelijk eerst een lijst met klanten worden opgesteld die volgens de opdrachtgever geschikt waren om te worden geïnterviewd. Hiervoor moesten de klanten worden benaderd en moest een geschikt interview worden opgesteld. Omdat de klanten niet snel reageerden en er veel terugkoppeling met de opdrachtgever nodig was voor het opstellen van het interview heb ik er voor gekozen om het te verschuiven naar de construction fase. De aanpassing binnen de planning is te zien in figuur 7.



Figuur 7 uitstel interviews klanten

5.3.1 Resultaten interview 1

Vraag	Robin Brusse	Maurice Jongmans	Wouter Olde Weghuis
Kunt u zichzelf even voorstellen?	Ik ben Robin Brusse, team manager bij Webtechniek.	Ik ben Maurice Jongmans, commercieel directeur bij Webtechniek.	Ik ben Wouter Olde Weghuis, technische directeur bij Webtechniek.
Heeft u al eens met een enquête systeem gewerkt?	Nee	Ja, ik werk regelmatig met enquêtesystemen.	Ja wel eens door eentje heen geklikt maar nooit mee gewerkt.

Welke hoofd onderdelen moet het enquête systeem volgens u hebben?	- Beheren van enquêtes - Beheren Layouts - Verspreiden van enquêtes - Resultaten van enquêtes	- Creëren enquêtes - Resultaten enquêtes - Distribueren enquêtes	- Aanmaken van enquêtes - Beheren van enquêtes - Layouts aanmaken - Verspreiden van enquêtes - Resultaten van enquêtes
Kan u wat dieper ingaan op deze hoofd onderdelen?	De onderdelen van een enquête (bij het aanmaken) in een logische stappenplan, eerst inleiding, header en de footer, de vragen, de afsluiting. Export mogelijkheden in SPSS en Excel.	Tijdens het creëren van enquêtes moeten het vooral mogelijk zijn om uit een ruime keuze van vraagsoorten te kunnen kiezen. Resultaten tonen door middel van duidelijke grafieken. De mogelijkheden voor het distribueren van enquêtes moet uitgebreid zijn oftewel via een link, social media of iframe.	Inleiding enquête. Waardering schaal (vraag) moeten flexibel zijn, bijvoorbeeld 1 t/m 10 of 1 t/m 3. Resultaten terug naar respondent via mail. Lay out heel flexibel en specifiek zijn, alle onderdelen van een enquête moeten worden "gestyle". SPSS export indien mogelijk.

Resultaten interview 1

Naar aanleiding van de resultaten uit het open interview zijn er een aantal requirements toegevoegd om een tweetal voorbeelden te noemen:

- De klant wil in het vragentype waarderingsschaal kiezen hoeveel antwoorden hij heeft, bijvoorbeeld: 1 t/m 10 of 1 t/m 3.
- De klant wil door middel van een duidelijk stappenplan de enquête, groep en layout opstellen.

5.3.2 Resultaten interview 2

De resultaten uit het tweede interview zijn verwerkt, de punten zijn bij elkaar opgeteld en op basis hiervan zijn de categorieën vanuit MoSCoW toegekend.

Categorieën:

1. Won't have, 0.0 - 7.5 punten
2. Could have, 7.5 – 15 punten
3. Should have, 15 - 22.5 punten
4. Must have, 22.5 – 30 punten

Deze verdeling zal worden gebruikt om de requirements te prioriteren.

Functionaliteit nummer	Maurice	Robin	Wouter	Totale punten	MoSCoW categorie
1	8	7	7	22	3
2	7,5	4	10	21,5	3
3	5	8	6	19	3
4	10	8	7	25	4
5	4	7	8	19	3
6	8	8	10	26	4
7	8	7	10	25	3
8	10	10	10	30	4
9	10	10	6	26	4
10	10	10	7	27	4
11	10	10	7	27	4
12, instructie	7,5	7	6	20,5	3
12, bestanden uploaden	7,5	6	6	19,5	3
13	8	7	6	21	3
13b	8	8	9	25	4
14	6	6	5	17	3
14b	2	4	7	13	2
15	9	9	9	27	4
16	4	4	7	15	2
17 mail (api mailtech)	9	8	9	26	4
17 papier	2	3	3	8	2
17 social	9	8	8	25	4
17 web link	9	8	8	25	4
17 web embed	9	7	8	24	4
17 popup	7	7	8	22	3
17 sms	4	4	3	11	2
18	7,5	7	10	24,5	4
19	9	7	8	24	4
20	2	8	8	18	3
21 (pdf, Excel, SPSS)	6	7	7	20	3
22 (18 dashboard)	5	3	5	13	2
23	10	10	10	30	4
24	8	8	8	24	4
25 a plaatsing textbox	8	3	4	15	2
25 b meerdere openvrag	9	3	5	17	3
26	2	5	9	16	3
27 anders...	6	7	10	23	4
28	6	6	6	18	3
29 Advies?	6	3	8	17	3
31	10	10	9	29	4
32 tijd bijhouden	6	5	10	21	3
33	6	6	5	17	3
34	2	3	6	11	2
35 aanvulling op 13	7	7	6	20	3
36	4	6	9	19	2
37	4	5	6	15	2
38 terug ja/nee	4	6	8	18	3
38 pauze	2	6	4	12	2
38 redirect (url)	8	6	5	19	3
39	8	8	5	21	3
40 (zie 21)	5	5	4	14	2
41*	10	10	4	24	4
42*	2	3	5	10	2
43	2,5	9(zelf kiezen) 3(random)	8	10,5	2
44 (13)	2,5	6	8	16,5	3
45 (13)	8	7	9	24	4
46	6	8	9	23	4
47 standaard layouts	7	5	7	19	3
50 help/handleiding	2	6	8	16	3
51	2	2	3	7	1

Figuur 8 Resultaten interview 2

Opmerkingen of verwijdering van functionaliteiten

Tijdens het houden van dit interview zijn er een aantal functionaliteiten afgevallen of aangepast namelijk de volgende:

#	Functionaliteit	Reden	
30	Bij het aanmaken van een enquête kan worden gekozen om standaard templates te kiezen zoals: event evaluatie, website evaluatie, imago, marktpositie/potentie, naamsbekendheid etc. Deze zijn al ingevuld met vragen en bijhorende antwoorden.	Teveel werk voor de eerste versie van het enquêtesysteem. De overige functionaliteiten zijn belangrijker.	Verwijderd
41	Registreren van een account door een gebruikersnaam op te geven, bedrijf, wachtwoord en email.	Webtechniek (Maurice) wil dat dit door Webtechniek gebeurt en niet door de gebruiker zelf.	Opmerking
42	Dit is een functie bij het aanmaken van een enquête, de vragenbank is een verzameling	Dit slaat terug op de voorgemaakte templates: oftewel de vragen uit de	Opmerking

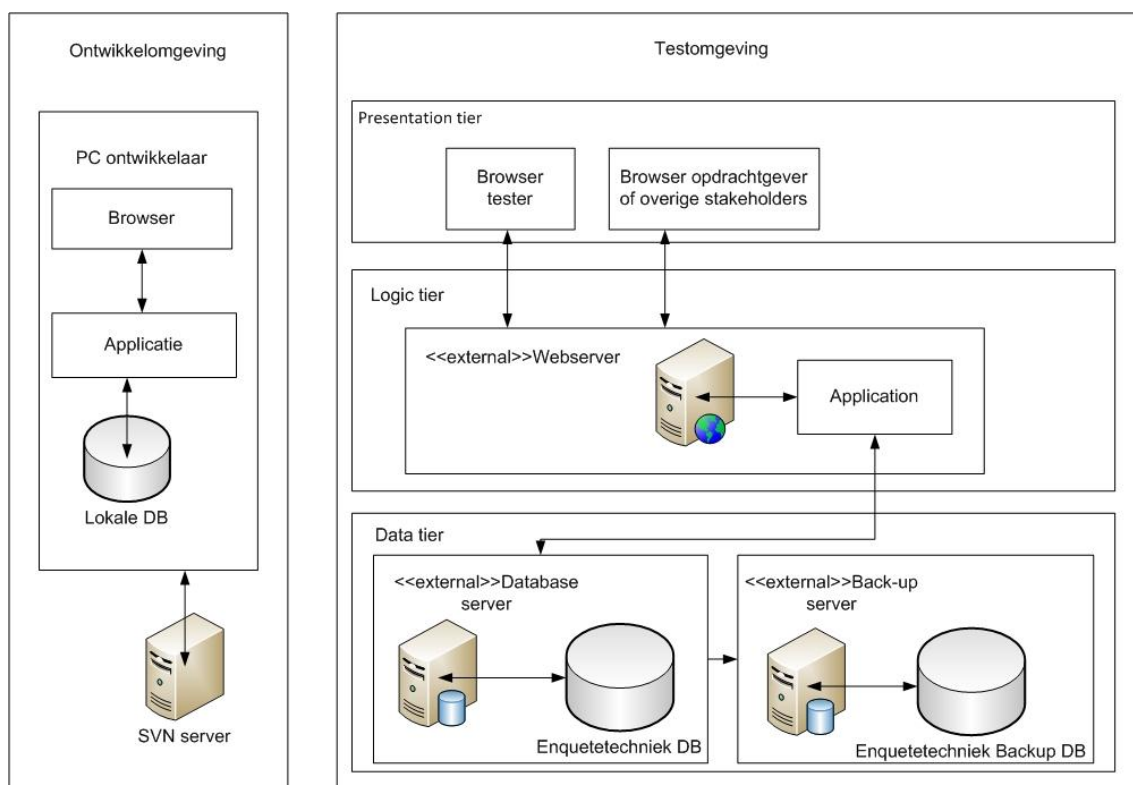
	van vragen die zijn opgesteld door specialisten waardoor men deze eerder opgestelde vragen gewoon kan invoeren in hun eigen enquête.	templates bewaren.	
48	De enquête kan worden uitgeprint in pdf/word.	Het is een web-based applicatie hierdoor ligt de nadruk op het verspreiden via het web en niet via papier.	Verwijderd
49	De verwijderen vragen kunnen worden herstelt. Ze worden namelijk bijgehouden.	Wanneer een vraag wordt verwijderd is die definitief weg, herstellen is niet nodig.	Verwijderd
52	De enquête kan op 3 manieren worden worden gepop-up't. Namelijk door middel van een embedded popup, een feedback tab of een normale popup.	Één manier van pop-up is genoeg. Tevens is de term "feedback tab" onduidelijk.	Verwijderd

5.4 Aanpassing planning: globale requirements

In de globale planning (te zien in figuur 5) van de inception fase is opgenomen dat er globale requirements tijdens deze fase worden opgesteld. Hoewel na het houden van het vooronderzoek en de eerste interviews er is begonnen met het opstellen van de requirements, waren er nog geen concrete requirements aan het einde van de inception fase. Hierdoor heb ik er voor gekozen om alle requirements op te stellen tijdens de elaboration fase. Als gevolg van deze keuze is het onderdeel "globale requirements" vervallen uit de planning.

5.5 OTAP

Tijdens het plan van aanpak is er gekozen om volgens OTAP te werken. Van alle omgevingen binnen OTAP was de ontwikkelomgeving nog niet aanwezig, de overige omgevingen waren al aanwezig in de huidige situatie. In figuur 9 zijn de ontwikkel en testomgeving te zien.



Figuur 9 Ontwikkel en testomgeving van OTAP

Ontwikkelomgeving

De ontwikkelomgeving draait lokaal bij de ontwikkelaar, er wordt gebruik gemaakt van een laptop waarop de applicatie en een lokale database staan. Als versiebeheertool zal gebruik worden gemaakt van SVN, binnen Webtechniek is een SVN server aanwezig.

Testomgeving

Wanneer een onderdeel is ontwikkeld binnen enquêteteknik zal deze worden getest in een omgeving die gelijk is aan de productieomgeving. De webserver zal op een externe locatie staan hetzelfde geldt voor de database server. Om de gegevens uit de database te allen tijde beschikbaar te houden zullen meerdere keren per dag back-ups worden gemaakt op een back-up server.

Acceptatie en productieomgeving

Wanneer alle onderdelen zijn getest zal het systeem naar de acceptatieomgeving worden gekopieerd, deze omgeving ziet er qua architectuur hetzelfde uit als de testomgeving. Het zal echter op een andere instantie staan. De acceptatieomgeving is voor een beperkte groep stakeholders beschikbaar, namelijk alleen de stakeholders binnen Webtechniek. Wanneer de applicatie klaar is voor gebruik zal hij toegankelijk worden voor alle stakeholders, dit is de productieomgeving.

5.6 Inception rapport

Om de eerste fase af te ronden, werd een inception rapport opgesteld. Dit rapport was de uitkomst van de werkzaamheden die in de eerste weken waren gedaan. Het doel hiervan was om de betrokkenen op de hoogte te houden van mijn werkzaamheden, en te laten zien wat de mogelijke ideeën waren.

Ter inzage is het inception rapport toegevoegd in Bijlage C.

6 Requirements en ontwerp van enquête-techniek

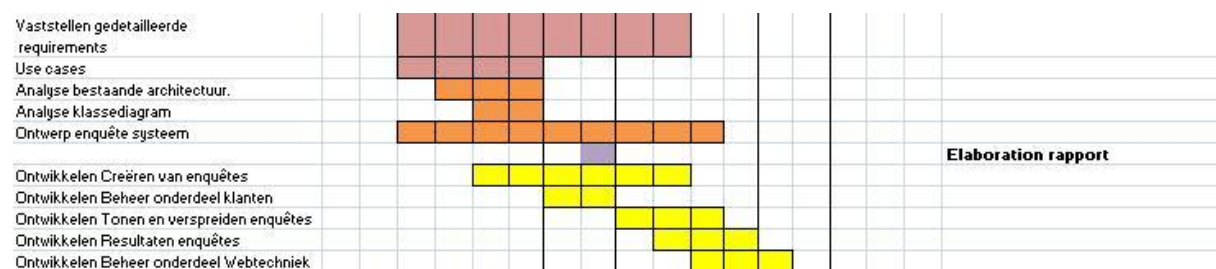
De tweede fase binnen RUP is de elaboration fase. In deze fase heb ik de gedetailleerde requirements van enquête-techniek opgesteld om vervolgens hieruit de architectuur en overige ontwerpen op te stellen. In de volgende paragrafen zal ik toelichten wat ik precies heb gedaan tijdens de elaboration fase.

6.1 Gedetailleerde requirements

Ik heb de requirements uit de beschikbare bronnen verzameld (vooronderzoek, interviews en opdrachtomschrijving). Nadat de lijst met requirements zich steeds meer aanvulde werd het al snel duidelijk dat de lijst met requirements heel groot was. Om onderscheid tussen de verschillende “soorten” requirements te maken is de verdeling uit het boek de Swart gebruikt. Hierdoor ontstonden de volgende categorieën: functionele requirements, niet functionele requirements, technische beperkingen, ontwerpbeslissingen en bedrijfsregels. Om te bekijken welke requirements belangrijker worden bevonden door de stakeholders heb ik een MoSCoW analyse toegepast op de functionele requirements. In deze paragraaf zal ik toelichten hoe het opstellen van de requirements tot stand is gekomen.

6.1.1 Aanpassing planning

Nadat duidelijk werd dat de lijst met requirements toenam qua grote is er voor gekozen (met overeenstemming van de opdrachtgever) om alle onderdelen te ontwerpen en vervolgens ze één voor één te bouwen (uiteeraard kan tijdens het ontwerpen al worden begonnen met het ontwikkelen). De belangrijkste reden hiervoor is dat mogelijk bepaalde onderdelen niet worden geïmplementeerd, de opdrachtgever wil echter naslagwerk hebben in de vorm van gedetailleerde ontwerpen, zodat overige medewerkers er ook mee aan de slag kunnen. Deze keuze heeft uiteraard invloed op de planning, de aanpassing op de planning is te zien in figuur 10.



Figuur 10 Aanpassing planning

Zoals te zien is, is het ontwerpen van het systeem twee weken naar voren geschoven, hierdoor is er ruim de tijd ingepland om het gehele systeem te ontwerpen. Tijdens het ontwerpen kan er al begonnen worden met het bouwen van de verschillende onderdelen. Ook hebben de eerder anonieme onderdelen een naam gekregen, deze onderdelen kwamen naar voren op basis van de opdrachtomschrijving en het vooronderzoek.

6.1.2 Opdeling requirements in categorieën

Tijdens het houden van het marktonderzoek werden er verschillende categorieën gebruikt om de functionaliteiten op te delen. Voor het opstellen van de requirements zijn wederom deze categorieën gebruikt (§5.2). Via deze verdeling ontstond er meer structuur en duidelijkheid binnen de requirements.

Categorie	Creëren enquêtes	Lay-out enquêtes	Media bibliotheek	Distribueren enquêtes	Resultaten enquêtes	Account
Aantal requirements	47	26	1	12	12	2

6.1.3 Vertaling, functionele requirements naar functionele gebruikers requirements

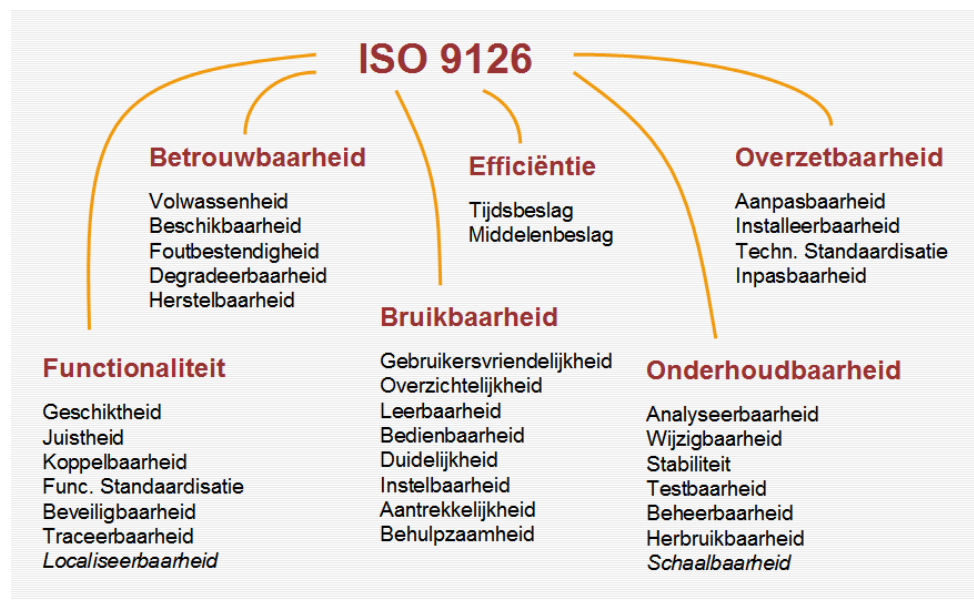
De interviews met de klanten waren nog steeds niet gehouden, de hoofdzakelijke reden hiervoor was dat de lijst met mogelijke klanten die kunnen worden geïnterviewd nog niet was opgesteld. Om de lijst op te stellen was veel afstemming met de opdrachtgever nodig en hier ging meer tijd inzitten dan verwacht. Mede hierdoor heb ik er voor gekozen om de functionele requirements te vertalen naar functionele gebruikers requirements.

Het verschil tussen beiden is dat functionele gebruikers requirements meer zijn afgesteld op de klanten van Webtechniek. Hierdoor kan er makkelijk worden gecommuniceerd met de klanten in de toekomstige interviews en terugkoppel momenten. Een voorbeeld is hieronder te zien:

#	Soort	Requirement
12	Functionele requirement	Het systeem moet aangeven wanneer er een vraag wordt beantwoord door een respondent ,er een melding wordt gegeven door middel van mail. De mail wordt verzonden naar de gebruiker van het systeem.
12	Gebruiker requirement	De klant geeft aan wanneer een vraag wordt beantwoord door een respondent hij een melding krijgt via mail tijdens het aanmaken van een vraag.

6.1.4 Niet functionele requirements, technische beperkingen en , bedrijfsregels

Zoals eerder genoemd is er gebruik gemaakt van de verdeling van de Swart binnen de requirements. Hierdoor ontstonden naast de categorie, functionele gebruikers requirements ook de categorieën: niet functionele requirements, technische beperkingen en bedrijfsregels. In deze paragraaf zullen per categorie een aantal requirements worden benoemd (behalve functionele gebruikers requirements, deze worden beschreven in §6.1.6). Beginnende met de niet functionele requirements, bij het opstellen van de niet-functionele requirements is gebruik gemaakt van de ISO 9126 normen. De niet functionele requirements vertellen iets over de kwaliteit van de applicatie verschillende per kwaliteitscategorie. In figuur 11 zijn de zes hoofdcategorieën uit de ISO 9126 normen te zien en hun subcategorieën.



Figuur 11 ISO 9126

#	Requirement	ISO categorie	Bron
126	Het systeem moet een up-time van 99% hebben.	Beschikbaarheid	Interview Webtechniek
111	Het systeem moet beschikbaar zijn op de vier standaard internet browsers: Internet Explorer, Safari, Chrome en Firefox.	Aanpasbaarheid	Interview Webtechniek
124	Het systeem moet beschikbaar zijn voor meerdere klanten.	Schaalbaarheid	Interview Webtechniek
129	Het systeem moet een back-up maken om 11:00, 15:00 , 00:00, 09:00 uur.	Beheerbaarheid	Interview Webtechniek

Technische beperkingen

De technische beperkingen komen voort uit het ICT-beleid binnen Webtechniek. Ze beperken de ruimte waarbinnen de functionele gebruikers requirements kunnen worden geïmplementeerd. Deze categorie van requirements heeft vooral invloed op de architectuur van het systeem.

#	Requirement	Bron
116	Het systeem moet in C# worden gebouwd.	Opdrachtomschrijving
117	Het systeem moet gebruik maken van een SQL DBMS.	Opdrachtomschrijving
118	Het systeem moet gebruik maken van Mailtechniek, voor het versturen van mails.	Interview Webtechniek
119	Het systeem moet TortoiseSVN als versietoolbeheer gebruiken.	Interview Webtechniek
120	Het systeem moet Microsoft Visual web developer 2010 gebruiken.	Opdrachtomschrijving
121	Het systeem moet Microsoft SQL Server 2008 R2 gebruiken voor het bouwen van de SQL DBMS.	Opdrachtomschrijving
125	Het systeem moet een webapplicatie worden.	Opdrachtomschrijving
139	Voor alle database handelingen (select, insert, update en delete) zal er gebruik worden gemaakt van de DB (database) klasse van Webtechniek.	Interview Webtechniek
143	De entiteiten enquête en company mogen nooit worden verwijderd uit de database.	Architectuur

Bedrijfsregels

Een bedrijfsregel is een regel die een bepaald aspect van de business definieert of beperkt. Het is bedoeld om de kenmerken van de business te handhaven of het gedrag van de business te beïnvloeden. De bedrijfsregels kwamen vooral voort uit het bedrijfsbeleid van Webtechniek.

#	Requirement	Bron
128	De inlognaam van klanten is een e-mailadres.	Interview Webtechniek
137	Per klant is maar één account beschikbaar.	Interview Webtechniek

6.1.5 Tegensprekende, aangepaste en verwijderde requirements

In §5.3.2 vielen al een aantal functionaliteiten af door het houden van interviews. Nu dat er begonnen is met het opstellen van de requirements kwamen er tegensprekende requirements, tegengestelde wensen en eisen naar voren of werden requirements aangepast of verwijderd aan de hand van feedback momenten met de stakeholders binnen Webtechniek.

Tegensprekende requirements

#	Requirement
149	De klant wil de enquête zijn status aangeven, die de volgende kan zijn: in ontwikkeling, actief en afgerond.
150	De klant wil de enquête zijn status aangeven, die de volgende kan zijn: actief en niet actief.

Tijdens het opstellen van de requirements kwam al snel naar voren dat requirement 149 en 150 elkaar tegenspreken qua soorten statussen, om hierin duidelijkheid te krijgen heb ik een kort gesprek gehouden met de opdrachtgever. Hem leek het beter om de statussen uit requirement 149 te nemen want wanneer je een enquête aanmaakt en nog niet alles heb ingevuld is hij nog niet actief maar in ontwikkeling. Als gevolg hiervan is requirement 150 verwijderd.

Tegengestelde wensen en eisen stakeholders

Tijdens een feedback moment met de opdrachtgever kwam hij met de voorkeur om voor het distribueren via mail gebruik te maken van mailtechniek. Omdat dit een bestaand systeem is heb ik een kort gesprek gehouden met de ontwikkelaar van mailtechniek en de opdrachtgever (die met dit voorstel kwam). Tijdens dit gesprek kwam naar voren dat de ontwikkelaar van mailtechniek niet echt tevreden was met de toekomstige koppeling tussen enquêtetechniek en mailtechniek, het zou namelijk betekenen dat voor elke contact_groep binnen enquêtetechniek een nieuwe contact_groep moet worden aangemaakt binnen mailtechniek. Er was dus veel synchronisatie tussen beiden systemen nodig. Om toch tot een oplossing te komen heb ik nogmaals een gesprek gehouden met beiden. Uit dit gesprek kwam uiteindelijk een middenweg tussen beiden wensen en eisen, namelijk de volgende punten:

- De mogelijkheid tot mailen moet alleen beschikbaar zijn voor klanten die al een account in Mailtechniek hebben.
- De contact_groepen moeten in mailtechniek aanwezig zijn, voordat ze in enquêteteknik aanwezig zijn.
- De contact_groepen in enquêteteknik staan gelijk aan de interests in mailtechniek.
- De mailtechniek maakt gebruik van de volgende onderdelen: Api_key, deze staat gelijk aan de company binnen Mailtechniek. De Group_key dit is de groep binnen Mailtechniek en de layout_key die gelijk staat aan de layout binnen Mailtechniek.

Zoals te zien is de koppeling tussen enquêteteknik en mailtechniek toch gemaakt, alleen is het synchronisatie gedeelte (waarover de ontwikkelaar van mailtechniek) niet tevreden was weg gelaten. Deze verantwoordelijkheid zal nu bij enquêteteknik komen te liggen. Bij de overige tegengestelde eisen en wensen van de stakeholders heb ik elke keer de zelfde aanpak gebruikt. Via deze manier wordt er een gesprek gehouden met de opdrachtgever mijzelf en de stakeholder(s) die met de betreffende eis of wens kwam om zo tot een juiste oplossing te komen waarbij alle stakeholders baat bij hebben.

Aangepaste requirement

#	Requirement	Aanpassing
4	De klant wil een taal kiezen waarin de enquête wordt gemaakt, meerdere talen zijn mogelijk.	Om de requirement specifieker te maken heb ik gevraagd aan de opdrachtgever welke talen moeten worden ondersteund en dat zijn de volgende: Duits, Engels, Spaans, Frans en Nederlands.

Verwijderde requirement

#	Requirement	Reden
21	De klant wil het vragetype: Matrix met textboxes kiezen tijdens het maken van een vraag.	Dit vraagtype onderscheidt zich niet van de overige matrixen, oftewel geen toegevoegde waarde.
108	De klant wil een pakket kiezen dat qua functionaliteiten en kosten past bij haar of zijn behoefte.	Deze functionaliteit hoeft niet in het enquêtesysteem omdat de opdrachtgever de kosten samenstelt, tevens worden de gebruikers ook aangemaakt door Webtechniek waardoor een selectie qua pakket niet mogelijk is door de klant.

Dubbelen requirements

#	Requirement
64	De klant wil de tekstsoort van de volgende onderdelen kiezen: Pagina, header, vragen, sub-vragen, response, knoppen en invulvelden. Tijdens het aanmaken van een layout.
68	De klant wil de lettertype zijn stijl van de volgende onderdelen kiezen: Pagina, header, vragen, sub-vragen, response, knoppen en Invulvelden. Tijdens het aanmaken van een layout.

Tijdens het ontwerpen van het onderdeel layout in de elaboration fase kwam ik een dubbele requirement tegen. Requirement 68 is namelijk het zelfde als requirement 64 ze gaan beiden over de tekstsoort en hebben betrekking op de zelfde elementen. Hierdoor is requirement 68 verwijderd.

6.1.6 MoSCoW- analyse

Nadat de requirements waren verdeeld in de verschillende categorieën (functionele requirements, niet functionele requirements, technische beperkingen, ontwerpbeslissingen en bedrijfsregels). Heb ik gebruik gemaakt van de meegegeven rating uit het tweede interview (dat is gehouden onder de medewerkers van Webtechniek). Om te zorgen dat alle meningen worden gehoord (qua hoogte van priorisatie) heb ik de ratings bij elkaar opgeteld, hierdoor worden alle stakeholders gehoord. Wanneer bijvoorbeeld één van de medewerkers een hoge rating geeft en de overige een lage rating worden deze ratings bij elkaar opgeteld en hebben hierdoor alle stakeholders invloed op de uiteindelijke rating.

Tegensprekende ratings

Wanneer er echter een te groot verschil tussen bepaalde ratings zat, bijvoorbeeld de ene stakeholder zegt één en de andere tien is het gemiddelde niet voldoende dekkend. Om een voorbeeld te geven hoe ik dit heb aangepakt zijn de volgende twee functionaliteiten als voorbeeld te zien:

#	Functionaliteit	Oplossing
29	Veel van de onderzochten aanbieders, bieden verschillende pakketten/accounts aan. Een voorbeeld: Single survey, Standard package, business package, individual package, student package, university package, major client en partner packages. Allemaal zijn ze afgestemd op een doelgroep en hun wensen.	Ik ben naar de opdrachtgever gegaan en neergelegd dat de ratings van de geïnterviewden van elkaar verschillen. Robin vond deze functionaliteit niet belangrijk (een rating van 3) terwijl Wouter dat juist wel vond (een rating van 8). Uiteindelijk legde ik de keuze bij de opdrachtgever neer en hij benadrukte dat hij over de soorten pakketten gaat (hoelang heeft een klant beschikking tot het systeem etc.) oftewel dit is niet een functionaliteit in de applicatie. De functionaliteit is naar de won't have categorie verplaatst.
32	Bij de resultaten is te zien wat de gemiddelde invul tijd van de enquête was.	Ook bij dit geval ben ik naar de opdrachtgever gegaan om aan te geven dat binnen de geïnterviewde stakeholders de ratings sterk van elkaar verschillen. Met name Wouter vond deze functionaliteit erg belangrijk (een rating van 10). De opdrachtgever zag inderdaad in dat deze functionaliteit een toegevoegde waarde heeft en relatief weinig tijd kost om te bouwen. De functionaliteit is verplaatst naar de categorie must have.

Zoals te zien in de twee bovenstaande voorbeelden heb ik elke keer contact opgenomen met de opdrachtgever omdat hij uiteindelijk de doorslag geeft. In bepaalde gevallen hebben de ratings van de overige stakeholders zijn gedachte veranderd zoals bij functionaliteit 32 te zien is.

Verdeling

Door de rating te verdelen in een viertal categorieën (het aantal categorieën die MoSCoW ook heeft) ontstond er een geprioriteerde lijst met functionaliteiten. Vervolgens zijn de functionaliteiten vertaald naar functionele gebruikers requirements. Hieruit bleek dat de nadruk vooral kwam te liggen op de requirements uit het onderdeel creëren van enquêtes. Het gevolg van deze bevinding is dat tijdens de overige onderdelen van dit project (het maken van het ontwerp, het bouwen van het systeem en testen) de nadruk vooral op dit onderdeel komt te liggen. Hieronder zijn alle requirements van het creëren van een enquête te zien:

Must have's	#	Requirement
	1	De klant wil een enquête aanmaken.
	5	De klant wil een titel geven aan een enquête tijdens het aanmaken van een enquête.
	7	De klant wil een vraag maken, tijdens het aanmaken van een enquête.
	8	De klant wil aangeven of de vraag verplicht is of niet, tijdens het aanmaken van een vraag.
	10	De klant wil aangeven of de vraag op een nieuwe pagina wordt getoond, tijdens het aanmaken van een vraag.
	15-29	De klant wil het vragentype keuzerondjes kiezen tijdens het maken van vragen. Overige vraagtypes zijn: selectievakjes (16), vervolgkeuzelijst (17), keuzelijst (18), matrix met keuzerondjes (19), matrix met selectievakjes (20), matrix met eindpunten (21), waarderingschaal (23), constante som (24), rangorde schaal (25), 1 tekstveld (27), meerdere tekstvelden (28) en 1 tekstveld met meerdere regels (29).
	39	De klant wil een antwoord op een vraag verwijzen naar een andere vraag waardoor conditionele vertakking mogelijk is, tijdens het aanmaken van vragen.

43	De klant wil bij meerkeuze vragen de optie, “anders” toevoegen wat een textbox is waarin de geënquêteerde zijn eigen suggestie kan invullen.
44	De klant wil na het aanmaken van de enquête, de enquête testen.
130	De klant wil in het vragentype waarderingsschaal kiezen hoeveel antwoorden hij heeft bijvoorbeeld: 1 t/m 10 of 1 t/m 3.
133	De klant wil een groep aanmaken.
134	De klant wil groepen koppelen aan een enquête.
140	De klant wil het vragentype likert schaal kiezen tijdens het maken van vragen.
141	De klant wil het vragentype Semantisch differentiaal kiezen tijdens het maken van vragen.
142	De klant wil het vragentype tegenstelling kiezen tijdens het maken van vragen.
145	De klant wil een start datum geven aan een enquête bij het aanmaken van een enquête.
146	De klant wil een eind datum geven aan een enquête bij het aanmaken van een enquête.
147	De klant wil een koptekst geven aan een enquête bij het aanmaken van een enquête.
148	De klant wil een voettekst geven aan een enquête bij het aanmaken van een enquête.
149	De klant wil de enquête de status aangeven van een enquête, die de volgende kan zijn: in ontwikkeling, actief en afgerond.

Should have's	#	Requirement
	2	De klant wil de welkoms tekst en afsluitende tekst die wordt ingevoegd tijdens het aanmaken van een enquête bewerken door middel van een ckeitor.
	6	De klant wil een start pagina aanmaken, tijdens het aanmaken van een enquête.
	30-33	De klant wil het vragentype: Instructie kiezen tijdens het maken van een vraag. Overige vraagtypes: Datumkiezer (31), Bestand uploaden (33),
	35	De klant wil een afsluitingspagina aanmaken, tijdens het aanmaken van een enquête
	36	De klant wil een alternatieve afsluitingspagina aanmaken, tijdens het aanmaken van een enquête
	37	De klant wil een bedankingsmail aanmaken, tijdens het aanmaken van een enquête
	38	De klant wil een “vertel een vriend” onderdeel toevoegen aan de bedankingsmail van een enquête.
	42	De klant wil suggesties toevoegen bij het vragentypes die een textbox bevat.

Could have's	#	Requirement
	12	De klant geeft aan wanneer een vraag wordt beantwoord door een respondent hij een melding krijgt via mail tijdens het aanmaken van een vraag.
	13	De klant wil een eerder gemaakte vraag aanpassen.
	40	De klant wil een antwoord op een vraag later terug laten komen in een vraag zijn stelling.
	45	De klant wil bij het aanmaken van een vraag de vragenbank raadplegen, dit is een verzameling van standaard vragen. Dit gebeurt tijdens het aanmaken van de enquête.
	136	De klant wil aangeven bij een meerkeuze vraag of de antwoorden per respondent het zelfde zijn, randomiseren, omwisselen per respondent , laatste optie altijd op de juiste positie of sorteren op alfabetische volgorde.

Won't have's	#	Requirement
	50	De klant wil een word bestand importeren om hieruit de vragen op te stellen, tijdens het aanmaken van een enquête.

Overige onderdelen

Om een beeld te geven van de omvang van de requirements waarmee ik te maken kreeg zal ik van nog twee categorieën (distribueren enquêtes en resultaten enquêtes) alleen de must-haves tonen. Om binnen dit document de overzicht te behouden is er maar een beperkt deel te zien van de in totaal 124 requirements.

Distribueren enquêtes

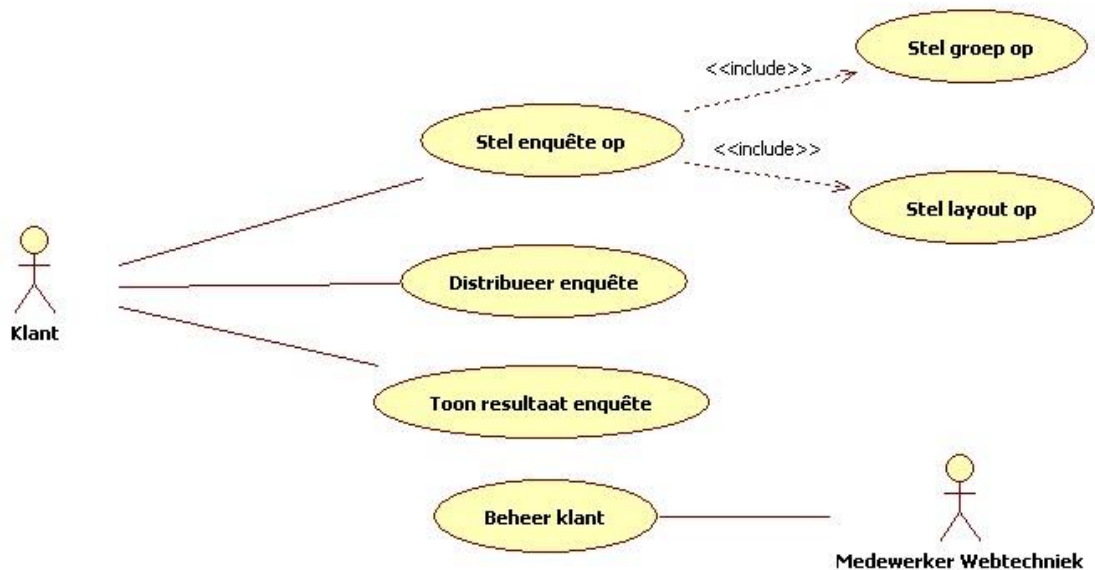
Must have's	#	Requirement
	80	De klant wil de enquête distribueren door middel van mail.
	82	De klant wil de enquête distribueren door middel van Facebook, Twitter en LinkedIn.
	83	De klant wil de enquête distribueren door middel van een link.
	84	De klant wil de enquête distribueren door middel van een embedded link.
	138	De klant wil dat respondenten een enquête kan invullen.

Resultaten enquêtes

Must have's	#	Requirement
	95	De klant wil een kort overzicht bekijken die alle belangrijke informatie van een enquête toont zoals: Titel enquête, status enquête, start datum en eind datum.
	96	Het systeem moet de resultaten van enquêtes overzichtelijk weergeven. Door middel van cirkeldiagrammen, staafdiagrammen en lijndiagrammen.
	131	Het systeem stuurt de resultaten via mail terug naar de respondent wanneer de enquête is ingevuld.

6.2 Use cases

Om de vertaalslag te maken van requirements naar een ontwerp heb ik als eerst een aantal use cases gemaakt die de hoofdfunctaliteiten van het systeem weergeven. Om te voorkomen dat er een groot aantal use cases ontstaan heb ik er voor gekozen om het te beperken tot alleen de hoofdfunctaliteiten. In figuur 10 zijn deze vijf functionaliteiten/onderdelen te zien. De use case dient er ondermeer voor om eventuele fouten in de requirements er uit te halen en als communicatie middel tussen mij en de stakeholders. Om vast te stellen welke requirements in welke use case thuis horen heb ik de categorieën uit §6.1.2 gebruikt. De use case diagram is te zien in figuur 12.



Figuur 12 Use case diagram enquêtetechniek

De use case diagram uit figuur 12 is gebaseerd op de opdrachtomschrijving en de volgende requirements:

#	Requirement
1	De klant wil een enquête aanmaken.
133	De klant wil een groep aanmaken.
134	De klant wil groepen koppelen aan een enquête.
110	De klant wil de groepen, enquêtes en layouts beheren.
109	De medewerker van Webtechniek wil een klant registreren door een gebruikersnaam op te geven, een

	bedrijfsnaam, een wachtwoord, e-mailadres en een userlevel.
96	Het systeem moet de resultaten van enquêtes overzichtelijk weergeven.
80	De klant wil de enquête distribueren door middel van mail.
51	De klant wil een layout aanmaken.
52	De klant wil een layout aan een enquête koppelen.

Van de use cases uit figuur 12, zijn use case omschrijvingen gemaakt. Tijdens het opstellen van de use case omschrijvingen kwam de volgende requirement sterk naar voren.

#	Requirement	ISO categorie	Bron
132	De klant wil door middel van een duidelijk stappenplan de enquête, groep en layout opstellen.	Gebruikersvriendelijkheid	Interview Webtechniek

Wat al snel duidelijk werd is dat de indeling van de stappen onduidelijk was. Om de indeling per stap duidelijk te maken heb ik dit besproken met zowel de opdrachtgever als de team manager (deze kwam namelijk met de requirement). De basic flow van het opstellen van een enquête is hieronder te zien.

Basic flow – Stel enquête op

Actor	Klant
Preconditions	1. Actor is ingelogd in het systeem
Scenario description	<ol style="list-style-type: none"> Actor stelt een enquête op, include (stel groep op & stel layout op). Actor stelt stap 1 samen van de enquête, hierin worden de volgende gegevens meegegeven: titel, startdatum[1], einddatum[2], welkomsttekst en welkomstafbeelding. Actor wilt naar de volgende stap. Systeem slaat de gegevens van stap 1 op. Systeem toont stap 2. Actor stelt stap 2 samen van de enquête, hierin worden de volgende gegevens meegegeven: koptekst, voettekst en taal. Actor voegt een layout aan een enquête toe. Actor wilt naar de volgende stap. Systeem slaat de gegevens van stap 2 op. Systeem toont stap 3. Actor stelt een vraag op. Actor stelt een antwoord op. Systeem voegt antwoord aan vraag toe. Herhaal stap 12 & 13 totdat alle antwoorden zijn toegevoegd aan de vraag. Herhaal stap 11 t/m 14 totdat alle vragen zijn toegevoegd aan de enquête. Actor wilt naar de volgende stap. Systeem toont stap 4. Actor stelt stap 4 samen van de enquête, hierin worden de volgende gegevens meegegeven: afsluitende tekst en afsluitende afbeelding. Actor wilt naar de volgende stap. Systeem slaat stap 4 op. Systeem toont stap 5. Actor voegt groepen aan een enquête toe[3]. Actor wilt naar de volgende stap. Systeem toont stap 6. Actor kiest de status in ontwikkeling. Actor eindigt de enquête.

	27. Systeem slaat de enquête met de status “In ontwikkel” op. 28. Systeem toont het enquête overzicht.
Exceptions	1. De startdatum moet vandaag zijn of later. 2. De einddatum moet later zijn dan de startdatum. 3. Er moet op ze minst één worden geselecteerd.
Postcondition	Het opstellen van een enquête is voltooid.

6.3 Architectuur enquêteteknik

Nadat de eerste requirements in kaart waren gezet, ben ik begonnen met het ontwerpen van de architectuur. De requirements uit de categorie technische beperkingen zullen vooral invloed hebben op de architectuur van het systeem, dit komt omdat ze de ruimte beperken waarbinnen de functionele gebruikers requirements kunnen worden geïmplementeerd. De architectuur beschrijft de structuur van enquêteteknik op een hoog abstract niveau. Om de architectuur op een structurele manier in kaart te zetten heb ik gebruik gemaakt van de methode uit het boek Software systems architecture van Nick Ronzanski en Eoin Woods. Om niet te lang te blijven hangen in de architectuur en omdat de implementatie van de applicatie buiten de scope valt, heb ik alleen de functional viewpoint omschreven.

6.3.1 Functional viewpoint

In de functional viewpoint worden alle functionele aspecten van de architectuur beschreven. De keuzes die tijdens deze viewpoint worden gemaakt zijn op een hoog abstract niveau en zullen de basis leggen voor het gehele systeem. Zo worden er keuzes gemaakt qua architectuur stijl en samenwerking met externe modules.

Architectuur stijl, tiered computing

Als architectuur stijl is er gekozen voor tiered computing er zijn twee redenen waarvoor deze is gekozen:

- Binnen Webtechniek is deze architectuurstijl een standaard.
- Enquêteteknik zal een web-based applicatie worden waardoor het zich meer leent voor een n-tier architectuur omdat niet alle onderdelen (cliënt, applicatie en database) op één locatie staan.

Presentation tier

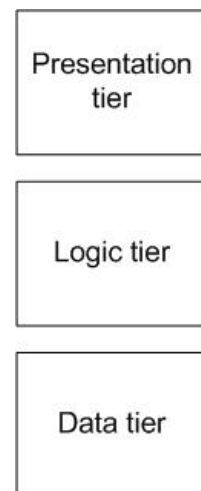
De presentation tier is de bovenste laag van de architectuur. De presentation tier toont informatie gerelateerd tot enquêteteknik. De presentation tier communiceert met de logic tier om alle handelingen die vanuit de representatie komen te verwerken.

Logic tier

De logic tier is de middelste laag van de architectuur. De logic tier controleert de functionaliteiten van de applicatie door de input van de presentation tier te verwerken. De logic tier communiceert met de data tier om de gegevens te verwerken en/of op te slaan.

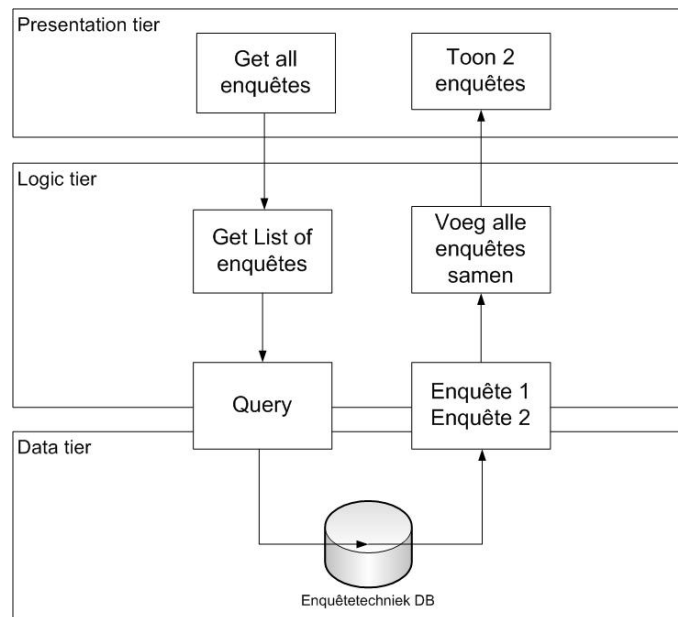
Data tier

De data tier is de onderste laag van de architectuur, op deze tier wordt de data opgeslagen en verwerkt. Welke data moet worden verwerkt of worden opgeslagen wordt aangestuurd door de logic tier.



Figuur 13 tiered computing

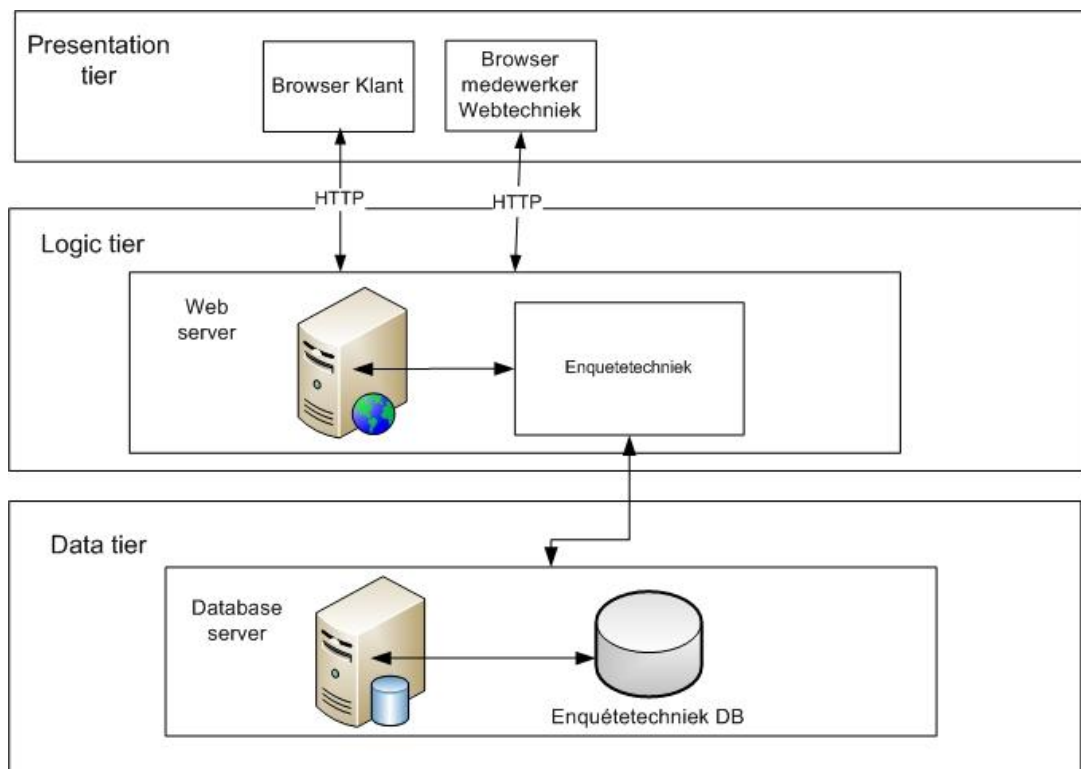
Om een voorbeeld te geven qua verantwoordelijkheden per tier is een schematische weergave gemaakt die te zien is in figuur 14. Hierin is duidelijk te zien dat in de presentation tier de handelingen vanuit de gebruiker worden afgevangen (get all enquêtes) en vervolgens worden getoond. In de logic tier wordt de input vanuit de presentation tier verwerkt (get list of enquêtes) en vervolgens samengevoegd om terug te sturen naar de presentation tier. In de data tier wordt afgehandeld welke data moet worden verzameld en terug gestuurd naar de logic tier.



Figuur 14 Verantwoordelijkheden per tier

6.3.1.1 Elementen binnen de architectuur

Tijdens het opstellen van de architectuur is er gekozen om te werken via het 3-tier model. Elke tier heeft zijn eigen verantwoordelijkheden en bevat andere elementen. Om duidelijkheid te creëren in welke laag welke elementen zitten is figuur 15 toegevoegd.



Figuur 15 invulling tier model

Presentatie tier: Op deze tier zal de web browser draaien. De verschillende gebruikers kunnen gebruik maken van hun web browser om enquêtetechniek te gebruiken. De web browser toont een HTML representatie van het systeem. Het voordeel van een web browser is dat het op allerlei verschillende devices kan draaien en hierdoor niet afhankelijk is van bijvoorbeeld één bepaald besturingssysteem. De web browser geeft input en krijgt output van de logic tier.

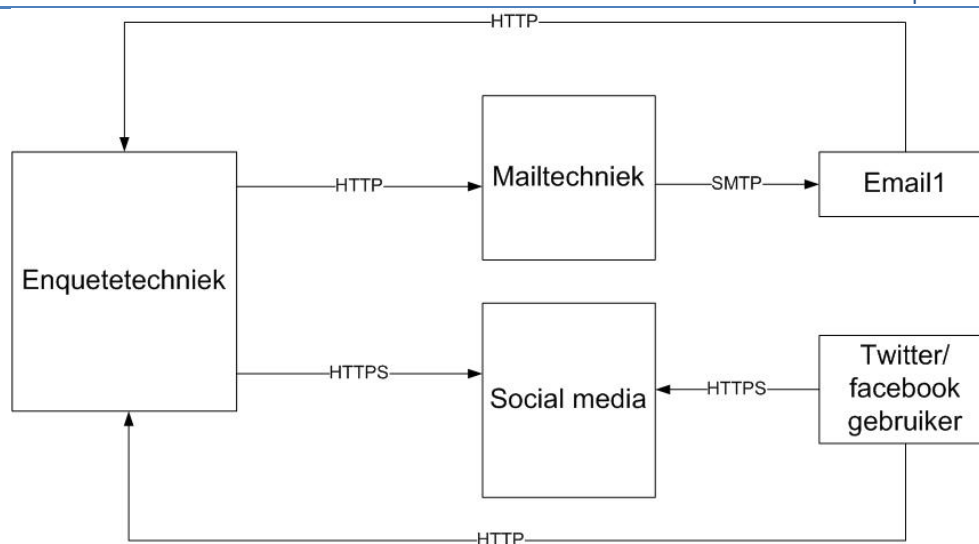
Logic tier: Op deze tier zal de applicatie gaan draaien, die gefaciliteerd wordt door een web server. De applicatie handelt alle bewerkingen af, die door de gebruikers worden opgegeven. De applicatie staat in direct contact met de database.

Data tier: Op deze tier zal de database gaan draaien, die gefaciliteerd wordt door een database server. De logic tier zal bepalen welke gegevens worden opgeslagen en welke worden opgevraagd.

6.3.1.2 Samenwerking externe modules

Enquêtetechniek zal samenwerken met een aantal externe applicaties en modules. Deze komen voort uit de volgende requirements:

#	Requirement	Bron
118	Het systeem moet gebruik maken van Mailtechniek, voor het versturen van mails.	Interview Webtechniek
82	De klant wil de enquête distribueren door middel van Facebook, Twitter en LinkedIn.	Vooronderzoek punt 17.



Figuur 16 Enquêtetechniek en betrokken modules

In figuur 16 is te zien wanneer de klant een enquête distribueert door middel van social media er een connectie wordt opgesteld vanuit enquêtetechniek met de betreffende gekozen social media (bijvoorbeeld facebook of twitter). Vervolgens kunnen de social media gebruikers de link van de enquête zien op hun account. Wanneer de social media gebruiker op de link klinkt zal hij worden verwezen naar enquêtetechniek waar de enquête wordt geopend. Wanneer de klant er voor kiest om de enquête te distribueren door middel van mail zal er als eerst verbinding worden gemaakt met mailtechniek, vervolgens zal mailtechniek een mailing sturen naar een aantal personen via het SMTP protocol. Dit is een standaard protocol voor het versturen van mails. De personen krijgen een mail met hierin een link van de betreffende enquête deze kan vervolgens worden ingevuld. Omdat alles via een browser gaat is het niet nodig om onderscheid te maken qua verschillende devices in de diagram.

6.3.2 Beperkt beschreven viewpoints

In deze paragraaf is per viewpoint een beargumentatie te zien waarom de betreffende viewpoint beperkt is beschreven in de architectuur beschrijving van dit systeem.

Informational viewpoint

In de informational viewpoint wordt beschreven hoe de architectuur informatie borgt, manipuleert, beheert en uitgeeft. Tijdens de informational viewpoint zijn de hoofdentiteiten beschreven. Omdat dit viewpoint beperkt is gebleven is deze niet toegevoegd aan dit document. Welke entiteiten er zijn en welke gegevens zij waarboren zal worden beschreven in §6.4.6 waar het relationeel representatie model te zien is.

Operational viewpoint

Dit viewpoint is heel beknopt beschreven in de architectuur beschrijving. Zo wordt er kort beschreven wie verantwoordelijk is voor de onderhoud aan het systeem, dit zijn de medewerkers van Webtechniek. En hoe back-up en restore plaatsvindt van de data die wordt bijgehouden binnen het systeem.

6.3.3 Niet beschreven viewpoints

In deze paragraaf is per viewpoint een beargumentatie te zien waarom de betreffende viewpoint niet is opgenomen in de architectuur beschrijving van dit systeem.

Development Viewpoint

In de development viewpoint wordt de architectuur beschreven die ondersteuning moet bieden tijdens het ontwikkelproces aan de ontwikkelaars van het systeem. Omdat de applicatie door maar één persoon wordt ontwikkeld (Mark Suurland), is er voor gekozen om dit viewpoint niet mee te nemen in de architectuur. Omdat de opdrachtgever eerder aangaf dat hij wel naslagwerk wil in de vorm van modellen zullen de requirements worden vertalen naar een ontwerp waarin bijvoorbeeld klassendiagrammen en sequencediagrammen te zien zijn. Omdat dit gedetailleerd ontwerpen zijn hoort het niet thuis in de architectuur beschrijving.

Concurrency Viewpoint

In de concurrency viewpoint worden de benodigheden beschreven hoe de afhandeling plaats vindt wanneer meerdere gebruikers de zelfde actie uitvoeren. Het systeem is zo opgebouwd dat er allerlei standaard oplossingen gebruikt kunnen worden om meerdere gebruikers tegelijkertijd aan te kunnen (zoals een webserver en een database server).

Ook is het belangrijk om te realiseren dat de gebruikers merendeel bezig zullen zijn met hun eigen enquêtes te beheren (oftewel niet meer dan één gebruiker per enquête, requirement 137), wanneer een enquête wordt gedistribueerd kunnen er wel tegelijkertijd meerdere respondenten de enquête invullen echter zijn dit alleen "read" en "create" functies die kunnen worden afgehandeld door een standaard DBMS (in dit geval SQL).

Hierdoor zullen er nooit situaties ontstaan dat meerdere gebruikers bij één unieke entiteit tegelijkertijd een actie kunnen uitvoeren die conflicten kan veroorzaken.

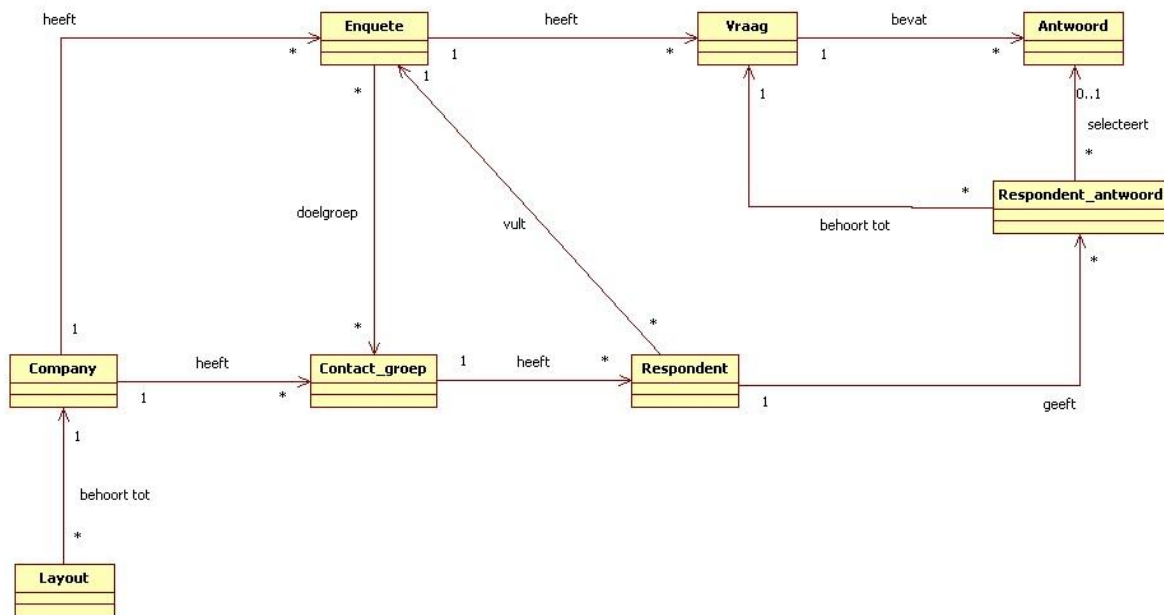
Deployment viewpoint

In de deployment viewpoint wordt de omgeving omschreven waarin het systeem zal gaan draaien. Omdat de implementatie buiten de scope van dit project valt (het eindproduct is een prototype) zal dit viewpoint op dit moment niet worden beschreven.

6.4 Ontwerp Enquêtetechniek

Om te kunnen beginnen met het ontwikkelen van een aantal systeemdelen heb ik gedetailleerd ontwerpen gemaakt. Zo ben ik begonnen met een klassendiagram om de statische structuur van de applicatie vast te leggen, de eerste versie van deze klassendiagram is te zien in figuur 17. In deze versie van het klassendiagram worden alleen de belangrijkste klassen getoond en hun onderling associaties. Vervolgens ben ik de lijst met requirements stuk voor stuk gaan doorlopen om tot een volledig klassendiagram te komen. Sommige requirements waren simpel weg een toevoeging van een attribuut aan een klasse (§ 6.4.3) terwijl bij anderen requirements er meer overleg met de stakeholders nodig was (§ 6.4.4). Zo zijn er voor bepaalde requirements uitgebreide ontwerpen gemaakt zoals design patterns of sequence diagrammen (§ 6.4.5). Om weer te geven op welke tier binnen het tier model zich alle klassen bevinden is § 6.4.2 toegevoegd.

6.4.1 Klassendiagram



Figuur 17 Eerste versie klassendiagram

Enquête: Deze klasse is verantwoordelijk voor alle algemene handelingen van een enquête, hij heeft een aantal vragen, een layout voor de vormgeving en een aantal contact_groepen om zo de enquête te distribueren aan verschillende doelgroepen.

Vraag: Deze klasse is verantwoordelijk voor de vragen van een enquête. Een vraag kan een aantal antwoorden hebben. Wanneer de vraag van het vraagtype openvraag of fileupload is heeft hij geen antwoord. Zoals te zien is een vraag per enquête uniek en kan dus niet worden hergebruikt over verschillende enquêtes, deze keuze is gemaakt omdat de requirement die deze keuze tegenspreekt laag in de priorisatie zit en tevens veel werk is om te realiseren (requirement 45, De klant wil bij het aanmaken van een vraag de vragenbank raadplegen, dit is een verzameling van standaard vragen, dit gebeurt tijdens het aanmaken van de enquête). In de toekomst kan er worden gekozen om deze functionaliteit wel toe te voegen.

Antwoord: Deze klasse is verantwoordelijk voor de antwoorden van een vraag. Een antwoord behoort altijd tot een vraag.

Company: Deze klasse is verantwoordelijk voor de gebruikers binnen Enquêtetechniek, hij heeft een aantal enquêtes en contact_groepen.

Contact_groep: Deze klasse is verantwoordelijk voor de contact_groepen dit zijn de verschillende doelgroepen binnen een enquête. Elke contact_groep heeft een aantal respondenten.

Respondent: Deze klasse is verantwoordelijk voor de respondenten van een enquête, een respondent zal altijd tot een contact_groep behoren. Een respondent heeft een aantal respondent_antwoorden. Een respondent kan maar één enquête invullen omdat hij altijd anoniem is. De associatie tussen respondent en enquête is nodig omdat er situaties kunnen voorkomen dat een respondent geen vragen invult maar dus wel heeft deelgenomen aan een enquête.

Respondent_antwoord: Deze klasse is verantwoordelijk voor de antwoorden die een respondent geeft op een vraag. Omdat bij sommige vraagtypes (fileupload en openvraag) er geen antwoorden aanwezig zijn moet zowel de vraag als het antwoord worden bijgehouden.

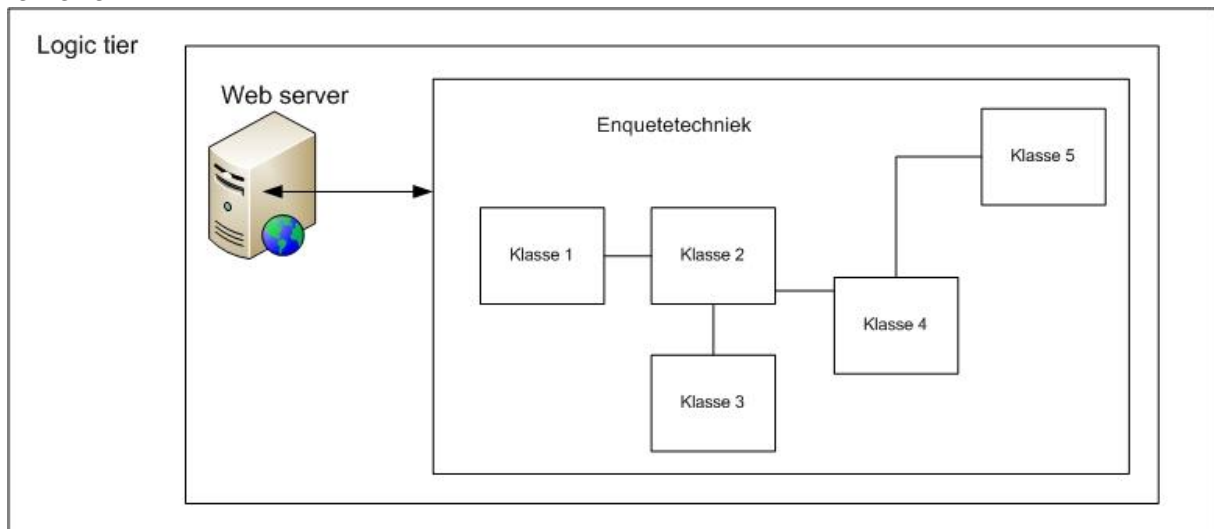
Layout: Deze klasse is verantwoordelijk voor de vormgeving van een enquête, een layout behoort altijd tot één company.

Decompositie van de enquête element

Zoals in figuur 17 te zien is zijn veel klassen aan de enquête klasse verbonden, hierdoor ontstaat er een hoge koppeling (coupling). In het boek Software Systems Architecture wordt dit ook wel een “god element” genoemd. Ik zat te denken om de enquête klasse op te breken in kleinere klassen. Echter heeft dit geen nut omdat de verschillende klassen al hun eigen verantwoordelijkheden hebben (cohesion) en het toevoegen van extra interfaces het systeem onnodig complex maken. Het is logisch dat in een enquêtesysteem de enquête het middelpunt is. Hierdoor is er niks aangepast aan het bestaande klassendiagram.

6.4.2 Plaats ontwerp binnen de architectuur

Om kort weer te geven binnen welke tier alle klassen zich bevinden is figuur 18 toegevoegd. Zoals te zien bevindt zich in de logic tier de applicatie en dus ook zijn bijhorende klassen om alle logische handelingen te verwerken.



Figuur 18 plaats klassen binnen tier-model

6.4.3 Toevoeging attributen

Om een voorbeeld te geven van een requirement waarbij alleen een attribuut wordt toegevoegd aan een bestaande klasse zijn hieronder requirement 5 en 8 te zien:

Requirement 5: Titel enquête

Attribuut titel van het type string toegevoegd aan de klasse enquête.

Requirement 8: Vraag verplicht

Attribuut verplicht van het type boolean toegevoegd aan de klasse vraag.

6.4.4 Overleg stakeholders

Om een voorbeeld te geven waarbij overleg met de stakeholders binnen Webtechniek nodig was zijn requirements 112 en 113 toegevoegd.

#	Requirement
112	De klant wil de URL waarmee de enquête kan worden gedistribueerd beveiligen (HTTPS).
113	De klant wil de URL waarmee de enquête kan worden gedistribueerd niet beveiligen (HTTP).

Zoals te zien zijn het tegensprekende requirements waardoor ik geen keuze kon maken tussen beiden. Uit Requirement 112 is namelijk op te maken dat het HTTPS protocol moet worden gebruikt terwijl uit requirement 113 op te maken is dat het HTTP protocol moet worden gebruikt.

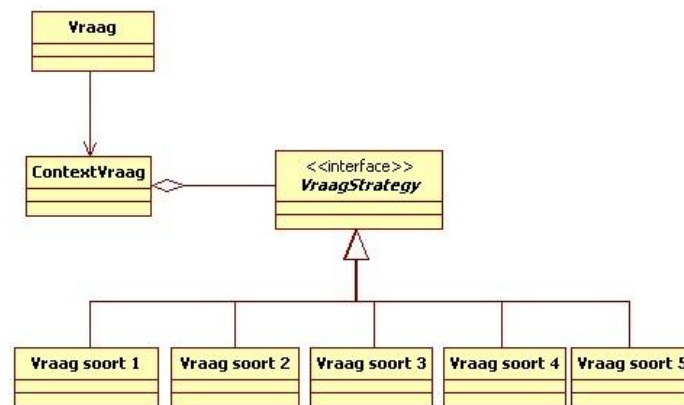
Omdat het een technisch aspect is heb ik een gesprek gehouden met de technische directeur binnen Webtechniek om te kijken wat hij als standaard of voorkeur heeft. Hieruit kwam naar voren dat eerst HTTP wordt gebruikt en wanneer in de toekomst blijkt dat de klanten van Enquêtetechniek gevoelige informatie uitwisselen het HTTPS protocol kan worden toegepast.

6.4.5 Uitgebreide ontwerpen

Voor sommige requirements waren meer gedetailleerde ontwerpen nodig zoals bijvoorbeeld sequencediagrammen en design patterns. In deze paragraaf zullen een aantal voorbeelden worden gegeven.

Requirement 15-33,130,140-142: Vraagtypes

In requirement 15 t/m 33, 130 en 140 t/m 142 worden de vraagtypes beschreven. Deze hebben allen een verschillende gedrag qua presentatie naar de respondent toe. Maar ook het opslaan van de antwoorden (vanuit de respondent) verschillen qua gedrag per vraagtype. Hierdoor heb ik het strategy pattern toegepast waarvan een schematische weergave te zien is in figuur 19. Per vraagtype zal er een aparte klasse bestaan waarin zijn unieke gedrag wordt beschreven. De VraagStrategy klasse zal als een interface dienen waarin een bepaalde template wordt gedefinieerd die door zijn onderliggende klassen moeten worden geïmplementeerd. De contextvraag klasse zorgt er voor dat de juiste strategy klasse wordt aangemaakt aan de hand van het vraag type (vanuit de vraag klasse).



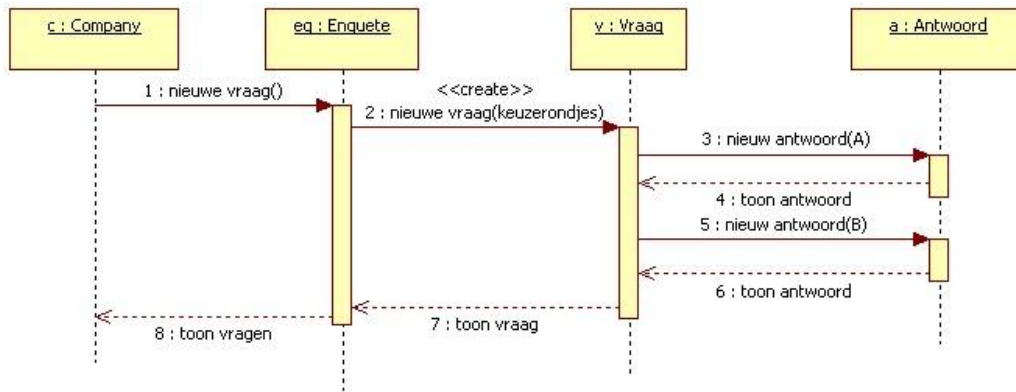
Figuur 19 Strategy pattern vraagtypes

Uiteraard heeft deze keuze gevolgen voor het systeem, vooral omdat elke soort vraag de zelfde antwoord klasse gebruikt. Ik zal kort uitleggen waarom dit toch correct werkt. Voor de meerkeuze vraagtypes zoals waardingsschaal, vervolgkeuzelijst, selectievakjes, matrixcheckbox, matrixradiobutton, keuzerondjes en keuzelijst wordt de antwoord klasse gebruikt om hun antwoorden op te slaan (bijvoorbeeld antwoord:A,B en C). Bij de vraagtypes waarbij geen antwoord mogelijk is zoals fileupload en instructie zullen er geen antwoorden worden opgeslagen.

De vraagtypes datum en open antwoord daarin tegen zijn uitzonderingen, deze hebben niet een specifiek antwoord om te tonen maar kunnen toch verschillen qua opbouw van de vraag. Bij een openvraag kan er bijvoorbeeld één regel worden getoond of drie regels (die vervolgens worden ingevuld door de respondent), dit wordt bijgehouden in een enkel antwoord (zoals bijvoorbeeld één regel), want de respondent is de gene die een datum of bepaalde tekst invult.

Vraagtypes waardingsschaal,vervolgkeuzelijst, selectievakjes, keuzerondjes en keuzelijst

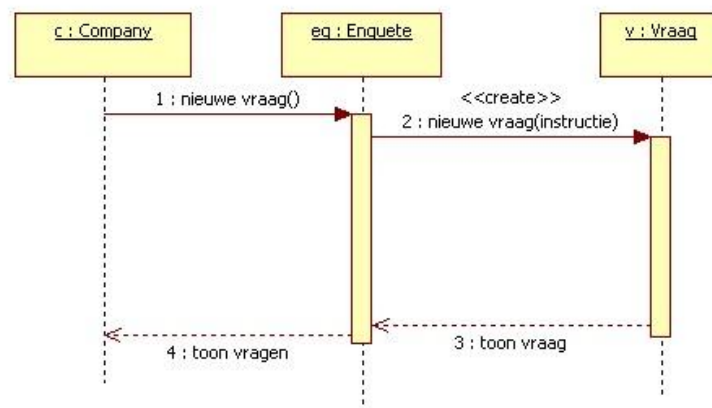
Al deze vraagtypes zijn meerkeuze, ze verschillen elk qua presentatie naar de respondent toe hierdoor zijn ze elk apart toegevoegd aan het strategy pattern. Qua aanmaken van deze vraagtypes (door de gebruikers) verschillen zij niet. Zo wordt er eerst een vraag gemaakt en vervolgens één voor één de antwoorden. De respondent kan vervolgens een antwoord kiezen (bij het tonen van een enquête). Een voorbeeld van het aanmaken van een keuzerondjes vraag is te zien in afbeelding 20.



Figuur 20 voorbeeld meerkeuze vraagtypes

Vraagtype fileupload en instructie

Deze twee vraagtypes hebben geen vooraf vastgestelde antwoorden. Bij een fileupload kan wel een bestand worden geüpload door de respondent die als respondent_antwoord wordt opgeslagen. Bij een instructie is het geven van een antwoord door de respondent niet mogelijk. In figuur 21 is een voorbeeld te zien hoe een instructie vraag wordt aangemaakt door de gebruiker (het toevoegen van een stelling spreekt voor zich hierdoor is deze stap niet toegevoegd).



Figuur 21 voorbeeld instructie vraagtype

Vraagtype datum en open

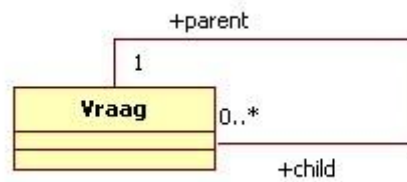
Deze twee vraagtypes hebben ook geen vooraf vastgestelde antwoorden. Bij een datum en open vraag wordt namelijk naar een waarde gevraagd die niet vooraf kan worden vastgesteld. Er wordt enkel een tekstvak getoond waar de respondent een eigen antwoord kan invoeren.

Vraagtype matrix

In requirement 19-21 wordt er gesproken over een vraagtype matrix deze is qua opbouw anders dan de overige vraagtypes. Na een gesprek met de opdrachtgever heb ik gekozen voor de volgende aanpak. Als eerst zal de opbouw het zelfde zijn als bij een meerkeuze vraag: Eerst worden de antwoorden opgeven zoals bijvoorbeeld A tot en met E. Vervolgens kunnen een aantal onderdelen worden opgegeven, deze “kopiëren” de antwoorden van hun “parent”. Nadat de gebruiker een aantal onderdelen heeft toegevoegd ontstaat de volgende situatie:

Vraag 1 (antwoorden A-E)
 Onderdeel 1: a b c d e
 Onderdeel 2: a b c d e
 Onderdeel 3: a b c d e

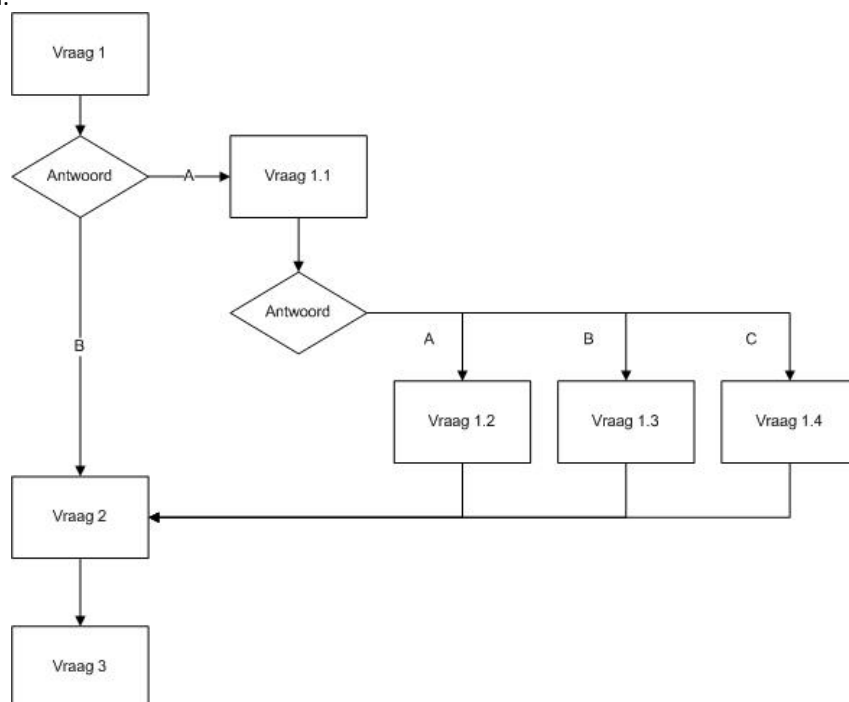
Als gevolg hiervan is er een nieuwe associatie toegevoegd namelijk van de vraag klasse naar zichzelf dit is de parent (vraag) en child (onderdeel) relatie. Het klassendiagram hiervan is te zien in figuur 22.



Figuur 22 parent en child relatie binnen vraag

Requirement 39: Conditionele vertakking

Omdat er conditionele vertakking binnen een enquête mogelijk moet zijn wou ik dit oplossen door bij een antwoord de mogelijkheid te geven om een doorvraag te kiezen. Hierdoor kan er afhankelijk van het gedrag van de respondent een andere enquête worden getoond. Een voorbeeld van de opbouw van enquête is in figuur 23 te zien.



Figuur 23 conditionele vertakking

Om de bovenstaande situatie uit figuur 23 te kunnen realiseren is er een associatie van antwoord naar vraag toegevoegd aan het klassendiagram. Hierdoor kan een antwoord een vraag als doorvraag hebben. In figuur 24 is dit schematisch weergegeven.

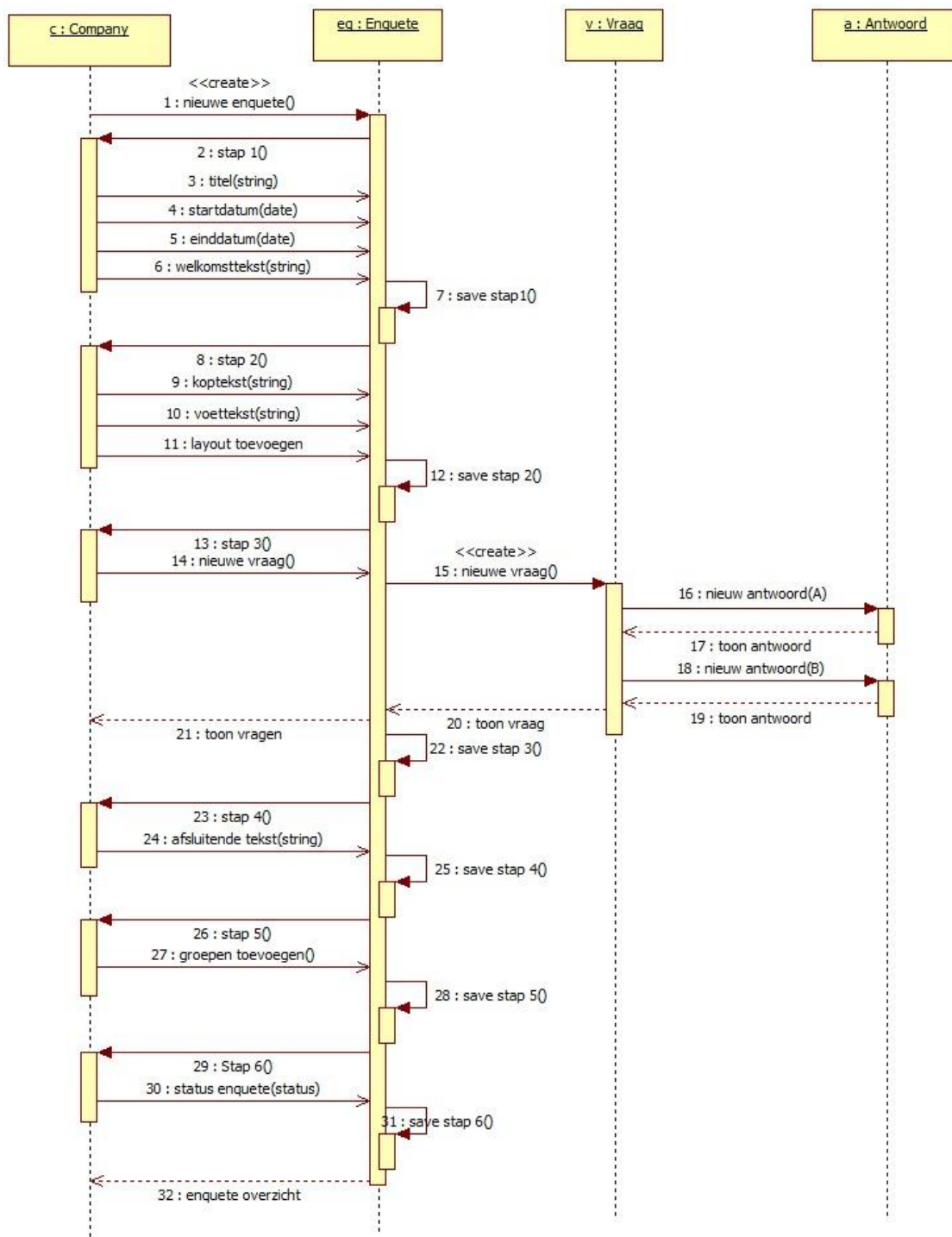


Figuur 24 associatie doorvraag

Omdat het niet logisch is dat bijvoorbeeld openvragen een doorvraag kan hebben zal alleen conditionele vertakking mogelijk zijn bij vraagsoorten waarbij één antwoord kan worden gekozen. Dit zijn de vraagtypes: keuzelijst, keuzerondjes, vervolgkeuzelijst en waardingschaal.

Requirement 132: Stappenplan opstellen enquête

In requirement 132 wordt beschreven dat het opstellen van de enquête via een stappenplan moet. Om een invulling te geven per stap en de totaal aantal stappen is een sequencediagram gemaakt. Deze diende er met name voor om terug te koppelen met de stakeholders om zo tot een duidelijk stappenplan te komen. De sequencediagram is te zien in figuur 25.



Figuur 25 Sequencediagram stappenplan enquête

6.4.6 Relatieve representatie model

Om een database te ontwerpen wordt er normaliter eerst een EERD (enhanced entity relationship diagram) opgesteld waarin de verschillende entiteiten, hun associaties en hun attributen te zien zijn. Omdat dit diagram veel weg heeft van het al bestaande klassendiagram zal ik alleen een RRM (relatieve representatie model) maken hierin worden de entiteiten zelf gedetailleerd omschreven.

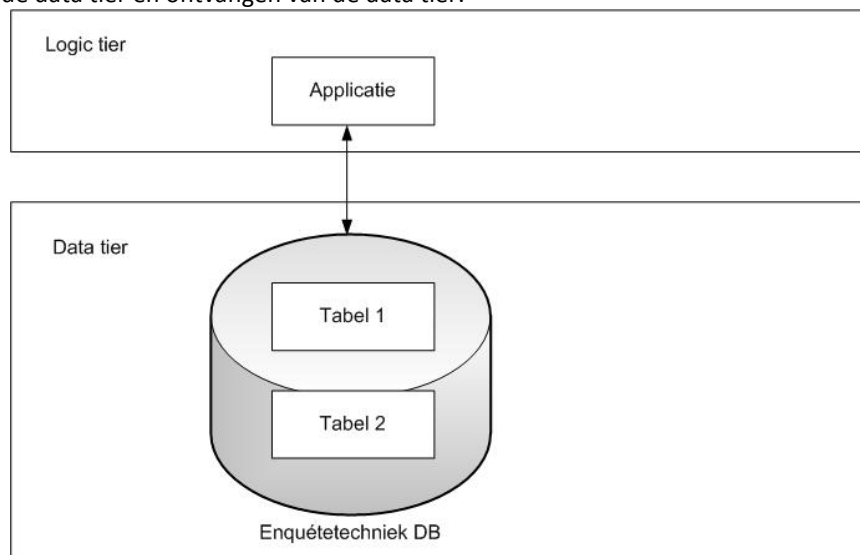
Om de database uiteindelijk op te zetten wordt er vaak een RIM (Relatieve implementatie model) gemaakt. In de RIM staan de create statements van de database tabellen. Binnen ASP.NET is er een tool aanwezig die tabellen op een simpele manier aanmaakt hierdoor zal ik ook dit diagram overslaan. Een voorbeeld uit het RRM van de tabel Respondent_antwoord:

Respondent_antwoord(id, respondent_id, vraag_id, antwoord_id, antwoord_text, skipped)

- Respondent_id is een vreemde sleutel die verwijst naar id in tabel Respondent NULL niet toegestaan.
- Vraag_id is een vreemde sleutel die verwijst naar id in tabel Vraag NULL niet toegestaan.
- Antwoord_id is een vreemde sleutel die verwijst naar id in tabel NULL wel toegestaan.

6.4.7 Plaats database binnen de architectuur

Om kort weer te geven binnen welke tier de database van enquête-techniek zich bevindt is figuur 26 toegevoegd. Zoals te zien bevindt de database op de data tier, de logic tier (de applicatie) zal informatie versturen naar de data tier en ontvangen van de data tier.



Figuur 26 Data tier

6.4.8 Overige details database

Tijdens het opstellen van de RRM kwamen ook meer details aan bod die tijdens deze paragraaf worden behandeld. Deze details zijn referentiële integriteit, relaties tussen primaire en vreemde sleutels en triggers.

Referentiële integriteit

Met regels voor referentiële integriteit wordt voorkomen dat een toewijzingen tussen verwante kolommen wordt verbroken. Om referentiële integriteit te behouden heb ik er voor gekozen dat elke unieke sleutel van een tabel niet kan worden aangepast en wanneer er een nieuwe tuple wordt toegevoegd aan de betreffende tabel de primaire sleutel automatisch worden opgehoogd met één, hierdoor is hij altijd uniek.

Relaties tussen primaire en vreemde sleutels

Omdat de gebruiker binnen enquête-techniek bepaalde entiteiten kunnen aanpassen of verwijderen zoals: enquête, vraag, antwoord, layout, elementen en contact_groep. Is er per entiteit bekeken welke gevolgen dit heeft voor zijn betrokken vreemde sleutels.

Wanneer een vraag wordt verwijderd is het logisch dat zijn bijhorende antwoorden ook worden verwijderd. Het herstellen van een vraag is hierdoor niet mogelijk.

Voor de entiteiten enquête, layout en contact_groep tabel heb ik een nieuw attribuut toegevoegd namelijk "deleted". Wanneer de gebruiker bijvoorbeeld een enquête wilt verwijderen zal dit attribuut op "true" worden gezet hierdoor is de enquête niet definitief verwijderd uit de database, maar kan hij binnen het systeem niet meer worden getoond. Hierdoor wordt een enquête, layout of contact_groep nooit definitief verwijderd en dus ook niet zijn betrokkenen vreemde sleutels zoals de vragen van een enquête of de elementen van een layout.

Triggers

Triggers zijn stukjes code die automatisch worden uitgevoerd wanneer een bepaalde actie wordt uitgevoerd op een tabel. De volgende triggers zijn toegevoegd:

- Wanneer een enquête zijn status leeg is bij het toevoegen wordt hij automatisch op "in ontwikkeling" gezet.
- Wanneer de invuldatum van een respondent leeg is, wordt de datum van vandaag ingevuld.

6.5 Elaboration rapport

Om de tweede fase af te ronden werd een elaboration rapport opgesteld. Dit rapport was de uitkomst van de werkzaamheden die tijdens de elaboration fase zijn gedaan zoals het opstellen van de architectuur en het ontwerp. Het doel hiervan was om de betrokkenen op de hoogte te houden van mijn werkzaamheden en te laten zien wat de mogelijke ideeën waren.

Ter inzage is het elaboration rapport toegevoegd in Bijlage D.

7 Bouw onderdeel 1: Creëren enquête

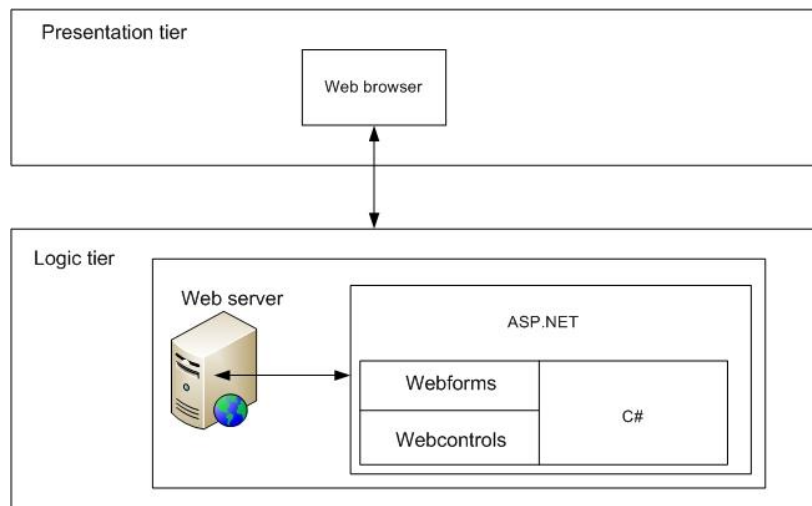
Het eerste onderdeel dat gebouwd kon worden was het creëren van enquêtes, tijdens de MoSCoW analyse (§6.1.6) kwam duidelijk naar voren dat dit het belangrijkste onderdeel was en hierdoor dit onderdeel als eerst werd gebouwd. In dit hoofdstuk zal een omschrijving worden gegeven hoe de verschillende requirements die behoren tot dit onderdeel zijn gebouwd. Per onderdeel zullen maar een aantal requirements aan bod komen de overige requirements zijn te zien in bijlage E.

7.1.1 Opstellen basis van de applicatie

Voordat kon worden begonnen met het bouwen van het eerste onderdeel heb ik eerst de basis van de applicatie gelegd. Naast het toevoegen van de standaard klassen kwamen nog een aantal interessante toevoegingen naar boven die ik even kort zal toelichten. In de paragrafen die hierop volgen zal het opstellen van een enquête, groep en layout aan bod komen.

Webforms

Binnen Asp.net Webforms, wordt er gebruik gemaakt van webforms. Dit zijn pagina's waarop controls worden getoond. Controls zijn HTML onderdelen zoals een tekstbox, afbeelding of label. In figuur 27 is het webform principe te zien, een pagina bestaat uit een webform met zijn webcontrols. De handelingen worden afgehandeld door de achterliggende code, wat in dit geval C# is.



Figuur 27 Web form binnen asp.net

Toevoeging DB klasse

Tijdens het opstellen van de database methodes (select, insert, update en delete) heb ik gevraagd aan de project manager wat binnen het bedrijf als standaard wordt gebruikt. Hieruit bleek dat zij een eigen DB (database) klasse hebben die moet worden gebruikt binnen elk project. De klasse is verantwoordelijk voor alle directe handelingen met de database. De volgende requirement is hierdoor toegevoegd aan de bestaande requirements:

#	Requirement	Bron
139	Voor alle database handelingen (select, insert,update en delete) zal er gebruik worden gemaakt van de DB (database) klasse van Webtechniek.	IW

Omdat per klasse maar één instantie van de database klasse beschikbaar mag zijn heb ik het singleton pattern toegepast wat te zien is in figuur 28.

```

private DB db1 = null;
private DB db
{
    get
    {
        if (db1 == null)
            db1 = new DB();
        return db1;
    }
}

```

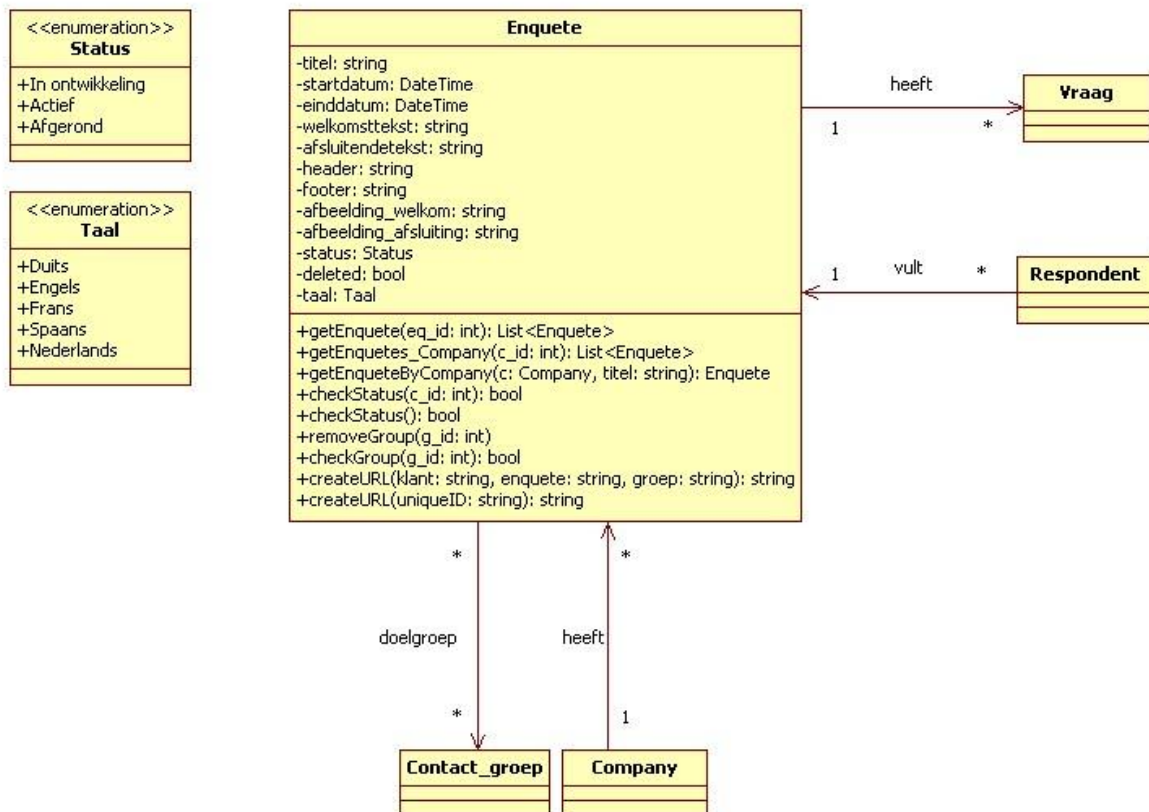
Figuur 28 Singleton pattern DB instantie

7.1.2 Creëren enquêtes

In deze paragraaf is de bouw van een aantal requirements te zien die betrekking hebben op het onderdeel creëren enquêtes, de bouw van de overige requirements is te zien in bijlage E.

Enquête en betrokken klassen

Omdat een groot deel van de requirements binnen de categorie creëren van enquête betrekking hebben op de enquête klasse, zal ik een klassendiagram tonen van de enquête klassen en zijn betrokken klassen. Om de diagram overzichtelijk te houden zijn alleen de attributen en methodes van de enquête klasse te zien.



Figuur 29 enquête klasse

Uitleg methodes

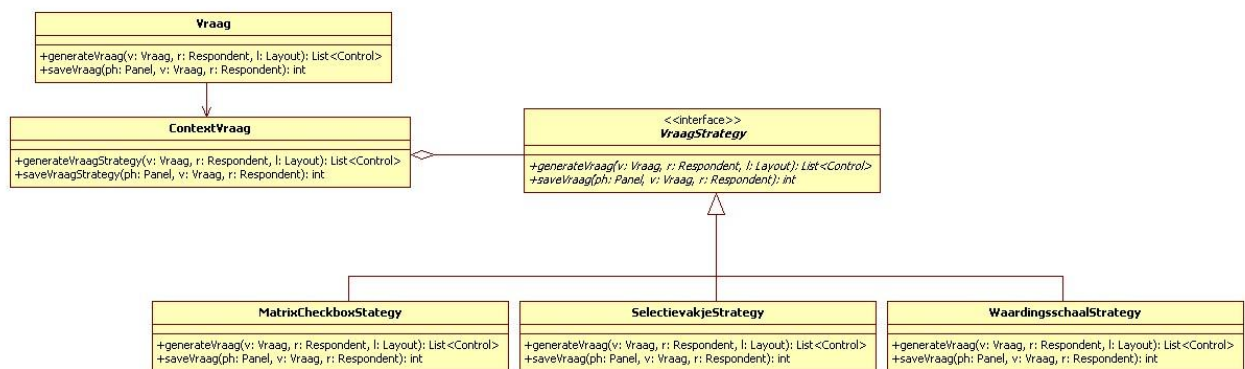
Om de enquêtes op te halen is de methodes getEnquete toegevoegd. Om de status te controleren van een bepaalde enquête is de methodes checkStatus toegevoegd.

Requirement 15-29: Implementatie strategy pattern vraagtypes

Tijdens het ontwerp van de verschillende requirements werd de keuze gemaakt om voor de verschillende vraagtypes het strategy pattern toe te passen. Deze keuze is gemaakt omdat per vraagtype er verschillend

gedrag wordt vertoond, het doel van het strategy pattern is om dit gedrag te scheiden in verschillende klassen. Het implementeren van het strategy pattern is in een aantal stappen gegaan namelijk de volgende:

1. Als eerst zullen er twee methodes worden toegevoegd binnen de Vraag klasse, in dit geval zijn dat de generateVraag en saveVraag. De generateVraag methode geeft een lijst met Controls (bijvoorbeeld een textbox of een tekst) terug die samen een vraag vormen die in de enquête kunnen worden getoond. De tweede methode is saveVraag, hierin wordt de vraag opgeslagen die door een respondent is ingevuld, deze returned de eventuele doorvraag zijn id.
2. Aan de hand van het vraag type zal er een vraagStrategy klasse worden aangemaakt. Vervolgens zal aan de hand van het type van de vraag een specifieke strategy worden aangemaakt (niet alle vraagtypes zijn toegevoegd om zo de afbeelding overzichtelijk te houden). Aan deze specifieke strategy klasse kan één van de twee methodes worden aangeroepen.
3. De hoofdklasse vraagStrategy zal als een interface figuren (geen implementatie aan de methodes) de subklassen zullen van dit interface overerven.
4. Alle subklassen zullen overerven van de hoofdklassen Vraagstrategy, de twee methodes zullen hier een implementatie krijgen, deze implementatie verschilt per vraagtype. Dit is dan ook gelijk de essentie van dit pattern om per vraagtype een andere implementatie van de generateVraag en saveVraag methode.



Figuur 30 Strategy pattern vraagtypes

Requirement 19-21: Implementatie singleton pattern

De klasse vraag heeft een instantie van contextVraag om gebruik te maken van het strategy pattern voor de verschillende vraagtypes. Echter mag er maar één vraagtype en dus contextVraag instantie per vraag aanwezig zijn. Om dit af te dwingen is het singleton pattern toegepast, de implementatie is hieronder te zien:

```

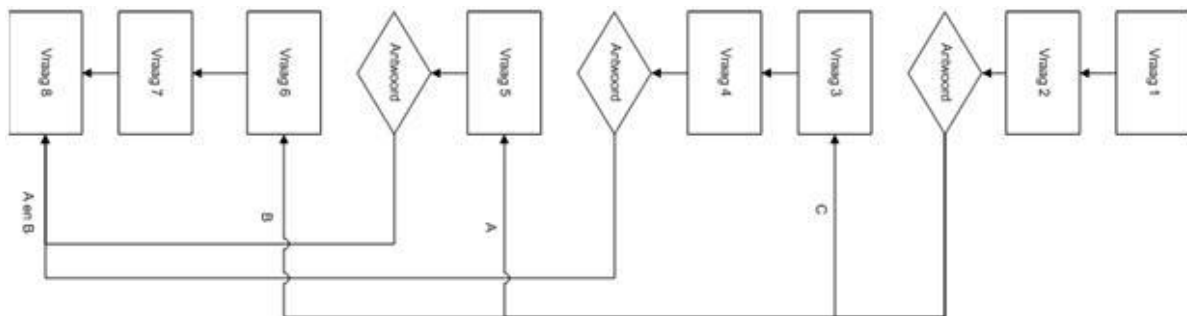
public class Vraag
{
    private ContextVraag _context = null;
    public ContextVraag Context
    {
        get
        {
            if (_context == null)
            {
                _context = new ContextVraag(_type.GetValueOrDefault(VraagType.Open));
            }
            return _context;
        }
        set { _context = value; }
    }
}
  
```

Figuur 31 Singleton pattern binnen Vraag klasse

Requirement 39: Conditionele vertakking

Na het realiseren van de situatie uit figuur 23, werd het onderdeel getest door middel van moduletesten. Wat al snel duidelijk werd is dat er nooit een aantal vragen konden worden overgeslagen. Bij een papieren enquête

is dit vaak wel mogelijk wanneer er een antwoord van een vraag wordt geselecteerd een aantal vragen worden overgeslagen. Om een voorbeeld van de nieuwe situatie te geven zal ik wederom een flowchart hiervan maken. Deze is te zien in figuur 32.



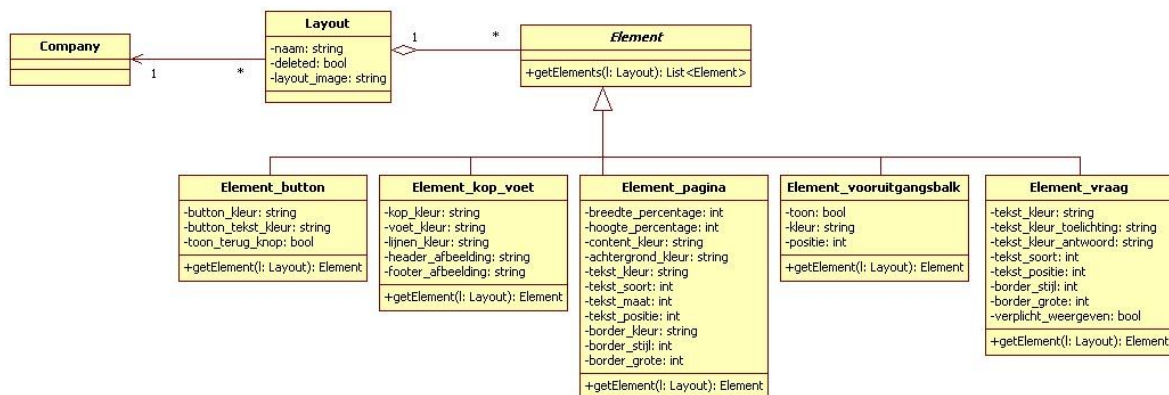
Figuur 32 doorvraag nieuwe situatie

7.1.3 Creëren layouts

In deze paragraaf is de bouw van een aantal requirements te zien die betrekking hebben op het onderdeel creëren layouts, de bouw van de overige requirements is te zien in bijlage E.

Layout met elementen

Aan de hand van de requirements is gekeken welke elementen elke layout heeft, hieruit zijn de elementen: button, kop_voet, pagina, vooruitgangsbalk en vraag ontstaan met hun bijhorende attributen. Vervolgens is de abstracte klasse element toegevoegd met een methode `getElements` die een lijst met elementen terug geeft, zodat de layout kan worden opgebouwd. Mede door deze opbouw bestaat een layout altijd uit een bepaald aantal elementen, mocht er in de toekomst nieuwe elementen bij komen kunnen deze makkelijk worden toegevoegd zonder dat dit gevolgen heeft voor het systeem.

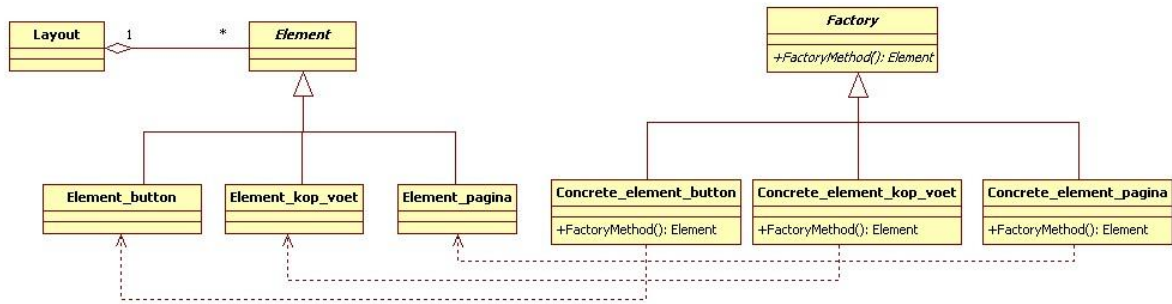


Figuur 33 Layout en zijn elementen

Factory pattern voor het creëren van elementen

Omdat een layout meerdere elementen heeft die elk verschillende attributen en methodes bevatten. Loop ik tegen het probleem aan dat ik niet weet welk element ik moet creëren per stap. Om dit probleem te verhelpen heb ik het factory pattern toegepast, in figuur 34 is deze te zien. Nu kunnen er objecten (elementen) worden geïnstantieerd zonder exact vast te hoeven leggen van welke klassen deze objecten zullen zijn (buiten het factory pattern). Het vaststellen van welke objecten er worden geïnstantieerd per stap is vastgelegd in de factorymethod van de concrete_element klasse.

Door het toepassen van dit pattern wordt de onderhoud van het programma ook vergroot. Als er een nieuw element wordt toegevoegd hoeft er alleen een nieuwe factory klasse te worden geïmplementeerd of kan hij worden geïnstantieerd in een bestaande concrete_element klasse.



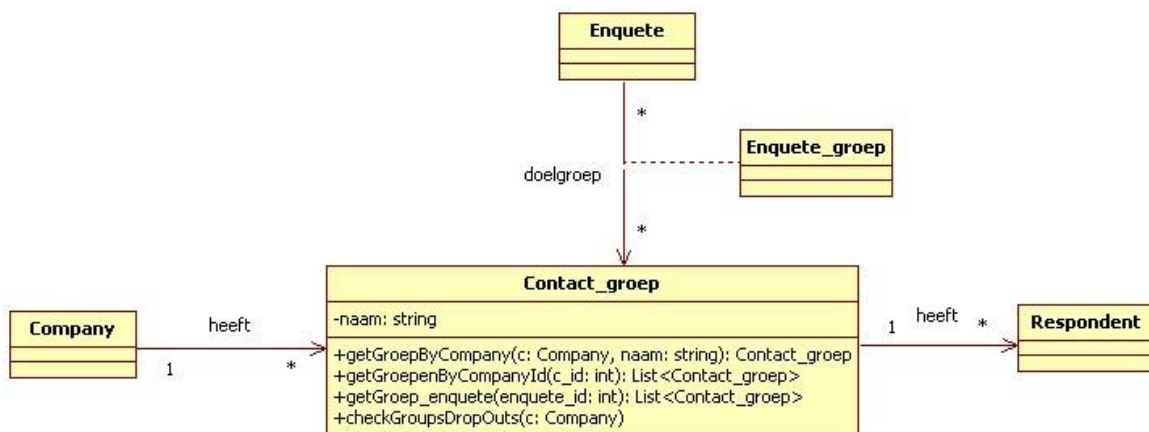
Figuur 34 Factory pattern vragen

7.1.4 Creëren groepen

In deze paragraaf zal het creëren van een groep worden besproken. Omdat er niet veel requirements tot deze categorie behoren, zal ik ze allemaal in deze paragraaf behandelen. Omdat de requirements van deze categorie nog niet eerder aan bod zijn gekomen binnen dit document zal ik kort toelichten wat het doel van de contact_groep klasse binnen het enquêtesysteem is. Wanneer een gebruiker zijn enquête wil distribueren en onderscheid wil maken tussen verschillende doelgroepen (bijvoorbeeld werknemers en niet-werknemers) kunnen voor verschillende doelgroepen een contact_groep worden aangemaakt. Deze dienen er voor om onderscheid te maken tussen de resultaten van de respondenten en het distribueren van enquêtes.

Requirement 133: groep aanmaken

In figuur 35 is de contact_groep klasse te zien en zijn betrokken klassen. De betrokken klassen zijn leeg gelaten qua attributen en methodes omdat ze zijn alleen toegevoegd zijn om te laten zien met welke klassen de contact_groep een associatie heeft.



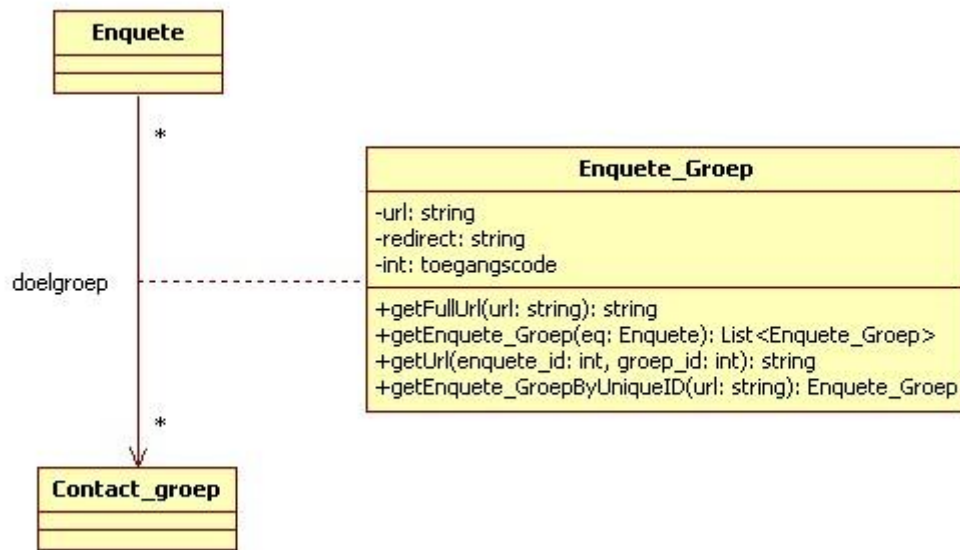
Figuur 35 Contact_groep klasse

Uitleg methodes

In de contact_groep klasse zijn drie methodes te zien die een bepaalde groep of groepen ophalen afhankelijk van de meegegeven parameters. Deze parameters kunnen een enquête of een company zijn. De methode "checkGroupsDropOuts" kijkt naar de groepen van een company of de respondenten een non-reponse of een response is.

Requirement 134: Groepen koppelen aan een enquête.

Normaliter wordt een tussenklasse niet getoond in een klassendiagram wanneer hij geen uniek gedrag vertoont (veel op veel relatie). Maar omdat de klasse attributen en methodes bevat is hij toegevoegd. In figuur 36 is de tussenklasse te zien.



Figuur 36 associatie klasse enquete_groep

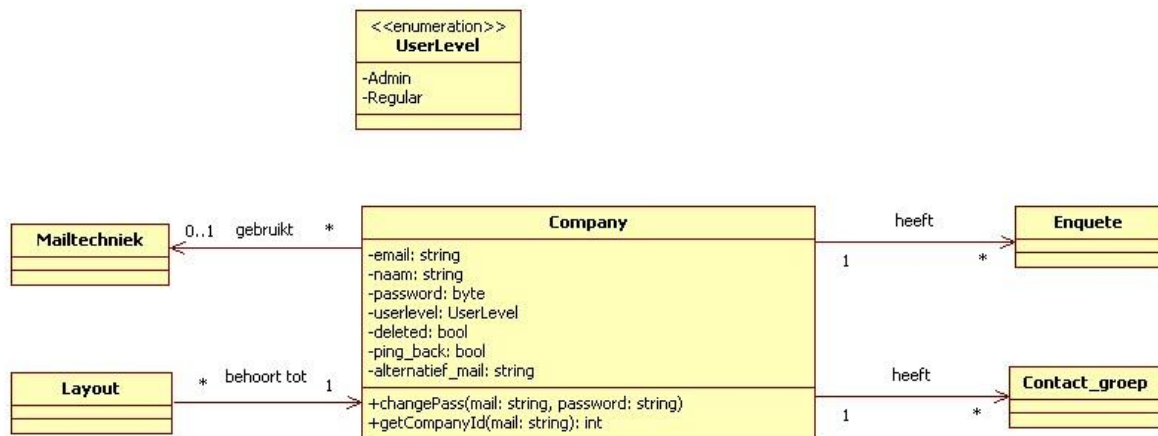
Uitleg methodes

In de enquete_groep klasse zijn twee methodes te zien (getFullUrl en getUrl) die een URL teruggeven van een bepaalde enquête en groep. Verder zijn er nog twee methodes die een enquete_groep of enquete_groepen terug geven aan de hand van meegeven parameters.

8 Bouw onderdeel 2: beheer klanten en beheer Webtechniek

Hoewel tijdens het plan van aanpak en het afstudeerplan beiden beheer onderdelen (beheer onderdeel voor klanten en beheer onderdeel voor Webtechniek) als apart werden beschouwd, is tijdens de Construction fase er voor gekozen om deze twee samen te voegen qua uitvoering. In dit hoofdstuk zal een omschrijving worden gegeven hoe de verschillende requirements die behoren tot dit onderdeel zijn gebouwd. Er zullen maar een aantal requirements aan bod komen, de overige requirements zijn te zien in bijlage E.

Requirement 109 & requirement 114: Registeren van klanten & en beheren van klanten



Figuur 37 Company klasse

Uitleg methodes

Een groot deel van de methodes binnen de company klasse zijn verantwoordelijk voor het manipuleren en controleren van het password van de company. Verder zijn er een aantal methodes die een bepaalde company ophalen, afhankelijk van de meegegeven parameter.

Uitleg Mailtechniek

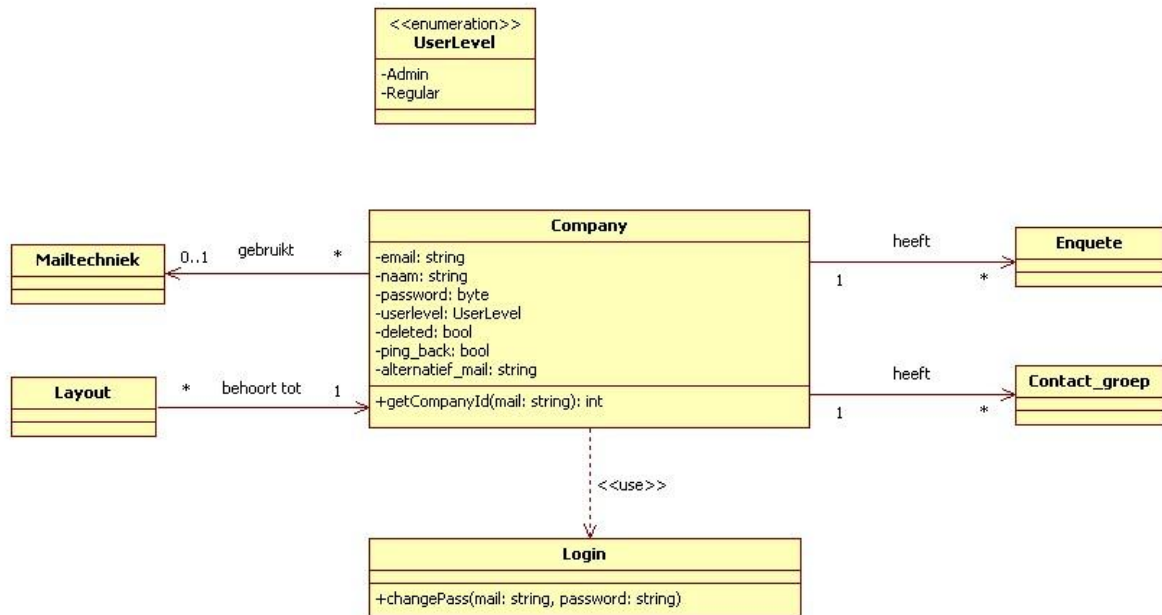
Wanneer een company (gebruiker) een bestaande account heeft bij mailtechniek zullen deze gegevens worden ingevuld tijdens het aanmaken van company. Deze gegevens worden bijgehouden in de mailtechniek klassen, deze gegevens worden vervolgens gebruikt om met de mailtechniek API te communiceren om mail te versturen. Wanneer een bepaalde company een enquête wilt versturen via mail wordt er dus gecheckt of hij een mailtechniek instantie bevat, is dit het geval kan er verbinding worden gemaakt met mailtechniek door de mailtechniek klassen te gebruiken. Meer informatie hierover zal ik geven tijdens Hoofdstuk 9.

Uitleg Userlevel

Om een onderscheid te maken tussen een normale gebruiker (de klanten) en de medewerkers van Webtechniek die in het systeem de gebruikers kunnen beheren, heb ik de verschillende userlevels toegevoegd "regular" voor de gebruikers en "admin" voor de medewerkers van Webtechniek.

Toevoeging Login klasse.

Binnen de company klasse zitten twee verschillende verantwoordelijkheden (lage cohesion) namelijk de methodes die over het login gedeelte gaan en het company gedeelte. Ik heb ik er voor gekozen om deze twee te scheiden wat ten goede komt aan de cohesion. De methodes die behoren tot de company klasse blijven uiteraard in de company klasse maar de methodes die tot het login gedeelte behoren zijn toegevoegd aan de klasse Login. Er is duidelijk te zien dat er maar één password en email per company beschikbaar is, dit is een bewuste keuze die is gemaakt op basis van de bedrijfsregel uit §6.1.4 (per klant is maar één account beschikbaar). De nieuwe situatie is te zien in figuur 38.



Figuur 38 scheiding van login verantwoordelijkheden

9 Bouw onderdeel 3: Distribueren enquête

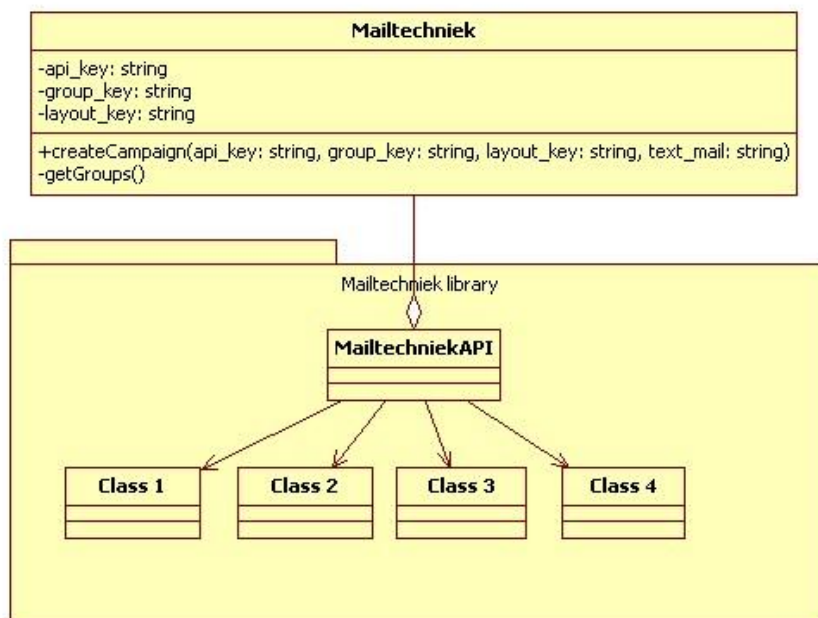
Als derde onderdeel kwam het distribueren van enquêtes aan bod, dit houdt kortweg in het verspreiden van de enquête door verschillende media en het tonen ervan naar de respondenten toe. In dit hoofdstuk zal een omschrijving worden gegeven hoe de verschillende requirements die behoren tot dit onderdeel zijn gebouwd. Er zullen maar een aantal requirements aan bod komen de overige requirements zijn te zien in bijlage E.

Requirement 80: Distribueren via mail

Om enquêtes te kunnen distribueren door middel van mail zal ik gebruik maken van Mailtechniek (requirement 118), dit is een applicatie ontwikkeld door Webtechniek voor het versturen van nieuwsbrieven en mails op grote schaal. De ontwikkelaar van Mailtechniek heeft een API geschreven, zodat andere applicaties gebruik kunnen maken van Mailtechniek.

Een nieuwe klasse Mailtechniek is toegevoegd aan het klassendiagram, die de attributen `api_key`, `group_key`, `layout_key` heeft, deze attributen zijn nodig om verbinding te kunnen maken met de mailtechniek API. De company klasse zal hierdoor een instantie van de mailtechniek klasse hebben die alles afhandelt met de API. De klassendiagram van het betreffende onderdeel is te zien in figuur 39.

Zoals te zien is, wordt er een losse koppeling en tevens abstractie gecreëerd. Doordat er een simpele interface ontstaat (mailtechniek) die de complexiteit van de API niet zichtbaar maakt voor de rest van de applicatie. Uiteindelijk houdt de API de complexiteit van mailtechniek achter.



Figuur 39 Koppeling mailtechniek

Requirement 82 social media

Om de enquêtes te kunnen distribueren via social media is tijdens de elaboration fase in eerste instantie de beslissing genomen om gebruik te maken van een aantal API's van bijvoorbeeld twitter en facebook. Om vervolgens een facade klasse te schrijven die de verschillende API's gebruikt. Tijdens de elaboration fase kwam echter naar voren dat er een simpelere manier is om een link te delen via de verschillende social media platformen. Naast de verschillende API's leveren Facebook, Twitter en LinkedIn een "share" optie. Via deze share optie kan een meegeven URL worden gedeeld. Wanneer de gebruiker een enquête wil distribueren van een bepaalde enquête en groep zal de URL worden opgehaald en vervolgens in de URL van Facebook, Twitter of LinkedIn worden geplaatst. Als gevolg hiervan is er een per socialmedia platform een method toegevoegd die de link bevat, vervolgens kan de URL worden gedeeld. Een voorbeeld hiervan is te zien in figuur 40.



Share this Link

Share: On your own Wall Friends

Write Something...

<http://enquete.ontwikkeltechniek.nl/486265>

Share Link Cancel

Figuur 40 delen via facebook

Requirement 83: link enquête

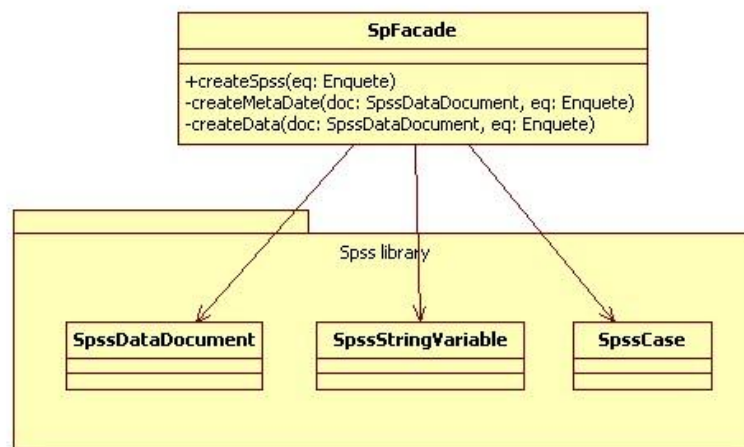
In eerste instantie werd er gekozen om de URL's van de enquêtes weer te geven op de volgende manier: Klant (naam)/Enquête (naam)/contact_groep (naam).Echter bleek deze URL niet geheel naar behoren, dit kwam omdat de klant van het enquête systeem niet altijd overeen komt met de doelgroep waaraan een enquête is gericht. Om een voorbeeld hiervan te noemen: www.enquetetechniek.nl/Piet/Klanttevredenheid/Facebook, in deze link is duidelijk te zien dat Piet niet duidelijk weergeeft wat voor bedrijf het is. Om dit probleem te verhelpen is na overleg met de opdrachtgever gekozen om de URL te laten bestaan uit het volgende: www.enquetetechniek.nl/klant (id), enquête (id), groep (id) en een random nummer. Oftewel een lang nummer die altijd uniek is.

10 Bouw onderdeel 4: Resultaten enquête

Als laatste onderdeel kwam het tonen van de resultaten van enquêtes aan bod, dit houdt kortweg in het tonen van de antwoorden die de respondenten hebben gegeven op de vragen binnen een enquête. In dit hoofdstuk zal een omschrijving worden gegeven hoe de verschillende requirements die behoren tot dit onderdeel zijn gebouwd. Er zullen maar een aantal requirements aan bod komen de overige requirements zijn te zien in bijlage E.

Requirement 103: SPSS

Het bedrijf dat SPSS heeft ontwikkeld is IBM. Zij leveren dan ook een aantal hulpmiddelen voor verschillende frameworks en programmeertalen. Na het testen van de library die wordt geleverd voor ASP.NET[IBM SPSS], werkte het niet naar behoren (het definiëren van variabelen ging niet goed). Ik heb dan ook verder gekeken naar andere oplossingen en een andere library gevonden, deze is specifiek ontwikkeld voor het lezen en creëren van .sav files (de extensie van SPSS) [SPSS library]. Tevens is het een stabiele versie en ondersteunt het SPSS versie vijftien of hoger. Omdat ik alleen een .sav file moest creëren, heb ik een facade klasse geschreven die een aantal klassen uit de library gebruikt, hierdoor “weet” maar één klasse over de functionaliteiten van de library en de rest van enquêteteknik niet.

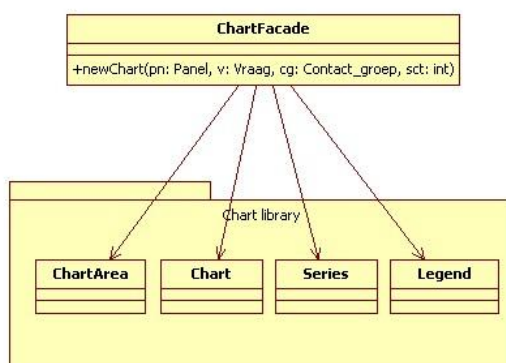


Figuur 41 SPSS facade pattern

In de methodes van figuur 41 wordt als eerst een nieuw .sav bestand aangemaakt, indien hij al aanwezig is wordt die verwijderd (de resultaten kunnen immers zijn veranderd). Vervolgens worden de variabelen aangemaakt in de createmetadata methode.

Requirement 96: Grafieken/Charts

Om de resultaten van een vraag grafisch weer te geven zal er gebruik worden gemaakt van een aantal grafieken. Tijdens de elaboration fase is de keuze gemaakt om van de ASP.NET chart library gebruik te maken. Omdat het ASP.NET framework over een eigen chart library beschikt leek mij dit de beste keuze, deze is namelijk het meest “compatible” met het ASP.NET framework en gratis qua kosten, overige chart tools of libraries zijn dit niet. Om de library op een juiste manier te implementeren heb ik een eigen facade klasse geschreven die gebruik maakt van een beperkt deel van de chart library om vervolgens grafieken te kunnen maken.



Figuur 42 chart facade pattern

10.1 Klanten interviews

Hoewel de interviews met de klanten eerder waren gepland (namelijk in de inception fase) zijn deze aan het einde van de Construction fase gehouden. Omdat de interviews pas op dit moment werden gehouden heb ik het doel van dit interview ook aangepast. In eerste instantie wilde ik de interviews met de klanten gebruiken om requirements te ontlokken, nu zal de nadruk vooral komen te liggen op het toetsen van bestaande requirements (waaruit eventueel nieuwe requirements komen). In deze paragraaf is te zien wie zijn ondervraagd, hun antwoorden en de resultaten (de contactgegevens van de klanten worden niet getoond in verband met de privacy van de klanten).

10.1.1 benaderde klanten

Naam persoon	Bedrijf/instelling/functie	Gereageerd?
Nannette Verschoor	Nota Bene	Nee
Lia Westenberg	HCD/HCK	Ja
Lydia Blaauw	Syndion	Ja
Victor van Rijn	Webstudio MM	Nee
Dries Verwaaijen	Overstag consult	Nee
Mariette van Zijl	Van Zijl PR & Communicatie	Nee
Martijn Blaauw	Bedrijfskundig Marketing Adviseur	Ja
Remco Legerstee	Salesmanager	Ja

10.1.2 Antwoorden

Vraag	Lia Westenberg	Lydia Blaauw	Martijn Blaauw	Remco Legerstee
1.Welke vraagsoorten zou u willen maken in het enquêtesysteem?	Meerkeuze vragen.	Meerkeuze vragen.	Meerkeuzevragen, openvragen Tabellen, likert schaal, OS-good schaal, tegenstellingen, matrix en tekstveld.	Meerkeuze vragen, open vragen en tabellen.
2.Via welke manieren zou u de enquête willen distribueren?	Mail, link op u site.	Mail.	Mail, social media, link op u site en aparte link.	Mail, link op u site.

3.Via welke manieren zou u de resultaten van de enquête willen exporteren?	Excel.	Grafieken.	SPSS, CSV, HTML.	Excel en CSV.
4.Zou u enquêtes gebruiken om gevoelige informatie en/of persoonlijke informatie te verzamelen?	Nee.	Nee.	Ja en nee.	Nee.
5.Zou u enquêtes gebruiken voor een gesloten of open groep?	Gesloten groepen.	Open groepen.	Gesloten en open groep.	Gesloten groepen.
6.Heeft u een idee over functionaliteiten die u graag zou willen aantreffen in een enquêtesysteem?	Nee.	Grafische weegaven van de uitkomsten.	Afbeeldingen video's toevoegen.	Probeer het zo generiek mogelijk te maken + mogelijkheid tot het maken van selecties.
7.Heeft u overige opmerkingen waarmee tijdens de ontwikkeling van het enquête systeem rekening gehouden moet worden?	Nee.	Nee.	Nee.	Probeer iets meer uit de gegevens te krijgen, een optie tot "wilt u ook onze nieuwsbrief gaan ontvangen" aan het einde v/d enquête maakt het mogelijk om gelijk een ledenbestand te vergroten.

10.1.3 Resultaten

In de resultaten van de interviews is te zien dat de klanten elkaar nogal tegenspreken over wat ze willen met het enquêtesysteem. Met name vraag 1-3 en 6 komen tegensprekende antwoorden naar voren, ik zal kort uitleggen hoe ik dit heb aangepakt.

Vraag 1: In deze vraag is duidelijk te zien dat de klanten meerkeuze vragen willen, deze zijn allemaal al geïmplementeerd. Hiernaast komen er een aantal nieuwe vraagsoorten die zijn toegevoegd als requirements, dit waren de volgende:

- Likert schaal
- OS-good schaal ook wel Semantisch differentiaal genoemd.
- Tegenstellingen

Vraag 2: Uit de resultaten van deze vraag is duidelijk te zien dat het distribueren via papier en sms geen toevoeging is aan het systeem. Hierdoor zijn de requirements die hierop betrekking hebben naar de categorie won't haves geplaatst. De overige requirements zijn wel al gebouwd.

Vraag 3: Uit de resultaten van deze vraag is duidelijk te zien dat het exporteren via powerpoint niet nodig is, de requirements die hierop betrekking hebben zijn naar de categorie won't haves geplaatst. De overige requirements zijn al gebouwd.

Vraag 4: Deze vraag was gesteld om te controleren of de gebruikers het enquêtesysteem zouden gebruiken om gevoelige informatie te ondervragen. Eerder in paragraaf §6.4.4 kwam dan ook de tegenstrijdige requirements naar boven die over het HTTP en HTTPS protocol gingen, hier is toen de keuze gemaakt om het HTTP protocol te gebruiken. De resultaten uit deze interviews bevestigen dit, want een groot deel wil alleen de enquêtes gebruiken voor geen gevoelige informatie.

Vraag 5: Geen opmerkingen.

Vraag 6: Zo generiek mogelijk qua vraagtypes.

Vraag 7: Mogelijkheid tot ontvangen van een nieuwsbrief aan einde van enquête. Dit is mogelijk door twee vragen te stellen namelijk, wilt u een nieuwsbrief ontvangen ja/nee? En wat is u e-mailadres?

11 Testen

Tijdens de bouw van de verschillende onderdelen was er al begonnen aan het testtraject. Door gebruik te maken van een gestructureerd testtraject dat is afgesteld op dit project (weergegeven in §4.5.1), heb ik op een doelgerichte manier het testtraject van testgoal ingeperkt. Het testtraject bestaat uit drie hoofdonderdelen namelijk: het mastertestplan, testontwerp en testrapport.

11.1 Mastertestplan

Het testtraject begon met het opstellen van het mastertestplan. In het mastertestplan werd de aanpak van het testen duidelijk opgesteld. Een onderdeel hiervan was onder meer de planning voor het testtraject, deze loopt gelijk met het bouwen van de verschillende onderdelen. Hierdoor kunnen er module testen worden uitgevoerd nadat een bepaald onderdeel van de applicatie is gemaakt. Bij het opstellen van de planning is er een duidelijk onderscheid gemaakt tussen de twee verschillende testsoorten die ik zal toepassen namelijk de module en de systeemtest.

Beperkte uitvoer module testen

Omdat er maar een beperkte tijd beschikbaar is voor het testtraject heb ik er voor gekozen om de module testen informeel te houden. Dit wil zeggen dat ik geen documentatie heb opgesteld voor het uitvoeren van de module testen. De moduletesten kunnen geautomatiseerd worden uitgevoerd. Dit heeft als doel om de werking van een onderdeel via herhaalbare wijze te testen. De moduletesten hoeven maar één keer te worden geschreven, waarna het op elk willekeurig moment kan worden getest. Hierdoor kan bij een wijziging worden getest of alles nog naar behoren werkt.

11.2 Testrisicoanalyse

Omdat het aantal functionaliteiten groot is, is het onmogelijk om tijdens de resterende tijd alles te testen. Het antwoord hierop is simpel, er zal slechts een beperkt deel van de applicatie intensief worden getest. De overige onderdelen zullen op een informele manier worden getest waardoor de belangrijkste fouten er worden uitgefilterd. Omdat er maar een deel kan worden getest zal er goed worden gekeken welke dit zal zijn. De testrisicoanalyse vormt hierbij het hulpmiddel. Het doel van een testrisicoanalyse is om de onderdelen van de applicatie te testen waar het risico het hoogste is. Normaliter wordt er een testboom opgesteld waarin alle functionaliteiten worden opgedeeld en vervolgens worden de betrokkenen (stakeholders) gevraagd om de taken van de testboom te voorzien van een prioritering.

Omdat hierin veel tijd gaat zitten en er beperkte tijd beschikbaar is voor het testtraject heb ik er voor gekozen om zelf te bekijken waar de meest kritische punten zitten binnen het systeem. Hieruit kwamen de volgende functionaliteiten:

- Aanmaken enquête: Kritisch
- Aanmaken vraag: Hoog
- Aanmaken antwoord: Midden

Op meer kritische functionaliteiten worden zwaardere testontwerptechnieken toegepast de volgende verdeling is gemaakt:

	CRUD	Grenswaardenanalyse	Beslissingstabellen test	Statement coverage	Equivalentieklassen	Syntax test	Use case test
Kritisch	X	X	X				
Hoog		X	X				
Midden				X	X	X	
Laag							X

Resultaat	Keuzerondjes, verplichte vraag en geen volgende stap	Waardingsschaal , geen verplichte vraag en volgende stap	Datum, verplichte vraag en volgende stap	Fileupload, geen verplichte vraag en geen volgende stap	Constantesom, verplichte vraag en geen volgende stap
------------------	--	--	--	---	--

Ongeldige grenswaarden

	Geval 6	Geval 7	Geval 8	Geval 9	Geval 10
Verplicht	J	N	J	N	J
Stap	N	J	J	N	N
Vraagtypes	-1	10	16	100	101
Resultaat	Error	Error	Error	Error	Error

CRUD matrix

Testbasis: requirements

Requirement 5: De klant wil een titel geven aan een enquête bij het aanmaken van een enquête.

Requirement 7: De klant wil een vraag maken, tijdens het aanmaken van een enquête.

Requirement 8: De klant wil aangeven of de vraag verplicht is of niet, tijdens het aanmaken van een vraag.

Requirement 10: De klant wil aangeven of de vraag op een nieuwe pagina wordt getoond, tijdens het aanmaken van een vraag.

Requirement 15-29,130,140-142: Vraagtypes.

Requirement 39: De klant wil een antwoord op een vraag verwijzen naar een andere vraag waardoor conditionele vertakking mogelijk is, tijdens het aanmaken van vragen.

	Enquete	Vraag	Antwoord
Titel	RU		
Vraag aanmaken	RU	CRU	CRUD
Verplicht		RU	
Nieuwe pagina		RU	
Vraagtypes		RU	
Conditionele vertakking		RU	RU

Logisch testgeval: Enquete

Functie	CRUD	Actie
Titel	-	Create een nieuwe enquête EQ1.
Titel	R	Check of EQ1 bestaat.
Titel	U	Voeg de titel aan EQ1 toe.
Vraag aanmaken	R	Lees EQ1.
Vraag aanmaken	-	Create een nieuwe vraag V1.
Vraag aanmaken	U	Voeg de V1 aan de EQ1 toe.

Logisch testgeval: Vraag

Functie	CRUD	Actie
Vraag aanmaken	C	Create een nieuwe vraag V1.
Vraag aanmaken	R	Check of V1 bestaat.
Verplicht	R	Check of V1 bestaat.
Verplicht	U	Update V1 naar V1A.
Nieuwe pagina	R	Check of V1A bestaat.
Nieuwe pagina	U	Update V1A naar V1B.
Vraagtypes	R	Check of V1B bestaat.
Vraagtypes	U	Update V1B naar V1C.
Conditionele vertakking	R	Check of V1C bestaat.
Conditionele vertakking	U	Update V1C naar V1D.

Logisch testgeval: Antwoord

Functie	CRUD	Actie
Vraag aanmaken	-	Create een nieuwe vraag V1.
Vraag aanmaken	-	Check of V1 bestaat.
Vraag aanmaken	C	Create een nieuw antwoord A1.
Vraag aanmaken	R	Check of A1 bestaat.
Vraag aanmaken	U	Voeg A1 aan V1 zijn antwoorden toe.
Vraag aanmaken	C	Create een nieuw antwoord A2.
Vraag aanmaken	R	Check of A2 bestaat.
Vraag aanmaken	U	Voeg A2 aan V1 zijn antwoorden toe.
Vraag aanmaken	D	Verwijder A2 van V1 zijn antwoorden.
Vraag aanmaken	-	Update V1 zijn antwoorden.

11.4.1.2 Informele testgevallen

#	Testcase	Reden voor test
1	Creëren enquête (stap 1: algemene info)	Het opstellen is het belangrijkste onderdeel binnen het enquêtesysteem hierdoor zal ik alle stap doorloop op verschillende invulwaardes, verplichtheid en correcte werking.
2	Creëren enquête (stap 2: header en footer)	"
3	Creëren enquête (stap 3: vragen)	"
4	Creëren enquête (stap 4: afsluiting)	"
5	Creëren enquête (stap 5: groepen)	"
6	Creëren enquête (stap 6: groepen)	"
7	Creëren vraag	Omdat er veel opties beschikbaar zijn voor het opstellen van een vraag is het belangrijk om deze verschillende opties te testen en/of de verschillende vraagtypes het juiste gedrag vertonen.
8	Conditionele vertakking	Conditionele vertakking is een belangrijke functionaliteit binnen het enquêtesysteem. Omdat de functionaliteit complex is qua werking is het van belang om te testen of het correct werkt.

11.4.2 Fysieke testgevallen

In het tweede onderdeel van het testontwerp worden de logische testgevallen vertaald naar fysieke testgevallen. Dit komt kortweg neer op het concreet maken van de testen door waardes mee te geven per testgeval. Om een aantal voorbeelden te geven van zowel de formele als informele testgevallen zijn §11.4.2.1 en §11.4.2.2 toegevoegd.

11.4.2.1 Formele testgevallen**Grenswaardeanalyse: Vraag specificaties****Geldige grenswaarden**

	Geval 1	Geval 2	Geval 3	Geval 4	Geval 5
Verplicht	True	False	True	False	True
Stap	False	True	True	False	False
Vraagtypes	Vraag.Vraagtype. Keuzerondjes	Vraag.Vraagtype. Waardingsschaal	Vraag.Vraagtype. Datum	Vraag.Vraagtype. Fileupload	Vraag.Vraagtype. ConstanteSom
Verwacht Resultaat	Keuzerondjes, verplichte vraag en geen volgende stap	Waardingsschaal, geen verplichte vraag en volgende stap	Datum, verplichte vraag en volgende stap	Fileupload, geen verplichte vraag en geen volgende stap	Constantesom, verplichte vraag en geen volgende stap

Ongeldige grenswaarden

	Geval 6	Geval 7	Geval 8	Geval 9	Geval 10
Verplicht	True	False	True	False	True
Stap	False	True	True	False	False
Vraagtypes	0	10	16	100	101
Verwacht Resultaat	Error	Error	Error	Error	Error

CRUD matrix

Van de CRUD matrix wordt één fysiek testgeval getoond, namelijk van de entiteit enquête.

Fysiek testgeval: Enquete**EQ1:**

ID: 1
 Titel: Test
 Start datum: 16-04-
 Eind datum
 Welkomsttekst: Hoi
 Afsluitende tekst: Doe
 Status: InOntwikkeling
 Deleted: False
 Taal: Spaans
 Company_id: 1
 Layout_id: 1

V1:

ID: 1
 Stelling: hoi
 Toelichting: test
 Afbeelding: -
 Type: -
 Enquete_id: -
 Verplicht: false
 Stap: false
 Parent_id: -
 Volgorde: 1

Update

V1: enquete_id: 1

Verwacht resultaat:

Het verwachten resultaat van deze test is dat de attribuut enquete_id van V1 de id heeft van EQ1.

11.4.2.2 Informele testgevallen

Van de informele testgevallen wordt één fysiek testgeval getoond, namelijk creëren enquête (stap 1).

Creëren enquête: invoerwaardes

#	1
Testcase	Creëren enquête (stap 1: algemene info).
Doel	Het testen op mogelijke invoer door de gebruiker.
Input	Om de eerste stap binnen een enquête te testen: <ul style="list-style-type: none"> • Voer titel in: "", "titel", "123" "\$%@". • Voer startdatum in, "", "27/03/2012", "test", "123" en "@#\$". • Voer einddatum in "", "27/03/2012", "27/04/2012", "test", "123" en "@#\$".

- Voer welkomsttekst in: "", "hoi", "123", "#\$&".
- Voeg een welkomstafbeelding toe.
- Klik op volgende.

Verwachte uitkomst Er wordt een nieuwe enquête aangemaakt en de ingevoerde waardes worden er aan toegevoegd. De volgende stap wordt getoond.

Commentaar Bij het invullen van de titel, start, einddatum en welkomsttekst wordt getest op verschillende invoer en of deze juist wordt afgehandeld.

11.5 Systeemtest

Nu dat een groot deel van het systeem is ontwikkeld kan de systeemtest worden uitgevoerd. De systeemtest heeft als doel om het totale systeem te testen en zijn functionaliteiten.

De basis voor de systeemtest is gelegd in §11.4.1 en §11.4.2, hierin zijn namelijk de testgevallen gedefinieerd. Ik zal in deze paragraaf kort beschrijven hoe de systeemtest is verlopen, de daadwerkelijke resultaten en vervolgacties zijn gedocumenteerd in het testrapport, een aantal resultaten zijn te zien in §11.6.

Uitvoer systeem

Er is in totaal 2 weken ingepland voor de systeemtest (week 13 en 14). Twee weken bleek ruim voldoende, de meeste tijd zat in het opstellen van de testen (testontwerp) en het verwerken van de resultaten (testrapport). Voor het uitvoeren van de testen bleek een paar dagen genoeg te zijn.

Niet functionele requirements

Naast de functionaliteiten kunnen er ook niet-functionele requirements worden getest tijdens een systeemtest. Omdat er niet genoeg tijd beschikbaar is om zowel de functionele requirements als de niet-functionele requirements te testen heb ik er voor gekozen om de systeemtest te beperken tot de functionele requirements. Mede doordat het eind product een prototype is kan er na de afstudeerperiode een gebruikeracceptatietest worden gehouden om via deze manier de gebruikersvriendelijkheid te testen, hiernaast kunnen er testen worden opgesteld om bijvoorbeeld de performance te testen.

11.6 Testrapport

Tijdens het uitvoeren van de systeemtest is er een testrapport opgesteld hierin zijn alle resultaten gedocumenteerd en de vervolg acties. Om een voorbeeld te geven van een aantal resultaten en hun bijhorende vervolgacties zijn §11.6.1 en §11.6.2 toegevoegd.

11.6.1 Formele testgevallen

Grenswaardeanalyse: Vraag specificaties

Testgeval	Resultaat	Vervolgactie
1	Verwacht resultaat	Geen
2	Verwacht resultaat	Geen
3	Verwacht resultaat	Geen
4	Verwacht resultaat	Geen
5	Verwacht resultaat	Geen
6	Error	Omdat het alleen mogelijk moet zijn om een vraag aan te maken wanneer een vraagtype wordt gekozen is er een controle toegevoegd die checkt of er daadwerkelijk een vraagtype is gekozen.
7	Error	Hoewel programmatisch deze error kan worden veroorzaakt is het niet mogelijk voor de gebruiker om een vraagtype te kiezen die niet bestaat. Tevens is het niet handig om af te vangen op deze waardes omdat in de toekomst mogelijk nieuwe vraagtypes worden toegevoegd.
8	Error	Zie vervolg actie 7.
9	Error	Zie vervolg actie 7.
10	Error	Zie vervolg actie 7.

11.6.2 Informele testgevallen

#	1
Testcase	Creëren enquête (stap 1).
Uitkomst	<ul style="list-style-type: none">• Het afvangen op vreemde karakters zoals “%\$” wordt nog niet juist afgevangen, bij alle invulvelden.• De invulvelden zijn nog niet verplicht en kunnen worden overgeslagen terwijl een titel, startdatum, einddatum en welkomstekst verplicht moeten zijn.• De startdatum moet eerder zijn dan de einddatum.
Vervolg actie	Binnen de bestaande code zijn extra afvangen toegevoegd waarin wordt afgevangen op vreemde karakters, verplichtheid en of de startdatum eerder is dan de einddatum.

#	5
Testcase	Creëren enquête (stap 5: groepen).
Uitkomst	<ul style="list-style-type: none">• Wanneer een groep niet word geselecteerd maar wel bestaat binnen de database, word deze niet correct verwijderd.
Vervolg actie	Binnen de bestaande code is een extra stukje toegevoegd die een groep verwijderd wanneer hij niet meer geselecteerd is (maar voorheen wel bestond).

#	6
Testcase	Creëren enquête (stap 6: definitief).
Uitkomst	<ul style="list-style-type: none">• De optie om een enquête definitief te maken moet alleen mogelijk zijn als de enquête vragen, een layout en een groepen bevat.
Vervolg actie	Binnen de bestaande code is een extra controle toegevoegd waarin wordt afgevangen of de enquête een vraag, layout en groep bevat.

12 Evaluatie

In dit hoofdstuk zal ik terug kijken op het afstudeerproject. Dit zal door middel van een aantal verschillende onderwerpen gebeuren. Zou zal ik eerst de aanpak evalueren om vervolgens de producten te evalueren die zijn gemaakt tijdens dit project. Tot slot komen de competenties aan bod.

12.1 Aanpak

Planning

De uitvoering van het project aan de hand van de planning liep in de hoofdlijnen zoals verwacht. Door middel van de planning kreeg mijn project een duidelijke structuur en had ik een rode draad om te volgen. De verschillende fases en disciplines van RUP gaven een duidelijke verdeling binnen het project waardoor per fase een aantal producten werden opgeleverd door middel van een rapport per fase. Tijdens het project is de planning meerdere keren aangepast zoals te zien is in §5.3, §6.1.1 en §11.3.

Gebruik RUP

Het gebruik van RUP binnen dit project is goed gegaan. Het heeft mij een houvast gegeven om de planning op te stellen en het project te controleren. Door het project op te delen in de verschillende fases en disciplines van RUP werd het al snel duidelijk welke producten wanneer moesten worden gemaakt en binnen welke fase.

Door op een incrementele en iteratieve manier RUP te gebruiken heb ik de gehele applicatie kunnen opdelen in logische onderdelen en was het mogelijk om de applicatie langzaam vorm te geven. Omdat het product veel overeenstemming met verschillende stakeholders nodig had, kwam deze manier van werken goed te pas. Zo zijn er requirements gedurende het volledige project aangepast, verwijderd of toegevoegd (§6.1.5).

De keuze om een beperkt deel van RUP toe te passen heeft goed uitpakt. Het gebruik van alleen de fases, disciplines, incrementele en iteratieve werkwijze bleek voldoende. Dit kwam vooral omdat dit een relatief klein project was qua duur en betrokken personen. Wanneer het project een langere doorlooptijd zo hebben en er meer mensen bij betrokken zouden zijn had ik RUP op een uitgebreidere manier toepast (zoals het toekennen van rollen).

Tijdens het houden van het feedback moment met mijn begeleider kwam naar voren dat het gebruik van RUP binnen de planning verkeerd was gebruikt, zo had ik naar mijn mening de planning opgedeeld aan de hand van fases echter bleek dit aan de hand van disciplines te zijn. Als gevolg hiervan was de planning qua notatie aangepast.

Gebruik de Swart, MoSCoW en ISO 9126

De keuze om de requirements op een uitgebreide manier onder te verdelen aan de hand van de Swart, ISO 9126 en te prioriseren door middel van MoSCoW (§4.2) heeft gedurende het project voordelen gehad. Zo werd tijdens het ontwerpen van de applicatie duidelijk welke requirements als eerst werden ontworpen en kon tijdens het bouwen de focus op een bepaald onderdeel worden gelegd. Omdat er veel requirements ontstonden was het soms lastig om overzicht te houden, vooral de MoSCoW analyse heeft mij hierbij geholpen om toch enigszins overzicht te behouden.

Gebruik testgoal

Het gebruik van testgoal is beperkt gebleven, dit kwam vooral omdat alleen de onderdelen uit het algemene testtraject van testgoal zijn gekozen die voor mijn project relevant waren (§4.5.1). Om te bekijken welke testsoorten tijdens het testtraject werden uitgevoerd is het V-model gebruikt (§4.5.2), dit bleek een handig hulpmiddel om snel een keuze te maken uit de mogelijke testsoorten die pasten bij mijn project. Doordat er niet veel tijd beschikbaar was en de applicatie niet werd geïmplementeerd is er gekozen om alleen moduletesten en een systeemtest uit te voeren.

Met het opstellen van de systeemtest had ik vooral moeite omdat eerder tijdens de testrisicoanalyse werd vastgesteld dat er maar één onderdeel werd getest namelijk het onderdeel waarin enquêtes worden gecreëerd. Dit spreekt echter het doel van de systeemtest tegen, want hierin wordt namelijk het gehele systeem getest. Omdat er geen tijd is om testgevallen op te stellen voor het gehele systeem is er voor gekozen om de systeemtest te beperken tot één onderdeel.

Gebruik UML en OTAP

Tijdens het ontwerpen van de verschillende functionaliteiten is er gebruik gemaakt van UML. De keuze om maar een beperkt aantal diagrammen te ontwerpen is een goede keuze geweest (§4.3). Hierdoor kon de nadruk worden gelegd op ontwerpen die voor mij en het bedrijf het meest duidelijk waren. De meest gebruikte diagram is de klassendiagram geweest, dit komt hoofdzakelijk omdat dit soort diagram op een niet al te gedetailleerd niveau toch veel over de implementatie laat zien en hierdoor goed kan worden gebruikt om te communiceren met de stakeholders binnen Webtechniek.

Tijdens het bouwen van de verschillende onderdelen is er gebruik gemaakt van OTAP (§5.5). Achteraf bleek dat de acceptatie en de productie omgeving helemaal niet werden gebruikt, dit kwam vooral omdat de applicatie uiteindelijk niet is geïmplementeerd. De ontwikkel en test omgeving daarin tegen zijn wel veel gebruikt gedurende dit project om de applicatie te ontwikkelen en te testen.

12.2 Producten

Inception rapport

Over het inception rapport ben ik tevreden. Het inception rapport bestaat uit een verzameling van producten die zijn opgesteld tijdens de inception fase. Deze producten hebben er voor gezorgd dat het project op een juiste manier werd opgestart en ondersteuning bod tijdens de overige fases. Het plan van aanpak was een aanvulling op het al bestaande afstudeerplan (bijlage A) en diende er hoofdzakelijk voor om overeenstemming tussen mij en de opdrachtgever te creëren over de opdracht (§5.1). De keuze om een vooronderzoek te houden naar zowel terminologie en bestaande functionaliteiten heeft een goede basis gelegd voor de requirements (§5.2). Het houden van interviews daarin tegen verliep anders dan ik had verwacht, zo werden de klanten pas laat in het ontwikkeltraject geïnterviewd. Omdat het interview pas laat werd gehouden heb ik het doel van dit interview veranderd (veel functionaliteiten waren al gebouwd), het werd een soort controle interview met de vraag of alle functionaliteiten er al inzaten? Uit dit interview kwamen een aantal interessante resultaten die ik vervolgens heb verwerkt (§10.1).

Elaboration rapport

Over het elaboration rapport ben ik ook tevreden. Het opstellen van de requirements en ontwerpen verliep goed, alleen door de grote omvang was het soms lastig om overzicht te behouden en de requirements consistent te houden over de verschillende documenten. De keuze om de architectuur in een beperkte vorm te omschrijven is een juiste keuze geweest die mij tijd winst heeft opgeleverd, hierdoor werd er niet veel tijd verspilt aan het omschrijven van viewpoints die niet van belang waren voor dit project. De vraag van de opdrachtgever om een volledige ontwerp te maken (in de vorm van gedetailleerde ontwerpen) heeft invloed gehad op dit project. Omdat ik echter tijd had bespaart tijdens het opstellen van de requirements en het ontwerpen van de architectuur had dit verder geen grote gevolgen voor de planning (§6.1.1).

Construction rapport

In eerste instantie had ik in het construction rapport code voorbeelden getoond van de gerealiseerd requirements. Tijdens het houden van het feedback moment met mijn begeleider kwam naar voren dat dit niet de correcte manier is van documenteren binnen de construction fase. In plaats van code voorbeelden heb ik meer gedetailleerde ontwerpen toegevoegd die bijvoorbeeld attributen en methodes tonen, deze zijn te zien in hoofdstuk 7 t/m 10.

Transition rapport

Het transition rapport is beperkt gebleven qua inhoud vergeleken met de overige rapporten. Dit kwam vooral omdat er alleen werd getest tijdens de laatste fase van RUP en dus het implementatie onderdeel verviel. Wat betreft het testen vind ik dat de aanpak goed is verlopen en de keuze om een beperkt onderdeel te testen juist is geweest. Echter het uitvoeren van de testen verliep soms moeizaam dit kwam vooral omdat ik in eerste instantie mij teveel vasthield aan de formele testtechnieken die in bepaalde situaties niet geschikt bleken te zijn. Het toevoegen van informele testgevallen bleek een goede keuze er kwamen namelijk een aantal fouten naar boven drijven die anders niet waren gevonden.

Enquêtetechniek

Over het eindproduct van dit project namelijk enquêtetechniek ben ik zeer tevreden. Er is een eerste versie van de applicatie gemaakt en kan nu worden getest door de gebruikers om vervolgens in gebruik te worden genomen.

Het ontwikkelen van enquêtetechniek verliep zonder grote tegenslagen. Doordat er een volledige ontwerp is gemaakt konden de functionaliteiten snel worden geïmplementeerd. Tijdens het ontwikkelen zijn er een aantal feedback momenten gehouden om eventuele aanpassingen te maken en onjuiste requirements/ontwerpen te verbeteren.

12.3 Beroepstaken

Uitvoeren van analyse door definitie requirements (niveau 4)

Het onderzoeken en ondervinden van de requirements is goed verlopen. Dit kwam omdat ik een gestructureerde aanpak heb toegepast. Als eerst heb ik vooronderzoek gedaan naar bestaande enquêtesystemen (§5.2), met als doel om te kijken welke functionaliteiten zij aanbieden. Hieruit kwam een lijst met functionaliteiten. Voordat ik deze zo neerleggen bij de stakeholders binnen Webtechniek heb ik eerst een open interview gehouden om hun met eigen ideeën te laten komen (§5.3.1). Vervolgens heb ik de lijst met functionaliteiten neergelegd bij de stakeholders van Webtechniek, om deze zowel te bespreken als te prioriteren (§5.3.2).

Om de requirements te kunnen opstellen moesten de functionaliteiten vertaald worden naar functionele requirements. Om de requirements verder op te delen heb ik de verdeling van de Swart gebruikt. Hierdoor ontstonden de volgende soorten requirements: Functionele gebruikers requirements, niet-functionele requirements, bedrijfsregels en technische beperkingen (§6.1.4). Deze verschillende soorten werden tijdens de elaboration fase voor verschillende ontwerpen gebruikt. Zo werden de bedrijfsregels en technische beperkingen vooral gebruikt tijdens het ontwerpen van de architectuur en de functionele requirements voor gedetailleerde ontwerpen.

Omdat de interviews met de klanten op zich lieten wachten, heb ik er voor gekozen om de functionele requirements te vertalen naar functionele gebruikers requirements (§6.1.3). Later bleek dat hierdoor het interview met de klanten makkelijker kon worden gehouden omdat de requirements vanuit hun perspectief waren beschreven.

Om te bekijken welke requirements belangrijkere worden bevonden dan andere is een MoSCoW analyse toepast. De uitkomst van deze analyse (namelijk een lijst met geprioriseerde requirements) is gedurende het project gebruikt (§6.1.6).

Tijdens het opstellen en ontwikkelen van het systeem is er voortdurende teruggekoppeld naar de betrokken stakeholders. Via deze werkwijze werden de requirements continue aangepast, dit kwam ten goede aan de kwaliteit van de requirements. Een voorbeeld hiervan zijn de interviews met klanten, omdat de klanten pas laat in het ontwikkeltraject werden geïnterviewd. Werden de resultaten gebruikt om eventuele aanpassingen/aanvullingen te maken binnen de bestaande requirements waardoor de kwaliteit ervan vooruit ging.

Ontwerpen systeemdeel (niveau 3)

Het opstellen van het ontwerp is goed gelukt. In eerste instantie was het idee om een beperkt ontwerp te maken, namelijk alleen van de requirements die binnen de “must have” categorie van de MoSCoW analyse vielen. Maar omdat de opdrachtgever een volledig ontwerp belangrijk vond is er meer tijd in gaan zitten dan verwacht, mede hierdoor werd de planning aangepast (§6.1.1).

De keuze om maar een beperkt aantal UML diagrammen soorten toe te passen is een goede keuze geweest. Er was namelijk voor gekozen om alleen klassen, sequentie en activiteiten diagrammen te ontwerpen. Met deze diagrammen was zowel het bedrijf als ik zeer bekend. De keuze om een beperkte architectuur te ontwerpen is een goede keuze geweest, het heeft mij veel tijd gespaard die ik vervolgens weer in het ontwerp kon steken.

Tijdens het opstellen van het ontwerp is er gekozen om een beperkt ontwerp van de database te maken. De keuze hiervoor is vooral gemaakt omdat er veel tijd in het ontwerpen hiervan gaat zitten terwijl ik zelf genoeg

kennis heb om de vertaalslag van een klassendiagram naar database te kunnen maken. Om het toch in beperkte zin te documenteren is er een RRM gemaakt hierin worden de tabellen kort opgesomd met hun primaire sleutels, vreemde sleutels en attributen (§6.4.6).

Om de kwaliteit van de ontwerpen vast te stellen is er gebruik gemaakt van een aantal review momenten met de teammanager, dit met name omdat hij over veel technische kennis beschikt. Uiteraard zijn er ook review momenten met de opdrachtgever en de technische directeur binnen het bedrijf gehouden.

Bouwen applicatie (niveau 3)

Omdat er veel tijd in is gestoken om een volledig en duidelijk ontwerp te maken verliep het bouwen van de applicatie soepel. Doordat tijdens het opstellen van de requirements naar voren kwam welke requirements als eerst moesten worden gerealiseerd (MoSCoW) heeft dit mij geholpen om structuur te brengen in de grote hoeveelheid te realiseren requirements. De complexiteit zat vooral in het onderdeel creëren van enquêtes, door het toepassen van verschillende design patterns konden er bijvoorbeeld makkelijk nieuwe vraagtypes worden toegevoegd en werd het gedrag gescheiden per vraagtype. In hoofdstuk 7 tot en met 10 wordt er per onderdeel (van enquêteteknik) de bouw toegelicht.

Uitvoeren van en rapporteren over het testproces (niveau 3)

Met het testen van de applicatie had ik enigszins moeite dit zat vooral in het ontwerpen en uitvoeren van de testen. Het opzetten van het testtraject en de keuze om een beperkt deel van het testtraject van testgoal toe te passen is een goede keuze geweest. Hierdoor ontstond er een testtraject dat perfect paste bij mijn project. Ook de keuze om gebruik te maken van het V-model was een correcte keuze, mede hierdoor kon er een keuze worden gemaakt welke testsoorten ik zou gaan toepassen.

Het opstellen van een testontwerp en het uitvoeren van de testen verliepen moeizamer. Dit kwam vooral omdat ik mijzelf in eerste instantie vasthield aan de officiële testontwerptechnieken terwijl voor sommige situaties deze niet geschikt waren. Om toch bepaalde functionaliteiten te testen heb ik een aantal informele testen opgesteld die bijvoorbeeld situaties creëren die door de gebruiker konden worden veroorzaakt. Het rapporteren van alle resultaten is goed verlopen, zo zijn er met name tijdens het uitvoeren van de informele testgevallen interessante fouten naar boven gekomen die vervolgens zijn opgelost.

Door de beperkte tijd was er geen ruimte meer om de niet-functionele requirements te testen en ook kon er geen gebruikeracceptatietest worden uitgevoerd. Na de afstudeerperiode ben ik van plan deze beiden uit te voeren om zo tot een volledig product te komen die in gebruik kan worden genomen.

Verklarende woordenlijst

Activiteitendiagram

Dit diagram geeft het verband tussen verschillende activiteiten of bewerkingen weer.

ASP.NET

Active server pages. ASP.NET is een manier om op een webserver webpagina's aan te maken door middel van code zoals HTML en C#.

API

API. Dit is een verzameling van methodes waarmee een computerprogramma kan communiceren met een ander programma of onderdeel.

Webapplicaties

Dit zijn applicaties die gebruik maken van het internet, hierbij komen andere eigenschappen kijken dan bij een client applicatie zoals HTML en web servers.

MVC

MVC is een applicatie model die staat voor model, view en controller elk onderdeel heeft zijn eigen verantwoordelijkheden.

Web-forms

Dit is een applicatie model die kan worden gebruikt om ASP.NET webapplicaties te maken.

MoSCoW

De term MoSCoW betekent zelf niks, maar de letters die als hoofdletters zijn weergegeven staan voor: M = must have, S = should have, C = could have en W = won't have. De vier verschillende categorieën worden gebruikt om een verdeling weer te geven waarvan must have het belangrijkste is en won't have het minste.

OTAP

Ontwikkel, test, acceptatie en productie omgeving. Deze omgevingen worden gebruikt om tijdens het ontwikkelen van een applicatie het te testen en uit te rollen. Het is belangrijk dat alle omgevingen zo gelijk mogelijk zijn qua architectuur.

RUP

Ration Unified Process. RUP is een ontwikkelmethodiek voor softwareontwikkeling waarbij naast de duidelijke fasering het iteratief & incrementeel werken één van de kenmerken zijn.

TRA

Testrisicoanalyse. De testrisicoanalyse wordt gebruikt om binnen een testtraject de nadruk op een bepaald deel van de applicatie te leggen met betrekking tot testen.

UML

Unified Modeling Language. UML is een modelleertaal om ontwerpen voor informatiesystemen te kunnen maken.

SPSS

Statistical Package for the Social Sciences. SPSS is een statistisch programma waarmee data kan worden geanalyseerd.

Klassendiagram

Beschrijft de objecten binnen een programma en de relaties tussen die objecten.

Sequentie diagram

Geeft de volgorde van een scenario in een programma.

ISO 9126

Dit is een internationale standaard voor het vaststellen van de kwaliteit van een stuk software.

Framework

Dit is een geheel van softwareonderdelen die worden gebruikt bij het programmeren van een applicatie. In dit geval het .NET framework.

SVN

Is een software versie of revisie beheer systeem, die wordt gebruikt om versies of revisies bij te houden van bestanden.

Bronnenlijst

Boeken

31-01-12

Enquête research, Ontwikkelen van vragenlijsten en steekproeven (2007)

A.M. Oudemans

K.A.R. Markus

31-01-12

Enquêteren, Het opstellen en gebruiken van vragenlijsten (2004)

E.P.W.A Jansen

Th.H. Joosten

D.R. Kemper

01-03-12

Handboek requirements (2010)

N. de Swart

29-03-12

Software Systems architecture (2008)

Nick Rozanski

Eoin Woods

Internetbronnen

Plan van aanpak:

<http://webtechniek.nl/Informatie.aspx>

Algemene informatie over webtechniek, 06-02-12

<http://www.economische-begrippen.nl/index.php?sectie=afkortingen&sub=m>

Afkorting MBK, 06-02-12

http://www.ru.nl/mercator/trainingen_workshops/begrippen/vm/begrippenlijst_m/

Definitie MBK, 06-02-12

<http://www.rupopmaat.nl/>

RUP, 20-02-12

<http://www.tmap.net/tmap>

TMap next, 01-03-12

Vooronderzoek:

Zoekwoord: "Enquete tool" 20-01-12

<http://www.checkmarket.com/>

<http://www.emailenquete.nl/>

<http://www.2ask.net/>

<http://www.thesistools.com/>

<http://www.enquetemaken.be/>

<http://www.netq-enquete.nl/>

Zoekwoord: "Enquete systeem" 20-01-12

<http://enquete.pmtotz.nl/index.php>

<http://www.kerninformatie.nl/gastenenquete/algemeen/inleiding>

Zoekwoord: "Survey tool" 25-01-12

<http://www.keysurvey.co.uk/>

<http://www.surveytool.com/>
<http://nl.surveymonkey.com/>
<http://www.questionpro.com/>
<http://www.surveygizmo.com/>

Ontwerp:

<http://msdn.microsoft.com/en-us/library/ms227427.aspx>

Resources files, betrekking op talen, 25-03-12

<http://spss.codeplex.com/>

SPSS library, 25-03-12

<http://googlecharts.codeplex.com/>

Google chart, 25-03-12

<http://stackoverflow.com/questions/335061/add-dynamic-charts-using-asp-net-chart-control-c-sharp>

ASP.NET chart control, 25-03-12

<https://www.ibm.com/developerworks/mydeveloperworks/groups/service/html/communityview?communityUuid=ab16c38e-2f7b-4912-a47e-85682d124d32>

IBM SPSS, 25-03-12

<http://www.ibm.com/developerworks/rational/library/content/RationalEdge/sep04/bell/>

Klassendiagram, 26-03-12

<http://www.bced.gov.bc.ca/imb/downloads/classdiagramstandards.pdf>

Domein klassendiagram, 26-03-12

<http://www.microsoft.com/download/en/details.aspx?id=5124>

Powerpoint library, 28-03-12

<http://www.ibm.com/developerworks/rational/library/3101.html>

Sequence diagrammen, 28-03-12

http://www.vincehuston.org/dp/all_uml.html

Design patterns, 29-03-12

[http://msdn.microsoft.com/en-us/library/wss56bz7\(v=vs.80\).aspx](http://msdn.microsoft.com/en-us/library/wss56bz7(v=vs.80).aspx)

Excel library, 29-03-12

http://www.karonaconsulting.com/downloads/UseCases_IncludesAndExtends.pdf

Use cases, 13-04-12

http://www.sparxsystems.com.au/resources/uml2_tutorial/uml2_activitydiagram.html

Activiteiten diagram, 19-04-12

Architectuur

Software Systems Architecture by Nick Rozanski and Eoin Woods

<http://msdn.microsoft.com/en-us/library/ff648105.aspx>

Three-tiered applications, 29-03-12

<http://www.asp.net/web-forms/what-is-web-forms>

Webforms, 07-03-2012

[http://msdn.microsoft.com/en-us/library/dd381412\(VS.98\).aspx](http://msdn.microsoft.com/en-us/library/dd381412(VS.98).aspx)

MVC, 07-03-2012

<http://www.asp.net/web-pages/tutorials/basics/1-getting-started-with-webmatrix-and-asp-net-web-pages>

Web pages, 07-03-2012

<http://weblogs.asp.net/mikaelsoderstrom/archive/2010/07/06/introduction-to-asp-net-web-pages.aspx>

Web pages, 07-03-2012

<http://msdn.microsoft.com/en-us/library/ms973842.aspx>

Framework .NET, 07-03-2012

<http://msdn.microsoft.com/en-us/library/ff647546.aspx>

Three tiered, 09-03-2012

<http://msdn.microsoft.com/en-us/library/ff650258.aspx>

Layered application, 09-03-2012

Requirements

<http://www.genevach.iiba.org/download/BABOK20overview.pdf>

Requirements analyse [The Guide to the Business analysis of knowledge2009], 29-02-2012

<http://www.smartest.nl/verdieping/kwaliteitsmodellen/iso9126>

ISO-norm 9126, 01-03-2012

<http://www.coleyconsulting.co.uk/moscow.htm>

MoSCoW, 07-03-2012

<http://www.ieee.org/index.html>

IEEE Std 830-1998, 04-03-2012

<http://www.microtoolsinc.com/Howsrs.php>

IEEE Std 830-1998, 04-03-2012

<http://www.leidenuniv.nl/fsw/psychologielexicon/index.php3-c=256.htm>

Likert schaal, 10-04-2012

<http://www.leidenuniv.nl/fsw/psychologielexicon/index.php3-c=257.htm>

Thurstone schaal, 10-04-2012

http://nl.wikipedia.org/wiki/Semantisch_differentiaal

OS-good schaal/ semantisch differentiaal, 10-04-2012

Ontwikkelomgeving

<http://www.phparch.com/2009/07/professional-programming-dtap-%E2%80%93-part-1-what-is-dtap/>

OTAP, 02-03-2012

<http://soa-thoughts.blogspot.com/2009/03/dtap-strategy-pricing-and-licensing.html>

OTAP, 02-03-2012

Construction

<http://msdn.microsoft.com/en-us/library/cc295391.aspx>

Bin folder, 03-04-2012

<http://msdn.microsoft.com/en-us/library/ms178684.aspx>

Web.config, 03-04-2012

[http://msdn.microsoft.com/en-us/library/ff400235\(VS.100\).aspx](http://msdn.microsoft.com/en-us/library/ff400235(VS.100).aspx)

Specifiekere informatie over de Web.config, 03-04-2012

<http://msdn.microsoft.com/en-us/library/tydxdyw9.aspx>

Web references, 03-04-2012

<http://msdn.microsoft.com/en-us/library/c76dd5k1.aspx>

Tag prefix, 03-04-2012

<http://msdn.microsoft.com/en-us/library/ms164640.aspx>

Specifiekere informatie over de tag prefix, 03-04-2012

<https://developers.facebook.com/docs/guides/web/>

Facebook API, 04-04-2012

<http://facebook.claritycon.com/help/>

Facebook API, 04-04-2012

<https://developers.facebook.com/docs/share/>

Facebook share button, 04-04-2012

<https://twitter.com/about/resources/buttons#tweet>

Twitter share button, 04-04-2012

<http://developer.linkedin.com/apis>

LinkedIn button, 04-04-2012

<http://www.codeproject.com/Articles/18378/Organization-Chart-Generator>

Flowchart, 05-04-2012

<http://msdn.microsoft.com/en-us/library/ms178472.aspx>

Lifecycle asp page, 06-04-2012

<http://support.microsoft.com/kb/303247>

In-line code & Code-behind, 06-04-2012

<http://msdn.microsoft.com/en-us/library/ee817667.aspx>

Factory pattern, 17-04-2012

<http://www.onestoptesting.com/sdlc-models/v-model.asp>

V-model origineel, 19-04-2012

Woordenlijst

<http://www.computerwoorden.nl/direct--31953--OTAP.htm>

OTAP omgeving definitie, 20-04-2012

<http://www.projectsmart.co.uk/moscow-method.html>

MoSCoW omschrijving, 20-04-2012