



## Ontwikkeling online planningssysteem “BonaPlan”

Auteur:	N.E. Roos
Studentnr:	20007220
Examinatoren:	Dhr. R.A. Mantel Dhr. P.A. Ritman
Plaats, datum:	Zwanenburg 8 oktober 2004
Opdrachtgever:	Bonarius Verwarming
Versie:	4.0

## Referaat

Aan:

Dhr. R.A. Mantel	Examinator
Dhr. P.A. Ritman	Examinator
Dhr. M.C.J. Dee	Bedrijfsmentor
Dhr. G.C. Molenkamp	Bedrijfsmentor

Afstudeerbedrijf:

Bonarius Verwarming  
Zwanenburgerdijk 291  
1161 NL Zwanenburg  
Telefoon: (020) 407 49 00  
Fax: (020) 407 49 99

Auteur:

Nicolai Roos  
20007220

Studie:

Afdeling I&I: Informatica en Informatiekunde  
Studievariant IVIT-4: Informatievoorziening & Informatietechnologie  
Afstudeerrichting: Informatievoorziening

Datum:

8 oktober 2004

Trefwoorden:

- Systeemontwikkeling;
- IAD ontwikkelmethode;
- Workshops;
- Online Planningssysteem.

## Voorwoord

Dit afstudeerrapport is geschreven in het kader van het afstuderen aan de Haagse Hogeschool, opleiding Informatica en Informatiekunde. Ik heb mijn afstudeeropdracht uitgevoerd bij het bedrijf Bonarius Verwarming. De afstudeeropdracht betrof het ontwikkelen van een online planningssysteem (BonaPlan).

Het primaire lezerspubliek van dit afstudeerverslag wordt gevormd door mijn examinatoren en de toegewezen gecommiteerde, die mijn afstudeerstage onder andere aan de hand van dit document gaan beoordelen. Daarnaast mag iedereen die interesse heeft in de ontwikkeling van BonaPlan dit document inzien, enige voorkennis van informatica is hierbij wel handig.

Voor het uitvoeren van deze opdracht gaat mijn dank in het bijzonder uit naar de opdrachtgevers en bedrijfsmentors dhr. Dee & dhr. Molenkamp voor het verlenen van de opdracht en voor de tijd en energie die zij in mij heeft gestoken. Daarnaast wil ik ook de collega's van de afdeling administratie benoemen voor de goede sfeer en collegialiteit tijdens het afstudeerproject. Tenslotte wil ik Ron Mantel en Peter Ritman bedanken voor hun tijd en begeleiding tijdens de afstudeerstage.

Nicolaï Roos  
Zwanenburg, 8 oktober 2004

## Inhoudsopgave

<b>Inleiding</b>	<b>6</b>
<b>1 Bonarius Verwarming</b>	<b>7</b>
<b>2 Omschrijving opdracht</b>	<b>10</b>
2.1 Aanleiding	10
2.2 Probleemstelling	10
2.3 Doelstelling van de opdracht	10
2.4 Uitgangssituatie	10
<b>3 Aanpak</b>	<b>11</b>
3.1 Gebruikte middelen	11
3.1.1 Ontwikkelomgeving	11
3.1.2 Ontwikkelmethodiek	11
3.1.3 Ontwikkeltechnieken	12
3.1.4 Overige technieken	13
3.2 De Planning	14
<b>4 Definitiestudie BonaPlan</b>	<b>17</b>
4.1 Plan van aanpak definitiestudie BonaPlan	17
4.2 Analyse	17
4.3 Ontwikkelscenario BonaPlan	18
4.4 Systeemeisen BonaPlan	19
4.5 Systeemconcept BonaPlan	20
4.6 Technische structuur voor BonaPlan	22
4.7 Organisatorische inrichting	24
4.8 Pilotplan	25
4.9 Verificatie en validatie	25
<b>5 Pilotontwikkeling</b>	<b>26</b>
5.1 Pilot 1: De database	26
5.1.1 Functionele structuur pilot database	26
5.1.2 Pilotontwikkelplan database	28
5.1.3 Verificatie en validatie pilot Database	30
5.2 Pilot 2: Planningssysteem	31
5.2.1 Functionele structuur pilot planningssysteem	32
5.2.2 Organisatorische inrichting pilot planningssysteem	40
5.2.3 Pilotontwikkelplan pilot planningssysteem	41
5.2.4 Verificatie en validatie pilot planningssysteem	41
5.3 Pilot 3: Gegevensbeheer	43
5.3.1 Functionele structuur pilot gegevensbeheer	43
5.4 Pilot 4: Overzichten	49
5.4.1 Functionele structuur pilot gegevensbeheer	50
<b>6 Invoering</b>	<b>57</b>
6.1 Testen van BonaPlan	57
6.2 Invoeren	57
<b>7 Evaluatie</b>	<b>58</b>
7.1 Proces	58
7.2 Product	59
7.3 Geboekte leerwinst	60
7.4 Conclusie	60
<b>Figuren lijst</b>	<b>61</b>
<b>Verklarende woordenlijst</b>	<b>62</b>
<b>Literatuurlijst</b>	<b>63</b>

<b>INTERNE BIJLAGEN .....</b>	<b>64</b>
<b>Bijlage 1 : Opdrachtschrijving .....</b>	<b>64</b>
<b>Bijlage 2 : Plan van aanpak .....</b>	<b>68</b>
<b>Bijlage 3: Workshop 1 .....</b>	<b>74</b>
<b>Bijlage 4: Workshop 2 .....</b>	<b>80</b>
<b>Bijlage 5: Interview .....</b>	<b>85</b>
<b>Bijlage 6: Wekelijkse contactmomenten .....</b>	<b>87</b>
 <b>EXTERNE BIJLAGEN .....</b>	 <b>89</b>
<b>1. Definitiestudie .....</b>	<b>89</b>
<b>2. Rapport Pilotontwikkelplan Database .....</b>	<b>89</b>
<b>3. Rapport Pilotontwikkelplan Planningssysteem .....</b>	<b>89</b>
<b>4. Rapport Pilotontwikkelplan Gegevensbeheer .....</b>	<b>89</b>
<b>5. Rapport Pilotontwikkelplan Overzichten .....</b>	<b>89</b>
<b>6. Operationeel maken BonaPlan .....</b>	<b>89</b>
<b>7. Testplan .....</b>	<b>89</b>

## Inleiding

Dit document is het afstudeerverslag dat betrekking heeft op het afstudeertraject van Nicolai Roos, een vierdejaars student aan de afdeling Informatica en Informatiekunde van de Haagse Hogeschool. Het afstudeerverslag zal de lezer een beeld verschaffen van de door mij uitgevoerde activiteiten, de gemaakte keuzes, de gehanteerde methodes en werkwijzen, kortom: welke activiteiten zijn uitgevoerd, hoe zijn deze activiteiten uitgevoerd, waarom zijn ze zo uitgevoerd en wat was het resultaat van deze activiteiten, zodanig dat het HBO-niveau kan worden vastgesteld en getoetst. Daarnaast wordt in dit document ook aandacht besteed aan de geboekte leerwinst en zal verantwoording worden afgelegd voor het eventueel niet halen van gestelde doelen.

Van 17 mei 2004 t/m 8 oktober 2004 heb ik gewerkt aan mijn afstudeeropdracht bij Bonarius Verwarming, een installatie bedrijf gevestigd te Zwanenburg. Deze opdracht luidde in het kort als volgt: "Ontwikkeling on-line planningssysteem die de bedrijfsprocessen van Bonarius Verwarming ondersteunt".

Dit document is gestructureerd naar de gevolgde stappen volgens de methodiek IAD (Iterative Application Design). Er is voor deze structurering gekozen omdat het doorlopen proces zo op een chronologische en gestructureerde manier kan worden beschreven.

In hoofdstuk één en twee zullen de organisatie en de opdracht in kaart worden gebracht. In hoofdstuk drie kunt u meer lezen over de aanpak van dit project. De gebruikte methoden, technieken en de planning zijn enkele onderwerpen die in dit hoofdstuk behandeld zullen worden. In het volgende hoofdstuk wordt de definitiestudie nader toegelicht, de definitiestudie is de eerste fase van de ontwikkelmethodiek IAD. Het vijfde hoofdstuk zal de manier van, de keuzes, problemen en oplossingen van de gehele ontwikkeling van BonaPlan toelichten. De verschillende paragrafen zijn de pilots welke tijdens de ontwikkeling gebruikt zijn. Nadat de ontwikkeling van BonaPlan is toegelicht zal de invoering van het systeem besproken worden in hoofdstuk zes. Tenslotte word in het laatste hoofdstuk het product en het proces geëvalueerd.

In dit rapport wordt regelmatig gerefereerd aan delen uit de producten die zijn opgenomen in de externe bijlagen. Daar waar delen uit de producten in dit rapport zijn gekopieerd worden die delen in een kader afgebeeld.

# 1 Bonarius Verwarming

Bonarius is een bedrijf met een lange geschiedenis. Volgens de papieren van de Kamer van Koophandel en Fabrieken voor Haarlem en Omstreken zou het tijdstip van vestiging van de onderneming zelfs teruggaan naar het jaar 1785.

In die tijd bestond centrale verwarming uiteraard nog niet. Het bedrijf Bonarius was oorspronkelijk een koperslagerbedrijf, annex lood- en zinkwerken en was gevestigd in de toenmalige Wijngaardstee, tussen de Jansstraat en de Kruisstraat in Haarlem. Pas veel later, in 1859, verhuisde het bedrijf naar de Bloemendaalseweg 106 – 108 in Bloemendaal. Daar is het bedrijf 100 jaar gevestigd geweest, tot het in 1968 verhuisde naar de Vijverweg 22, eveneens in Bloemendaal. Al twee jaar later, in 1970, moest worden uitgezien naar een groter pand dat strategischer was gelegen ten opzichte van de potentiële opdrachtgevers. Bloemendaal werd verlaten en Bonarius vestigde zich aan de Schipholweg 957 in Boesingheliede.

Om de relaties nog beter van dienst te zijn en het onderhoud van installaties niet afhankelijk te maken van beschikbare ruimte bij de aanleg van nieuwe installaties, werd in 1984 officieel gestart met een zelfstandig servicebedrijf, Bonarius Service B.V.

Bonarius heeft in de afgelopen 50 jaar bijna 100.000 verwarmingsinstallaties aangelegd. Het bedrijf bestaat inmiddels uit 130 medewerkers. Voor bijna 60.000 huishoudens in de Noordelijke Randstad (Amsterdam, Nieuw Vennep, Haarlem, Alkmaar, Hoorn, Zaanstad) verzorgt Bonarius het periodiek en storingsonderhoud van cv-installaties. Bonarius streeft er naar een kwaliteitsorganisatie te zijn, kwaliteit in drie vormen:

1. Kwaliteit als zorg voor de eigen medewerkers;
2. Kwaliteit in de vorm van certificering (BRL 6001 en VCA\*\*);
3. Kwaliteit door klantgericht te werk te gaan.

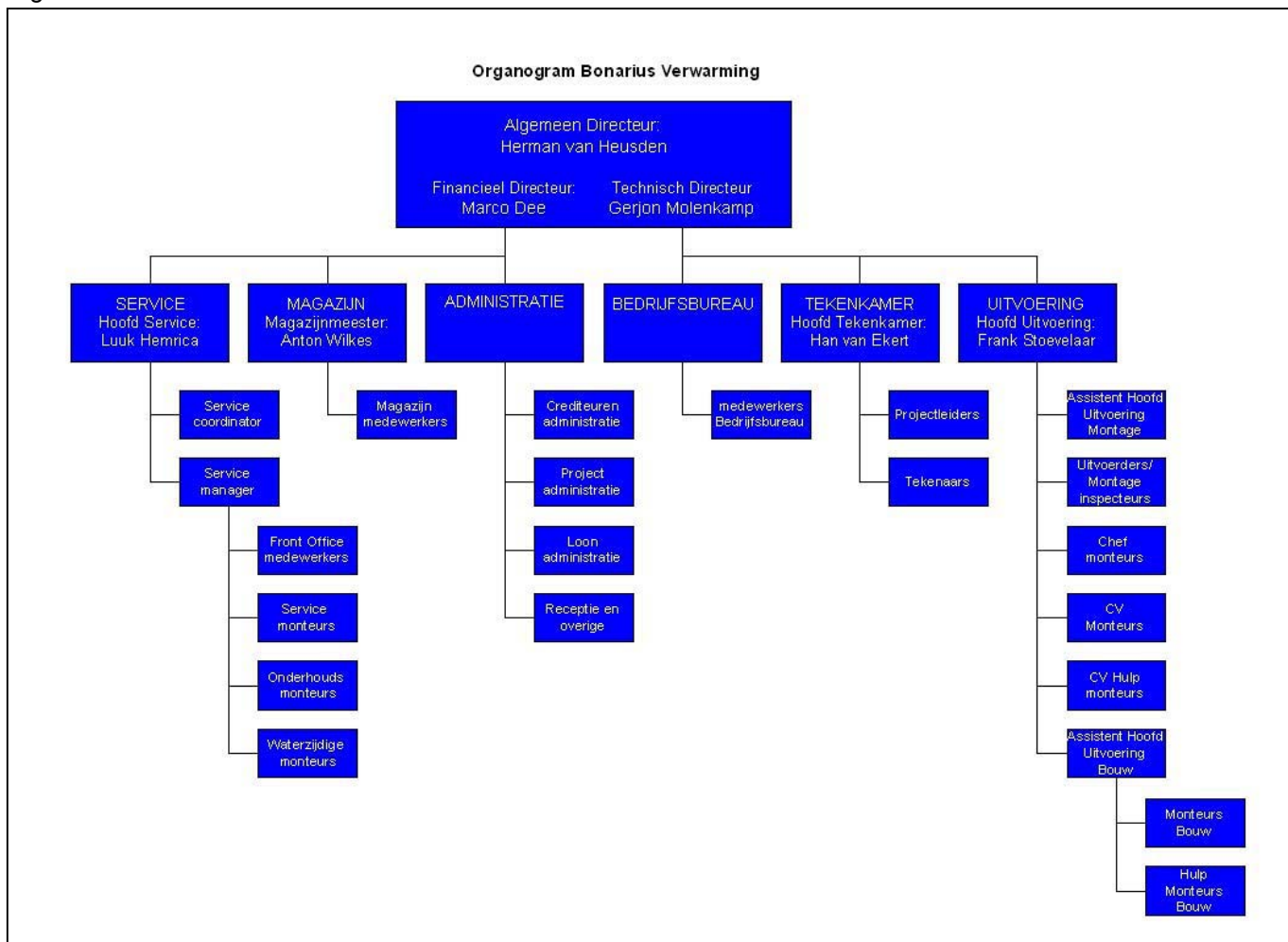
Hoewel Bonarius behoort tot één van de oudste bedrijven van Nederland is met name de afgelopen jaren een periode van sterke groei ingezet. Dit heeft geleid tot een duidelijke omslag in de organisatie, met accenten op een verdere professionalisering en een servicegerichte werkwijze. Tot dit veranderingsproces behoort ook de verhuizing, in maart 2004, naar een nieuwe en moderne locatie in Zwanenburg.

De Bonarius bedrijven hebben zich gespecialiseerd in individuele verwarmingsinstallaties voor de particuliere markt, voornamelijk huurwoningen, maar ook koopwoningen. In de meeste gevallen gaat het om traditionele HR-ketels, maar in toenemende mate eveneens om vernieuwende, energiezuinige systemen voor zonne-energie, mechanische ventilatie en toepassingen van warmtepompen.

Bonarius Verwarming bestaat uit drie onderdelen te weten:

1. Bonarius Techniek;
2. Bonarius Service;
3. Bonarius Bouw.

De structuur van Bonarius Verwarming wordt in het onderstaande figuur als organogram gevisualiseerd.



**Figuur 1.1: Organogram**

### Bonarius Techniek

Bonarius Techniek verzorgt het ontwerp, de voorbereiding en de uitvoering van nieuwbouw en renovatie-projecten. Veelal gaat het hierbij om zogenaamde comfort systemen, waarin het totaal van de cv- en warmwatervoorziening is betrokken. Dergelijke systemen kunnen worden uitgevoerd in een veelheid aan uitvoeringsvarianten (extra warmwater, vloerverwarming, (design) radiatoren, convactorputten, enz). Ook het vervangen van aanwezige, dan wel aanbrengen van nieuwe mechanische ventilatie (eventueel in combinatie met warmteterugwinning) kan onderdeel uitmaken van een dergelijk systeem.

Op het gebied van het seriematig aanbrengen van individuele HR-installaties heeft Bonarius Techniek een krachtige en concurrerende marktpositie opgebouwd. Een sterk punt van Bonarius Techniek is de aanwezige expertise. Dat betekent dat installaties van alle ketelfabrikanten, al of niet gecombineerd met vloerverwarming of radiatoren, kunnen worden berekend en geïnstalleerd. De tekenkamer kan iedere opdrachtgever, architect of aannemer, reeds in een vroegtijdig stadium van een project deskundig adviseren over systeem- en materiaalkeuze, alsmede de situering van de diverse componenten in een woning.



Installatie van centrale verwarming in bestaande woningen vereist veel organisatie, overleg met de bewoners en een goede logistiek in het uitvoeringstraject. Voor een aantal grote woningbouwverenigingen heeft Bonarius Techniek reeds in vele duizenden woningen nieuwe HR-installaties aangebracht, waarbij veelal ook een compleet stelsel van luchttoevoer- en rookgasafvoerkanalen werd geïnstalleerd. Deze systemen, die voldoen aan de aangescherpte voorschriften op het gebied van gas- en ventilatievoorzieningen (NEN en Bouwbesluit), vervangen de doorgaans aanwezige niet meer toepasbare keukengeiser en lokale verwarming. In overleg met opdrachtgevers kunnen ook 'pakketten' in voorraad worden genomen, zodat bij het wisselen van huurders direct tot installatie kan worden overgegaan. Dit voorkomt onnodige leegstand.

### **Bonarius Service**

Het verzorgen van service en onderhoud is de speerpunt activiteit binnen de Bonarius bedrijven. Bonarius Service zorgt ervoor dat 60.000 individuele cv-installaties in goede staat worden gehouden door preventief en correctief onderhoud uit te voeren. Tot de opdrachtgevers behoren woningcorporaties, beleggers en een beperkt aantal particulieren.

Storingen worden gemeld bij de medewerkers van de Front Office. Deze sturen de storingsmonteurs aan om nog diezelfde dag de storing op te lossen. Tevens worden van hieruit de onderhoudsmonteurs aangestuurd, die volgens een vaste planning preventief onderhoud plegen. De bewoners worden daar van te voren van op de hoogte gesteld.

Doelstelling van Bonarius Service is: het minimaliseren van het aantal bezoeken aan een woning en het optimaliseren van de standtijd van de in onderhoud zijnde cv-installaties. Om de bezoekgraad aan de woningen daadwerkelijk te verlagen, heeft Bonarius Service haar werkwijze verder geoptimaliseerd. Meten is weten! Aan de hand van uitgebreide rapportering is het mogelijk de bezoek- en onderhoudsfrequentie omlaag te brengen. Aan de hand van de door Bonarius Service opgestelde rapportages wordt overleg gevoerd met de fabrikanten over verdere verbeteringen aan de cv-ketels. Soms staat men zelfs aan het begin van zo'n ontwikkeling, zoals bij de populaire HR-ketel van Intergas, die thans op grote schaal wordt toegepast. Ook bij leveranciers van andere materialen wordt gewezen op onderdelen met een verhoogde kans op storingen. Een andere ontwikkeling waar Bonarius aan meewerkt is het ontwikkelen van systemen om op afstand cv-ketels te kunnen monitoren en besturen. Dergelijke ontwikkelingen maken het mogelijk de bezoeken aan de woning anders te organiseren en de onderhoudsfrequentie te verlagen.

### **Bonarius Bouw**

Bonarius Bouw voert bouwkundige aanpassingen uit en zorgt voor de aanleg van luchttoevoer- en rookgasafvoerkanalen, alsmede uitvoering van mutatiewoningen. Bonarius Bouw werkt uitsluitend in opdracht van Bonarius Service en Bonarius Techniek.

### **Plaats binnen Bonarius**

Binnen Bonarius Verwarming is mijn werkplek gevestigd op de afdeling administratie. Binnen Bonarius word ik begeleid door twee personen.

M.C.J. Dee	<a href="mailto:Dee@bonarius.com">Dee@bonarius.com</a>	tel: 020-4074904
G.C. Molenkamp	<a href="mailto:Molenkamp@bonarius.com">Molenkamp@bonarius.com</a>	tel: 020-4074923

Tijdens kantooruren zijn dhr. Dee en dhr. Molenkamp te vinden in het complex van Bonarius Verwarming op de Zwanenburgerdijk 291 te Zwanenburg.

BonaPlan wordt ontwikkeld voor het onderdeel Bonarius Techniek.

## **2 Omschrijving opdracht**

Dit hoofdstuk betreft de opdracht. In dit hoofdstuk zal een korte omschrijving van de opdracht worden gegeven. Tevens zullen de aanleiding, probleemstelling en doelstelling van de opdracht worden toegelicht. Dit hoofdstuk is in dit document opgenomen om de lezer inzicht te geven in de context waarbinnen dit project zich heeft afgespeeld. Dit hoofdstuk is gebaseerd op de opdrachtomschrijving welke is terug te vinden in de interne bijlage: Opdrachtomschrijving

### **2.1 Aanleiding**

Bonarius behoort tot de oudste bedrijven van Nederland (vestigingsjaar 1785). De afgelopen jaren heeft Bonarius een periode van sterke groei doorgemaakt. Dit heeft geleid tot een verandering in de organisatie, met accenten op een verdere professionalisering en een servicegerichte werkwijze.

### **2.2 Probleemstelling**

De grote groei van Bonarius Verwarming heeft er voor gezorgd dat er behoefte is aan een andere manier van plannen. Waarbij in het kader van klantgericht werken deze via internet voor opdrachtgevers (klanten) beschikbaar moet zijn voor status navragen. Op dit moment worden alle plannings gemaakt in Excel. Alle overzichten worden handmatig gemaakt en weer ingevoerd in Excel. Dit is niet alleen een tijdrovende klus maar ook erg omslachtig.

### **2.3 Doelstelling van de opdracht**

Het doel van de afstudeeropdracht is het ontwerpen, ontwikkelen en implementeren van een online planningssysteem voor Bonarius Verwarming (BonaPlan). Het planningssysteem moet ervoor zorgen dat mensen en middelen efficiënt en minder omslachtig, kunnen worden ingepland voor de verschillende projecten. Tevens dient het planningssysteem meer inzicht te geven aan opdrachtgevers (klanten) zodat zij ten alle tijden de status van een project kunnen bekijken. Dit is onderdeel van de verdere professionalisering van Bonarius Verwarming. Ook wil men middels deze functie van BonaPlan de service van Bonarius Verwarming proberen te ontlasten (minder status navragen door opdrachtgevers).

### **2.4 Uitgangssituatie**

Het plannen van de verschillende projecten wordt op dit moment gedaan door de invulling van verschillende zelf ontwikkelde Excel sheets. Zo zijn er de volgende Excel sheets:

- Planlijst verketelingen;
- Planlijst;
- Mutatieplanning;
- Planning tekenkamer.

Deze worden vervolgens weer handmatig gevisualiseerd in het zogenoemde balkenschema. Omdat dit handmatig gebeurt is dit een tijdrovende klus welke bovendien nog inconsistentie van gegevens met zicht mee kan brengen aangezien dit handmatig gedaan wordt (een mens kan fouten maken). Meer informatie over de huidige manier van plannen vind u in paragraaf 4.2.

### 3 Aanpak

De aanpak van dit project is beschreven doormiddel van een plan van aanpak. In dit plan van aanpak staat beschreven welke activiteiten uitgevoerd moesten worden, hoeveel tijd er voor deze activiteiten beschikbaar was en welke middelen er nodig waren om de activiteiten uit te kunnen voeren. Het plan van aanpak is opgesteld om voor mijzelf inzicht te verschaffen in het uit te voeren project en om een leidraad te creëren voor het uitvoeren van het project. Het inzicht is verschaft door na te denken over de te gebruiken middelen. De leidraad is gecreëerd door het opstellen van een planning. Zowel de gebruikte middelen als de planning zullen in dit hoofdstuk behandeld worden.

#### 3.1 Gebruikte middelen

Tijdens het project zijn er van een aantal middelen gebruik gemaakt om de benodigde activiteiten uit te kunnen voeren. De keuze om dit op te nemen in het afstudeerverslag is om de lezer inzicht te verschaffen in de context waarbinnen dit project heeft plaatsgevonden.

De middelen die in dit hoofdstuk behandeld worden staan hieronder opgesomd:

- Ontwikkelomgevingen: PHP, HTML, MySQL, PHPMyAdmin;
- Complete pc configuratie;
- Webserver;
- Ontwikkelmethodiek: IAD;
- Ontwikkeltechniek: UML;
- Overige technieken: interviews, workshops en timeboxing.

##### 3.1.1 Ontwikkelomgeving

Een aantal middelen hebben geen verklaring nodig zoals een complete PC configuratie. Andere gebruikte middelen verdienen wel een verklaring. Voor het daadwerkelijk programmeren van het systeem is gebruik gemaakt van PHP en HTML. PHP is een taal waarmee websites ontwikkeld kunnen worden en waarmee tussen een website en een database kan worden gecommuniceerd. HTML is ook een taal waarmee websites kunnen worden gebouwd, alleen een stuk meer gericht op de lay-out van een website en niet zo zeer de communicatie met een database. De keuze voor de scripttaal PHP word nader toegelicht in hoofdstuk vier van dit verslag. HTML is de taal geweest voor het ontwikkelen van de lay-out, met behulp van deze taal valt namelijk op een vrij simpele manier een lay-out te ontwikkelen tevens had ik hier al enige ervaring mee.

MySQL is een database die doormiddel van SQL queries gekoppeld kan worden aan een website. De keuze voor MySQL word wederom in hoofdstuk vier van dit verslag verantwoord.

PHPMyAdmin is een front-end voor MySQL waarmee een database beheerd kan worden. Aangezien PHPMyAdmin gratis is en dat de ontwikkelaar eerdere ervaring hiermee heeft was de keuze voor deze database beheer-tool snel gemaakt.

Het was noodzakelijk dat de webserver (ontwikkelomgeving) was noodzakelijk in het begin van het project werd opgesteld omdat anders het te ontwikkelen systeem niet ontwikkeld kon worden. Dit systeem is namelijk afhankelijk van de ontwikkelomgeving. Daarnaast was de ontwikkelomgeving ook de uiteindelijke omgeving waar het complete systeem geïmplementeerd zou worden.

##### 3.1.2 Ontwikkelmethodiek

Voor het uitvoeren van het project is er gebruik gemaakt van een ontwikkelstrategie. Als ontwikkelstrategie is er gekozen voor IAD (Iterative Application Management). IAD is een methode die uit drie fases bestaat. Deze fases zijn:

- Definitiestudie;
- Pilotontwikkeling;
- Invoering.

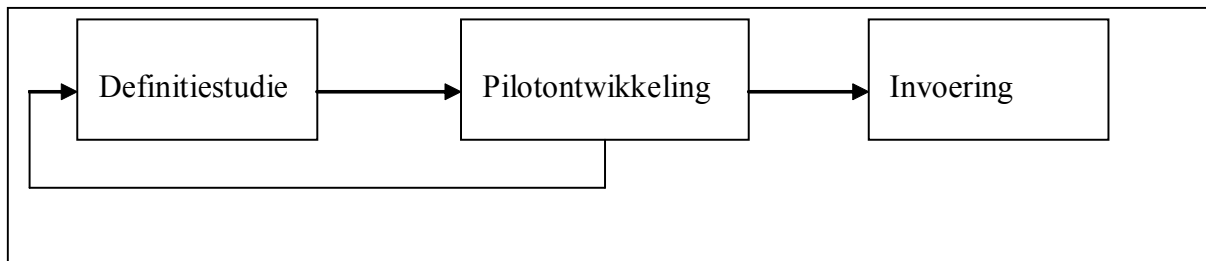
In de fase definitiestudie worden de doelen van het systeem geanalyseerd, evenals de beperkingen. In deze fase wordt de rest van het verloop van het project bepaald. Tijdens de pilotontwikkeling worden de verschillende onderdelen, oftewel de pilots van het systeem ontworpen en gebouwd. In de

laatste fase, de invoering, worden de ontwikkelde pilots daadwerkelijk operationeel gemaakt en wordt het systeem overgedragen aan de opdrachtgever.

Er is voor dit project om verschillende redenen gekozen voor de ontwikkelmethodiek IAD. Deze redenen worden hieronder opgesomd:

- Het gebruik van IAD stelt de ontwikkelaar in staat de gebruikers en opdrachtgever nadrukkelijker bij het ontwikkelproces te betrekken. Dit heeft als groot voordeel dat er uiteindelijk een systeem zal worden opgeleverd dat zo goed als mogelijk aansluit op de wensen van de opdrachtgever;
- IAD leent zich voor projecten met onduidelijke systeem eisen: De systeemeisen voor het project moesten dan ook groeien naarmate het project vorderde;
- Het feit dat ik al behoorlijk wat ervaring met de toepassing van IAD had door reeds eerder uitgevoerde projecten. Het toepassen van IAD zou hierdoor niet alleen een kwalitatief beter resultaat opleveren, maar tevens behoorlijk wat tijd besparen omdat er dan geen nieuwe ontwikkelstrategie eigen gemaakt hoefde te worden.

IAD kan volgens verschillende scenario's worden toegepast. Het scenario bepaald hoe de verschillende fases worden doorlopen. Voor de ontwikkeling van dit project is gekozen voor het Big-Bang-Invoeren. Deze strategie wordt zo genoemd omdat het project middels een aantal iteraties wordt ontwikkeld en daarna in een klap (Big-Bang) wordt ingevoerd. In het onderstaande figuur is een schematische weergave te zien van deze iteratiestrategie.



Figuur 3.1: Iteratie strategie Big-Bang-invoeren

Big-Bang-invoeren is een ontwikkelstrategie waarin de eerste twee fases (definitiestudie en pilotontwikkeling) iteratief worden uitgevoerd, maar de laatste (invoering) niet. Er worden dus weliswaar diverse pilots na elkaar ontwikkeld, maar geen van die pilots wordt ingevoerd in de organisatie; dat gebeurt pas met de laatste, definitieve pilot. De fase invoering wordt dus slechts eenmalig uitgevoerd. De rede dat er juist voor dit scenario was gekozen, was het feit dat snelle ingevoerde resultaten niet nodig waren in dit project. Het te ontwikkelen systeem zou pas zinnig gebruikt kunnen worden, wanneer alle delen waren ingevoerd. Big-Bang-invoeren brengt helaas ook een aantal nadelen met zich mee zo zou door het gebruik van deze strategie te weinig feedback uit de praktijk kunnen worden verkregen aangezien het gehele werkende product pas aan het eind van het project wordt ingevoerd. Dit nadeel heb ik zoveel mogelijk beperkt door tussentijds verschillende workshops en demonstraties te geven over werkende delen van het systeem. Met deze workshops / demonstraties kreeg ik niet alleen feedback voor een nieuwe iteratie slag binnen het project maar ook feedback welke overeenkomt met feedback uit de praktijk, immers tijdens workshop werkt de eindgebruiker met het systeem.

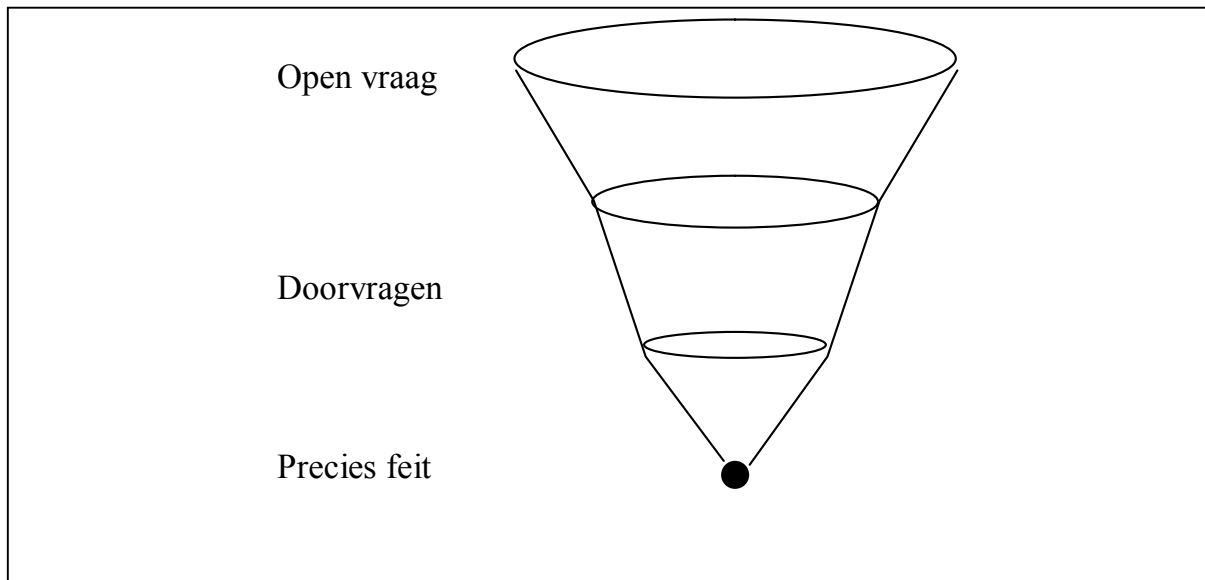
### 3.1.3 Ontwikkeltechnieken

Als modelleringstechniek is er gekozen voor UML (Unified Modeling Language). UML bestaat uit een aantal modelleertechnieken als use-cases, activity diagrammen, sequence diagrammen en klassediagrammen. Deze modelleertechnieken zijn met name gebruikt tijdens de definitiestudie bij het opstellen van het systeemconcept. Er is er voor gekozen om een klassediagram te maken voor de database. Daarnaast zijn er voor het systeem use-case diagrammen en use-cases gemaakt. De belangrijkste redenen dat ik voor de modelleringstechniek UML gekozen heb zijn:

1. UML is een techniek waarmee goed valt te communiceren naar de opdrachtgever. Voor iemand zonder kennis van deze techniek is de techniek relatief simpel te begrijpen.
2. Deze techniek wordt door de Haagse Hogeschool aan zijn studenten aangeleerd, deze techniek beheers ik dan ook. De keuze voor UML was daardoor snel gemaakt.

### 3.1.4 Overige technieken

Om de opdracht duidelijk te krijgen, leek het interviewen van de opdrachtgever en gebruikers een geschikte techniek. Gedurende de opleiding is er een module geweest die interviewtechnieken heeft behandeld. Aan de hand van deze module is de interviewstrategie bepaald. Middels de 'fuikprocedure' zal er getracht antwoorden te krijgen op de gestelde vragen. In de 'fuikprocedure' begint de interviewer met een aantal algemene open vragen waarop vervolgens wordt doorgevraagd totdat de opdrachtgever gedetailleerd genoeg heeft geantwoord. In figuur 3.2 is een schematische weergave te zien van de 'fuikprocedure'.



Figuur 3.2: Fuikprocedure

De fuikprocedure is dus toegepast binnen de verschillende interviews, gesprekken en wekelijkse contactmomenten welke tijdens dit project hebben plaatsgevonden. De fuikprocedure is niet iets geweest wat vast stond, deze techniek werd gebruikt om een echt conversatie te creëren om zo tot hetgeen te komen wat de opdrachtgever en/of gebruiker daadwerkelijk bedoelde.

Om het te ontwikkelen systeem beter te kunnen laten aansluiten op de wensen en eisen van de opdrachtgever en gebruikers is er gebruik gemaakt van workshops. Een workshop is een interactieve sessie waarin samen met de opdrachtgever het systeem gespecificeerd, ontworpen en gevalideerd wordt. Deze workshops zijn uitgevoerd aan de hand van prototyping. Voor het gebruik van workshops is gekozen omdat ervaringen met workshops, tijdens eerdere projecten met de ontwikkelmethodiek IAD, uitermate goed zijn bevallen. De intensieve samenwerking met de opdrachtgever levert namelijk doorgaans goede resultaten op. Het gebruik van prototyping was ideaal tijdens deze workshops doordat de visuele ondersteuning veel feedback van de opdrachtgever en gebruikers opleverde en daardoor het systeem beter aan de wensen en eisen van de opdrachtgever en gebruikers kon worden aangepast.

Daarnaast zal er nog gebruik worden gemaakt van time-boxing voor het plannen van de pilots. Het voordeel van het gebruik van time-boxing is dat de verschillende pilotdelen goed in de beschikbare tijd kunnen worden ingedeeld en dat middels de timebox de voortgang van het project op ieder moment gecontroleerd kan worden.

### 3.2 De Planning

Voordat er daadwerkelijk aan het project begonnen kon worden, diende er eerst een planning te worden opgesteld. Om een globaal overzicht te creëren van de werkzaamheden die voor dit project moesten worden verricht, is er eerst een globale planning opgesteld. De planning is opgesteld aan de hand van de opdrachtschrijving en de gesprekken / interviews die met de opdrachtgever gehouden waren. Hieronder vind u de laatste versie van de globale planning.

Activiteit	Dagen
<b>Globaal Plan van aanpak</b>	<b>3</b>
<b>Oriëntatie bedrijf + opdracht</b>	<b>2</b>
<b>Definitiestudie</b>	<b>21</b>
Specificeren plan van aanpak	1
Analyse	1
Opstellen ontwikkelscenario	2
Bepalen systeemeisen	3
Opstellen systeemconcept	5
Opstellen technische structuur	5
• Onderzoek webserver	1
• Onderzoek PHP of ASP	1
• Onderzoek Database architectuur	1
• Bouw webserver	1
Opstellen pilotplan	3
Verificatie en validatie	1
<b>Pilotontwikkeling</b>	<b>44</b>
Specificeren plan van aanpak	2
Bepalen globaal functionele structuur	15
Bepalen globaal organisatorische inrichting	2
Opstellen pilotontwikkelplan	4
Bouwen bouweenheden	20
Verificatie en validatie	1
<b>Testen</b>	<b>5</b>
<b>TOTAAL</b>	<b>75</b>
<b>Eindverslag</b>	<b>15</b>

Figuur 3.3: Planning

De gekozen ontwikkelmethodiek IAD is van invloed op de planning. De planning bestaat namelijk uit de fasen van IAD. Wat direct opvalt is dat de fase invoering ontbreekt. Dit heeft te maken met het feit dat workshop één (wordt nader toegelicht in het hoofdstuk 5.2 Pilot twee: Planningssysteem) grote veranderingen met zich meebracht en dat daardoor de invoering na de afstudeerperiode zal plaatsvinden. Wel zal er nog worden getest.

Na het opstellen van de globale planning ben ik verder gegaan met het opstellen van de detailplanning. Vanwege de omvang van dit project hechtte ik veel waarde aan een concrete structuur in de werkzaamheden. Om deze reden koos ik dan ook om per onderdeel te gaan plannen, zo bestond de complete detailplanning dus uit verschillende deelplanningen. Deze deelplanningen zijn in de verschillende rapporten (definitiestudie en pilotontwikkel rapporten, welke in de externe bijlagen te vinden zijn) verwerkt.

Om toch in het begin van het project een duidelijk overzicht van het project te creëren, is er tussenoplossing bedacht. Op basis van de globale planning zijn er per iteratie en per onderdeel, de uit te voeren activiteiten bepaald. Tevens is per onderdeel het aantal dagen bepaald. In figuur 3.4 is een fragment uit deze planning te zien



Eerste iteratie	
• Definitiestudie ( 15 dagen)	
o Plan van aanpak	
o Analyse	
o Ontwikkelscenario	
o Systeemeisen	
o Systeemconcept	
o Technische structuur	
▪ Onderzoek naar webserver	
▪ Onderzoek PHP of ASP	
▪ Onderzoek database architectuur	
▪ Bouw webserver	
o Organisatorische inrichting	
o Pilotplan	
o Verificatie en validatie	

Figuur 3.4: Tussenoplossing planning

Op basis van deze planning kon bij de start van elk onderdeel, een gedetailleerde planning voor dat onderdeel worden opgesteld (de zogenaamde deelplanningen). In figuur 3.5 is de detailplanning van de definitiestudie te zien van de eerste iteratie. Zoals in het figuur te zien is, worden per dag de uit te voeren activiteiten beschreven, het aantal uur dat per activiteit beschikbaar is en de betreffende iteratie waarin het onderdeel zich bevindt.

Week 2	Onderdeel	Activiteit	# uur	Iteratie
24-05-2004	Definitiestudie	• Specificeren P.V.A	4	1
		• Analyse huidige situatie	4	
25-05-2004	Definitiestudie	• Analyse huidige situatie	6	1
		• Ontwikkelscenario	2	
26-05-2004	Definitiestudie	• Ontwikkelscenario	8	1
27-05-2004	Definitiestudie	• Systeemeisen	8	1
28-05-2004	Definitiestudie	• Systeemeisen	8	1
Week 3	Onderdeel	Activiteit	# uur	Iteratie
31-05-2004	Definitiestudie	• Systeemconcept: Use-cases	8	1
01-06-2004	Definitiestudie	• Systeemconcept: Use-cases/klasediagram	8	1
02-06-2004	Definitiestudie	• Systeemconcept: klassendiagram / use-case diagram	8	1
03-06-2004	Definitiestudie	• Technische structuur: Onderzoek webserver	8	1
04-06-2004	Definitiestudie	• Technische structuur: Onderzoek PHP /ASP	8	1
Week 4	Onderdeel	Activiteit	# uur	Iteratie
07-06-2004		• Technische structuur: Onderzoek database	8	1
08-06-2004		• Technische structuur: Bouw van database +testen	8	1
09-06-2004		• Pilotplan	8	1

Figuur 3.5: Detailplanning (deelplanning)

Door de keuze voor het Big-Bang invoeren worden de definitiestudie en de verschillende pilots meerdere malen doorlopen terwijl de invoering in een klap wordt gedaan. Het was erg moeilijke om een planning op te stellen, van te voren bepalen hoe lang een iteratie duurt is niet de makkelijkste bezigheid. Als algemene regel is genomen dat de eerste iteratie langer duurt dan de daaropvolgende iteraties. Dit is gedaan omdat in de eerste iteratie de definitiestudie en de gehele pilots opgezet moesten worden en in de daarop volgende iteraties de definitiestudie en de verschillende pilots verder verfijnt moeten worden. In deze iteraties kan er dan ook veel hergebruikt worden wat de tijd voor een iteratie flink naar beneden brengt.

In de beginfase van het project is besloten dat het systeem doormiddel van twee iteratieslagen ontwikkeld zou gaan worden. Dit had te maken met het feit dat het gehele project 75 dagen in beslag mocht nemen. In de negende week van het project heb ik een workshop / demonstratie. Deze workshop is uitgevoerd om de opdrachtgevers meer inzicht te geven in de ontwikkelde onderdelen van het systeem, en om eventueel vroegtijdig het project te kunnen bijsturen. De workshop / demonstratie bestond uit het uitvoeren van een aantal opdrachten plus een demonstratie van het planningsgedeelte. Hieronder ziet u een fragment uit deze workshop. De gehele workshop is te vinden als interne bijlage in dit document.

**Opdracht 1.2**

Alle gegevensbeheer onderdelen werken op dezelfde manier, daarom zullen we niet alle schermen doorlopen.

6. U bevindt zich in het scherm gegevensbeheer druk op de link PROJECTLEIDER
7. U ziet een overzicht van de aanwezige projectleider. Druk op het ID om de detail gegevens te zien.
8. Druk op de link terug
9. Druk op de link nieuwe projectleider en voeg een nieuwe projectleider met de volgende gegevens toe  
Naam: Nicolai Roos  
Telefoonnummer: 079-3317731  
Na het toevoegen druk u op de link terug.
10. U ziet dat de projectleider is toegevoegd

**Vragen, opmerkingen, wensen of eisen m.b.t. toevoegen:**

**Figuur 3.6: Workshop 1**

Naar aanleiding van de workshop kwam naar voren dat er binnen het planningsysteem een grote verandering moest worden doorgevoerd (nadere uitleg in hoofdstuk 5.2 planningsysteem). Deze verandering zorgde ervoor dat de planning herzien diende te worden. Als gevolg van deze verandering is de invoering uitgesteld tot na de afstudeerperiode en is ervoor gekozen om alleen het volledige systeem te testen.



## 4 Definitiestudie BonaPlan

De definitiestudie heeft in het afstudeerproject een belangrijke plaats ingenomen. Het is de eerste stap geweest in de richting van het BonaPlan. In dit hoofdstuk worden de activiteiten beschreven die zijn uitgevoerd in het kader van de definitiestudie. Daarnaast worden de gemaakte keuzes, gebruikte technieken, problemen en resultaten beschreven. Alle activiteiten binnen de definitiestudie werden afgestemd met de opdrachtgever doormiddel van gesprekken welke zijn gevoerd nadat de opdrachtgever de definitiestudie had gelezen. De gehele definitiestudie is als externe bijlage in te zien.

Voor de start van de definitiestudie zijn er enkele gesprekken en interviews geweest. Deze gesprekken hadden ten doel inzicht te krijgen op de huidige manier van plannen, de wensen en eisen te achterhalen met betrekking tot BonaPlan en het achterhalen van de gegevens welke met BonaPlan moesten konden worden geregistreerd. Eén interview is toegevoegd als interne bijlage: Interview met betrekking tot systeemeisen.

Tevens is tijdens de definitiestudie de webserver gebouwd. De webserver was namelijk nodig voor het creëren van een test / ontwikkelomgeving en uiteindelijk bedoelt voor de invoering van BonaPlan. Zonder deze test/ ontwikkelomgeving kon BonaPlan niet gebouwd worden vandaar dat tijdens de definitiestudie deze webserver is gebouwd.

### 4.1 Plan van aanpak definitiestudie BonaPlan

In het plan van aanpak staat gespecificeerd welke activiteiten er in deze fase zullen worden uitgevoerd, uit welke onderdelen de definitiestudie zal gaan bestaan en hoeveel tijd deze onderdelen in beslag zullen gaan nemen. Het plan van aanpak voor de definitiestudie bepaald de aanpak gedurende deze fase. Het belangrijkste onderdeel uit het plan van aanpak was de planning. In deze planning staat nauwkeurig vermeld welke activiteiten wanneer dienden te worden uitgevoerd en hoeveel tijd hiervoor beschikbaar was. Het was belangrijk dat de aanpak van de definitiestudie al in een vroeg stadium van het project wordt vastgesteld. Het plan van aanpak hielp structuur te geven aan de werkzaamheden en sloot vertraging en het niet halen van deadlines zo veel mogelijk uit. Het plan van aanpak van de definitiestudie is afgeleid van het globale plan van aanpak dat aan het begin van het project was opgesteld.

### 4.2 Analyse

De analyse van de definitiestudie is gebruikt om erachter te komen hoe de huidige manier van plannen in elkaar steekt en wat de problemen hierbij zijn. Middels gesprekken met dhr. Molenkamp (de huidige planner) is het planproces doorgenomen. De keuze voor gesprekken met dhr. Molenkamp was een logische keuze aangezien hij de huidige manier van plannen heeft ontwikkeld, opgezet en uitvoert.

#### Huidige manier van plannen

De huidige planning bestaat zoals eerder vermeld uit verschillende Excel-sheets. Wanneer een aanvraag of opdracht binnen komt wordt deze toegevoegd in een van de volgende Excel-sheets:

- Planning tekenkamer indien project bestaande bouw > 1 woning of nieuwbouw;
- Planning mutatie indien individuele woning complete CV;
- Planning verketelingen via bedrijfsbureau indien verketeling of slangen vervangen.

Na verloop van tijd komt meer informatie binnen wat in de diverse lijsten wordt toegevoegd, zoals: type radiator en type ketel.

Op een gegeven moment wordt de uitvoeringsdatum bekend en dan wordt dit in de diverse bovenstaande planningen ingevuld, daarnaast worden deze ook nog in de volgende planningen gezet:

- Planlijst 2004 (in 2005 is dat planlijst 2005);
- Planlijst Ronald (verketelingen);
- Balkenplanning.

Na uitvoering wordt het project in de diverse planningen gereed gemeld.

### Probleem en doelstelling

De huidige manier van plannen brengt enkele problemen met zich mee, het is namelijk omslachtig en niet efficiënt genoeg. Daarnaast brengt het invullen van dezelfde gegevens in verschillende planningen gevaar van inconsistentie met zich mee. Tevens wil men meer inzicht geven voor de klant in het gehele proces voor de probleem en doelstelling wil ik dan ook de stukken uit de opdrachtomschrijving citeren (opdrachtomschrijving is als interne bijlage toegevoegd).

*De grote groei van Bonarius verwarming heeft er voor gezorgd dat er behoefte is aan een andere manier van plannen. Waarbij in het kader van klantgericht werken deze via internet voor opdrachtgevers (klanten) beschikbaar moet zijn voor status navragen. Op dit moment worden alle planningen gemaakt in Excel. Alle overzichten worden handmatig gemaakt en weer ingevoerd in Excel. Dit is niet alleen een tijdrovende klus maar ook erg omslachtig.*

*Het doel van de afstudeeropdracht is het ontwerpen, ontwikkelen en implementeren van een online planningssysteem voor Bonarius verwarming (BonaPlan). Het planningssysteem moet ervoor zorgen dat mensen en middelen efficiënt, en minder omslachtig, kunnen worden ingepland voor de verschillende projecten.*

*Tevens dient het planningssysteem meer inzicht te geven aan opdrachtgevers (klanten) zodat zij ten alle tijden de status van een project kunnen bekijken. Dit is onderdeel van de verdere professionalisering van Bonarius verwarming. Ook wil men middels deze functie van BonaPlan de service van Bonarius Verwarming proberen te ontlasten (minder status navragen).*

Figuur 4.1: Probleem en doelstelling

### 4.3 Ontwikkelscenario BonaPlan

Het ontwikkelscenario is een adequate manier om na te kunnen denken over de eventuele pilots waaruit het systeem gaat bestaan. Zo kwam ik erachter dat bij de ontwikkeling van het systeem er een kernel-pilot ontwikkeld diende te worden. Een pilot is een op zichzelf werkend deel van een systeem die als zelfstandige, bruikbare eenheid kan worden ingevoerd in de organisatie. Een kernel-pilot daarentegen wordt uitsluitend ontwikkeld als basis voor de ontwikkeling van latere pilots. Aangezien BonaPlan gebruik maakt van een database, en een database een niet op zichzelf werkend deel van het systeem is wat tevens de basis vormde voor de hier opvolgende pilots, was de eerste kernel-pilot geboren.

Tijdens de ontwikkelstrategie zijn de pilotstrategie en de iteratiestrategie bepaald. De keuze voor de pilots van het systeem was als volgt:

1. Database (Kernel-pilot);
2. Planningssysteem;
3. Gegevensbeheer;
4. Overzichten.

Waarom het systeem in deze pilots opgedeeld is, is als volgt:

#### 1. Database

De database is een apart onderdeel van het systeem welke eerst ontwikkeld diende te worden voordat de andere onderdelen ontwikkeld konden worden. De database werd dan ook als eerste ontwikkeld worden en was dus tevens een kernel-pilot (zie paragraaf 5.3).

#### 2. Planningssysteem

BonaPlan is de naam van het gehele systeem. Het systeem bestond uit een planningsgedeelte en een gedeelte dat overzicht gaf voor onder andere de opdrachtgever. Het planningssysteem was dan ook het tweede gedeelte van het systeem omdat dit alle functionaliteiten omtrent het plannen omvatte.

#### 3. Gegevensbeheer

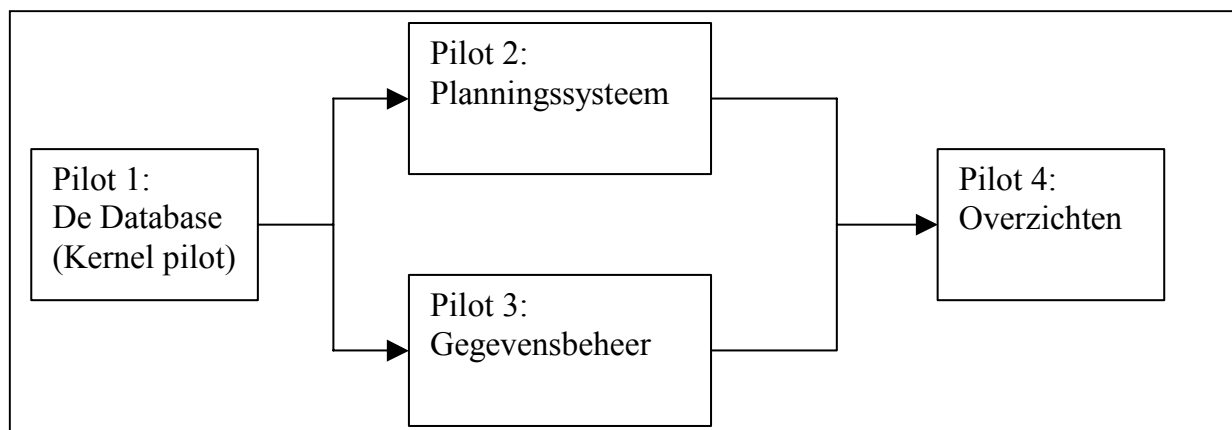
BonaPlan is een planningssysteem voor projecten. Projecten moesten dan ook geregistreerd kunnen worden binnen dit systeem. Een project bevat verschillende gegevens welke bestaan uit een vaste keuze voor ieder project. Zo zijn er werktypes, statussen, radiatoren, projectleiders, uitvoerders, urgenties, ketels, opdrachtgevers en monteurs. Dit zijn alle gegevens die aan een willekeurig project kunnen worden toegevoegd. Deze gegevens dienden dan ook apart beheerd te kunnen worden. Onder het beheren wordt toevoegen,

wijzigen en verwijderen verstaan. Aangezien dit een aanzienlijk deel van het systeem betreft en het op zichzelf kan werken is dit tot derde pilot gekozen.

#### 4. Overzichten

Naast het kunnen plannen binnen BonaPlan was een ander belangrijk gedeelte de inzicht voor de opdrachtgever, en de medewerkers binnen het bedrijf. De vierde en laatste pilot bevatte dan ook het gedeelte van BonaPlan wat het overzicht (via internet) voor de opdrachtgevers verzorgde en de overzichten welke binnen BonaPlan voor alle medewerkers zoals bijvoorbeeld agenda's en statistieken en projectlijsten aanwezig dienden te zijn. Aangezien dit stuk wederom als apart gedeelte van het systeem gezien kon worden is deze benoemd tot de vierde pilot.

Voor deze vier pilots is een strategie gekozen voor de ontwikkeling hiervan. Besloten was dat er begonnen zou worden met de database, dit was immers een kernel-pilot. Hierna werd er begonnen met de pilots "Planningssysteem" (pilot twee) en "Gegevensbeheer" (pilot drie). Gekozen is om deze parallel aan elkaar te ontwikkelen. Dit omdat het plangedeelte van BonaPlan projecten vereist en projecten gegevens vereist die in pilot drie kunnen worden beheerd. Door deze twee pilots parallel te ontwikkelen kon het systeem tussentijds beter getest worden. De pilots zijn niet samengevoegd tot één pilot omdat dit samen een te groot gedeelte van het systeem zou omvatten en er zo onduidelijkheid zou kunnen ontstaan over de functionaliteiten en het overzicht van de gehele pilot. Pilot "Overzichten" (pilot vier) volgde hier achteraan omdat pilot vier de gegevens van de vorige pilots op een andere manier (in overzichten) visualiseert en dus pas bruikbaar en functioneel was wanneer pilot twee en drie ontwikkeld waren. Tevens bood deze pilot het inzicht voor de klanten (opdrachtgevers) van Bonarius en dit kan en mag pas beschikbaar zijn voor gebruik wanneer het volledige systeem binnen Bonarius ontwikkeld is. De voorgaande strategie is in figuur 4.2 gevisualiseerd.



Figuur 4.2: Pilotstrategie

Naast de pilot strategie was er in dit stadium van het project een keuze gemaakt voor de iteratiestrategie. Binnen IAD bestaan er een aantal gedefinieerde iteratiestrategieën. Eén daarvan is het zogenoemde "Big-Bang-invoeren", dit is dan ook de strategie die ik gekozen heb (zoals beschreven in paragraaf 3.1.2).

#### 4.4 Systeemeisen BonaPlan

De systeemeisen vormen de eisen waaraan het systeem moet voldoen en waar het product aan het einde op getest kon worden. Deze moesten dan ook zorgvuldig worden opgesteld en tijdens het ontwikkelen van de pilots in de gaten worden gehouden. De definitiestudie werd dan ook regelmatig geraadpleegd tijdens de ontwikkeling van de pilots. Dit om te toetsen of de pilot voldeed aan de vooraf gestelde eisen.

Voor de systeemeisen heb ik interviews/gesprekken gevoerd met de opdrachtgevers (bedrijfsmentors) en de eindgebruiker. Om deze interviews en gesprekken te ondersteunen zijn er tijdens deze gesprekken schetsen van het product gemaakt. Deze schetsen hebben geholpen om de eisen van de gebruikers en opdrachtgevers specifieker te maken. Daarnaast zijn doormiddel van de workshops welke gehouden zijn, de systeemeisen nader gespecificeerd. Deze systeem eisen zijn in de eerste volgende iteratie van het project aangepast en ontwikkeld.

De systeemeisen heb ik onderverdeeld in twee groepen. Namelijk de functionele eisen en de non-functionele eisen. De functionele eisen heb ik weer onderverdeeld in algemene eisen, eisen voor de beheerder, eisen voor de planner, eisen voor de opdrachtgever en eisen voor de viewer (dit zijn de vier actoren die het systeem onderscheidt). De non-functionele eisen heb ik onderverdeeld in de interface eisen, de integriteitseisen, de performance eisen en de operationele eisen. In figuur 4.3 volgt een fragment uit de functionele systeemeisen.

#### **Functionele systeemeisen**

De functionele systeemeisen van het te ontwikkelen meldingsysteem zijn:

1. Het systeem dient 4 actoren te onderkennen beheerder, viewer, planner en een opdrachtgever.
2. Het systeem dient beveiligd te zijn met een inlogschermb en alleen toegang tot de gebruiker te verschaffen nadat hij een juiste combinatie van gebruikersnaam heeft ingevoerd. Afhankelijke van deze combinatie wordt het actortype bepaald.
3. Het informatiesysteem moet de volgende functionaliteiten bieden:  
Voor de **Planner (P)**:
  1. Planner moet projecten kunnen plannen (**P1**)
  2. Planner moet projecten kunnen toevoegen (**P2**)
  3. Planner moet projecten kunnen wijzigen (**P3**)
  4. Planner moet projecten kunnen verwijderen (**P4**)
  5. Planner moet Monteurs kunnen plannen in combinatie met een project (**P5**)
  6. Planner moet Monteurs kunnen toevoegen, wijzigen (**P6**)
  7. Planner moet Opdrachtgevers kunnen toevoegen, wijzigen (**P7**)
  8. Planner moet Opdrachtgevers kunnen toekennen aan een project (**P8**)
  9. Planner moet Projectleiders kunnen toevoegen, wijzigen (**P9**)
  10. Planner moet Projectleiders kunnen toekennen aan een project (**P10**)
  11. Planner moet Ketels kunnen toevoegen, wijzigen (**P11**)
  12. Planner moet Ketels kunnen toekennen aan een project (**P12**)

**Figuur 4.3: Fragment systeemeisen BonaPlan**

In dit fragment is te zien dat alle systeemeisen genummerd zijn “(**P1**)”. Deze nummering is toegepast om zo de systeemeisen in opvolgende documenten te kunnen identificeren.

Systeemeis drie is opgesplitst naar de vier actoren welke het systeem onderscheidt:

- Beheerder;
- Planner;
- Viewer;
- Opdrachtgever.

In het fragment uit figuur 4.3 ziet u een gedeelte van de systeemeisen voor de Planner. Op deze manier zijn ook de eisen voor de andere actoren uitgeschreven.

De systeemeisen kunnen worden ingezien in de externe bijlage: Definitiestudie.

De systeemeisen zijn gedurende het project regelmatig gewijzigd. Gedurende het ontwikkelen van de vier pilots werd er een lijst bijgehouden van functionaliteiten die in de volgende iteratie gewijzigd of juist toegevoegd moesten worden. De systeemeisen werden veroorzaakt doordat de wensen en eisen van de opdrachtgever steeds concreter en specifiekter werden naar mate het project vorderde. Neem bijvoorbeeld de verandering naar aanleiding van workshop één (zie paragraaf 5.2).

## **4.5 Systeemconcept BonaPlan**

Het systeemconcept geeft het te ontwikkelen systeem conceptueel weer zodat het eenduidig genoeg is voor de ontwikkelaars en toch begrijpelijk genoeg is voor de opdrachtgever. Dit systeemconcept heb ik opgesteld omdat ik een eenduidig overeenstemming wilde bereiken met de opdrachtgever, maar ook omdat een conceptuele weergave van de werkelijkheid noodzakelijk was voor de pilotontwikkeelfase. In het systeemconcept heb ik de use-cases opgesteld, een klassediagram en een use-case diagram opgesteld.

Use-cases zijn een middel om de functionele eisen die gesteld worden aan het informatiesysteem weer te geven. Ze gaan uit van de gebruiker van het systeem en beschrijven hoe hij met het systeem om wilt gaan. De use-cases zijn opgesteld aan de hand van de systeemeisen. Sommige use-cases bestaan uit een enkele systeemeisen, andere uit een cluster van systeemeisen. Als voorbeeld zal de

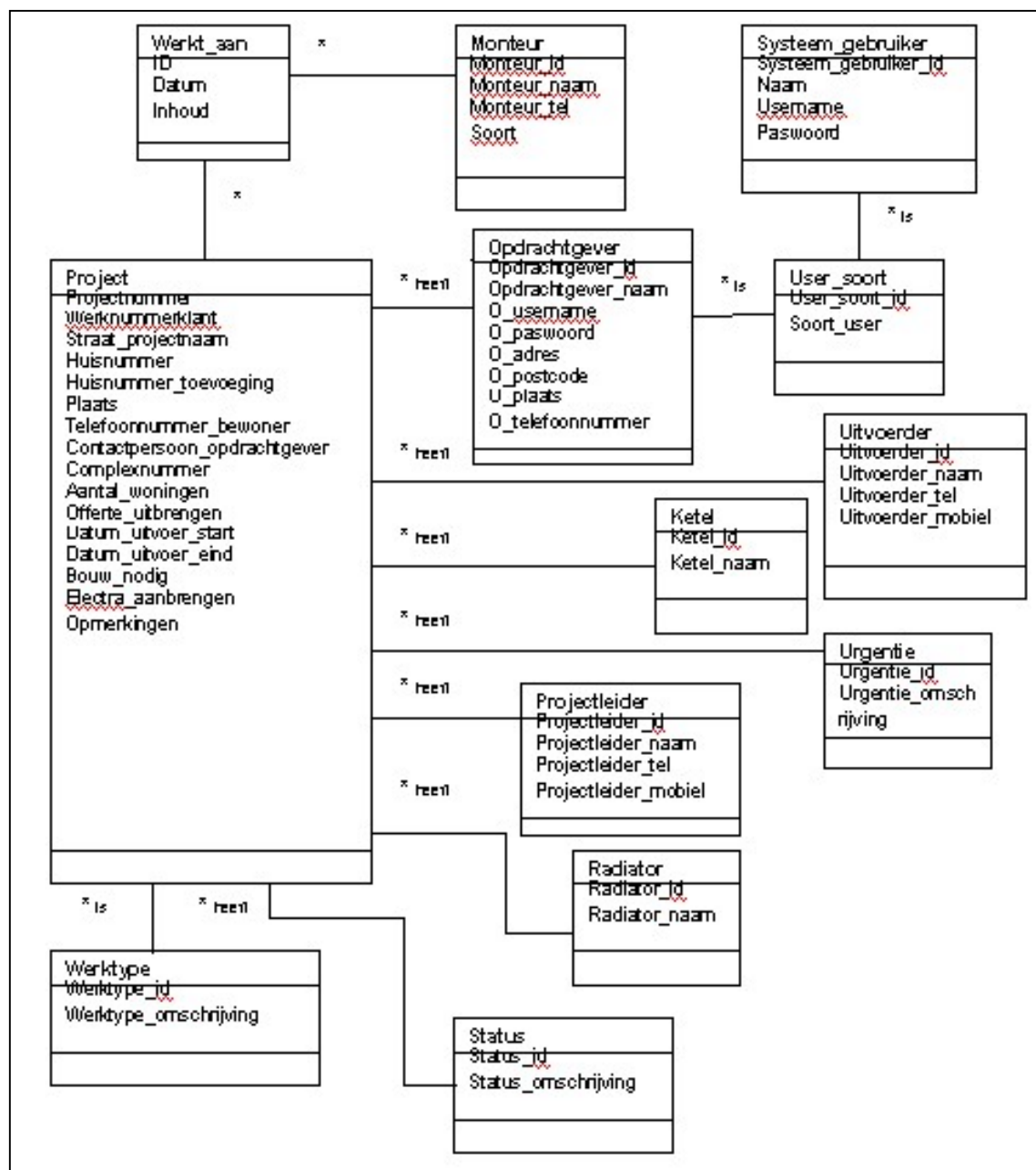
use-case project toevoegen worden getoond. Project toevoegen kan zowel door de beheerder als door de planner worden gedaan. Deze use-case heeft dat ook de betrekking op systeem eis P2 en B2. figuur 4.4 laat de use-cases zien, alle use-cases en het use-case diagram zijn te vinden in de externe bijlage: Definitiestudie.

<b>USE-CASE 3 (U3)</b>	
Naam	Project toevoegen.
Samenvatting	Er wordt een nieuw project toegevoegd in het systeem.
Actoren	Beheerder, Planner.
Aannamen	Beheerder, Planner beschikt over de minimaal in te voeren gegevens voor een project. Beheerder, Planner is ingelogd. Beheerder, Planner bevindt zich op de projecten pagina.
Beschrijving	<ol style="list-style-type: none"> <li>(1) Beheerder, Planner maakt aan het systeem bekend dat hij/zij een project wil toevoegen aan het systeem en geeft de minimaal benodigde gegevens (Projectnummer, Straat_projectnaam en Plaats) in. Tevens wordt de mogelijkheid gegeven op een werktipe, opdrachtgever, ketel, uitvoerder, urgentie, projectleider, radiator en status toe te kennen aan het project.</li> <li>(2) De Beheerder, Planner verzendt de gegevens</li> <li>(3) Het systeem checkt of de gegevens compleet en van het juiste format zijn. Is dit niet het geval dan treden de uitzonderingen incomplete gegevens en/of verkeerde format op.</li> <li>(4) Het systeem slaat het project op en meldt dit aan de opdrachtgever.</li> </ol>
Uitzonderingen	<p>[incomplete gegevens] Als de Beheerder of planner niet alle benodigde gegevens invoert meldt het systeem dat het project niet kan worden opgeslagen i.v.m incomplete gegevens De Beheerder of planner dient de use-case opnieuw te starten.</p> <p>[Verkeerde format] Als de beheerder of planner de gegevens in een verkeerd format invoert meldt het systeem dat het project niet kan worden opgeslagen i.v.m een verkeerde format fout. De beheerder of Planner dient de use-case opnieuw te starten.</p>
Resultaat	Er is minstens één project toegevoegd.

**Figuur 4.4: Use-case project toevoegen**

De use-case project toevoegen maakt het voor de gebruiker mogelijk om een project op te slaan in het systeem en hier tevens een aantal attributen aan toe te kennen. Zo kan de gebruiker aangeven om wat voor werktipe het gaat, de bijbehorende opdrachtgever, ketel, uitvoerder, urgentie, projectleider, radiator en tevens de status van het project.

Nadat er een concreet beeld van de interactie tussen mens en machine was gecreëerd, middels de use-cases konden de componenten (klassen) van het systeem gemaakt worden om deze interacties te ondersteunen. Deze stap visualiseert zich in het zogenoemde klassediagram. Een klassediagram is een model dat alle klassen en attributen van de te ontwikkelen database bevat en de onderlinge relaties tussen de verschillende klassen laat zien. Dit klassediagram is opgesteld op basis van de systeemeisen in de definitiestudie en gebaseerd op de gegevens uit de originele plannings. In een gesprek met dhr. Molenkamp hebben wij samen besloten welke gegevens BonaPlan diende te bevatten. Na dit gesprek zijn al deze gegevens gecategoriseerd en deze categorieën zijn daarna de uiteindelijke klassen van de database geworden. Daarnaast is deze database besproken met de bedrijfsmentor dhr. Dee aangezien deze over enige ervaring met databases beschikte. Het klassediagram was een belangrijk onderdeel van het project omdat pilot twee, drie en vier hiervan afhankelijk waren (de database welke o.a. aan de hand van het klassediagram is ontwikkeld). Het klassediagram is in figuur 4.5 te zien.



Figuur 4.5: Klassediagram

Als we even terugkijken naar figuur 4.4 zien we dat de gegevens welke aan een project kunnen worden toegekend op zijn genomen als aparte klassen. Zo kan bijvoorbeeld een projectleider meerdere projecten leiden. Daarnaast hoeft de projectleider maar één keer in het systeem te worden geregistreerd en kan deze daarna binnen BonaPlan aan alle projecten worden gekoppeld. Alle gegevens uit deze tabellen, dienden in het toekomstige systeem door de beheerder geregistreerd, verwijderd en gewijzigd te kunnen worden.

## 4.6 Technische structuur voor BonaPlan

In dit onderdeel van de definitiestudie heb ik de huidige technische structuur geanalyseerd en tevens de toekomstige architectuur beschreven. Het nut hiervan is dat in een vroegtijdig stadium van het project eventuele investeringen ten behoeve van het project gesignaleerd konden worden. Tijdens het schrijven van de technische structuur kwam er naar voren dat er een webserver gebouwd diende te worden, als test/ontwikkelomgeving en later voor de implementatie van BonaPlan. Bij het opzetten van deze webserver heb ik dan ook voor verschillende keuzes gestaan. Hieronder zullen deze keuzen kort behandeld en toegelicht worden.



## **Besturingssysteem**

Uiteraard was er een besturingssysteem nodig om een ontwikkelomgeving voor een informatiesysteem op te bouwen. Op de taken en de doelen van een besturingssysteem zal ik hier verder niet op ingaan, die veronderstel ik als bekend.

De keuze voor het besturingssysteem was snel gemaakt. Omdat dit project zo min mogelijk kosten met zich mee moest brengen, er zoveel mogelijk gebruik diende te worden gemaakt van de aanwezige apparatuur viel de keuze vrijwel direct op "Windows 2000 server". Dit omdat die al ter beschikking stond binnen het bedrijf en ten tweede is dit systeem een bewezen alternatief in de webserver markt (door zijn stabiliteit, veiligheid en meegeleverde webserver software).

De kwalificatie stabiel en veilig behoeft nog enige toelichting. Windows 2000 server is op papier gelijkwaardig aan bijvoorbeeld Linux. Het voornaamste probleem bij Windows 2000 server is bekendheid. Doordat Windows 2000 server vergeleken met Linux door meer bedrijven gebruikt wordt, is deze een doelwit voor hackers. Echter door Windows 2000 server up-to-date te houden met de beschikbare updates (van Microsoft) en door de server te configureren met het oog op veiligheid is de stabiliteit en veiligheid, gegeven de beschikbare middelen, zo goed mogelijk te waarborgen. Tot slot wordt opgemerkt dat zich ook beveiligingsproblemen voordoen bij het gebruik van Linux of andere Unix besturingssystemen.

## **Webserver software**

Om een computer als webserver in te richten is er naast een besturingssysteem software nodig om deze als webserver te kunnen laten fungeren. Aan deze software waren de volgende eisen gesteld:

- De software moet kunnen draaien onder Windows 2000 server;
- De software moet goedkoop zijn in aanschaf en onderhoud;
- De software moet stabiel zijn.

Er zijn veel verschillende webserver verkrijgbaar, maar de markt wordt hoofdzakelijk door de exemplaren Apache en IIS (Internet-Information-Server) gedomineerd.

IIS wordt standaard met Windows 2000 server geleverd en Apache is gratis te downloaden van het internet. In overleg met de opdrachtgever is voor IIS gekozen, om de volgende redenen:

- IIS wordt al geleverd met de Windows 2000 besturingssysteem;
- Gemakkelijk te beheren en te configureren.

## **PHP of ASP**

Nadat de keuze voor de webserver bepaald was stond ik voor de keuze om PHP of ASP te gaan gebruiken. Aangezien BonaPlan een webapplicatie betreft, zou deze komen te bestaan uit dynamische-pagina's. Om de overstap van een statische-pagina (HTML pagina) naar een dynamische-pagina te maken, komen er andere technologieën om de hoek kijken zoals bijvoorbeeld de databasetechnologie en de ASP en PHP methode. In de bijlage PHP of ASP staat een korte geschiedenis van beide talen beschreven. In deze fase van het project heb ik dan ook veel over beide talen gelezen: de werking, welke hulpmiddelen heb ik nodig en wat zijn de voor en nadelen van beide.

Uiteindelijk is in overleg met de opdrachtgever gekozen voor het gebruik van PHP. En wel door de volgende voordelen:

1. PHP is gratis: In tegenstelling tot ASP. ASP licenties zijn erg kostbaar.
2. PHP is crossplatform: PHP draait onder elk besturingssysteem. ASP draait officieel alleen onder Microsoft producten.
3. PHP is snel: Zeker in het begin was PHP vele malen sneller dan ASP. Inmiddels is de performance van ASP aanzienlijk verbeterd maar is nog steeds niet sneller dan PHP
4. PHP en bugs: binnen PHP worden bugs sneller opgelost dan bij ASP. Dit heeft vooral te maken met de organisatie welke achter de scripting talen staat. PHP staat een relatief klein ontwikkelteam achter. ASP het grote Microsoft waardoor een verandering (bugfix) door vele lagen van de organisatie moet en dit kost veel tijd.

De belangrijkste reden voor de opdrachtgever om PHP te gebruiken was wel reden één. Daarbij was ik van mening dat het toekomstige systeem met behulp van PHP gebouwd kon worden. Bijkomend voordeel was dat ik al enige ervaring met PHP had.

## Database

Naast de keuze voor de webserver en de scripttaal stond ik nog voor de keuze welke databasetechnologie te gebruiken. Bij het kiezen van een databasetechnologie heb ik de volgende selectie voorwaarden gehanteerd:

- Databasetechnologie moet met PHP kunnen werken;
- Databasetechnologie moet relatief lage kosten met zich meebrengen;
- De databasetechnologie moet de verwachte belasting aankunnen en moet ook eventuele verhoging van de belasting aankunnen.

De databasetechnologieën die in beschouwing zijn genomen waren, MS SQL server, MySQL en MS Access. Al deze database technologieën kunnen werken in combinatie met PHP. MS Access voldeed niet aan het tweede selectie criterium omdat die niet beschikbaar was in het officepakket van het bedrijf. De opties MySQL en MS SQL bleven daardoor over, aangezien MySQL gratis is en MS SQL binnen Bonarius aanwezig is. De overgebleven twee database technologieën konden ook beiden de verwachte belasting van de database aan (we gingen uit van een aantal connecties van tien). Uiteindelijk is de keuze gemaakt voor MySQL. Dit is gedaan omdat de opdrachtgever de salarisadministratie, welke draait op MS SQL, gescheiden wil houden van BonaPlan. Daardoor diende er een nieuwe licentie van MS SQL aangeschaft te worden en dat zou te veel kosten met zich mee hebben gebracht, vandaar de keuze voor MySQL.

In de bovenstaande tekst valt te zien dat veel beslissingen zijn genomen aan de hand van de kosten die de keuze met zich meebrengt, dit is dan ook één van de belangrijkste eisen van de opdrachtgever geweest.

## 4.7 Organisatorische inrichting

In dit onderdeel van de definitiestudie is beschreven wat voor gevolgen BonaPlan op de huidige organisatorische inrichting zou hebben. Er is gekeken naar:

- De manier van plannen;
- De service tegenover de klant (opdrachtgever);
- Globale opleidingsplan;
- Specificatie gebruikersdocumentatie.

Door dit in een vroegtijdig stadium te beschrijven waren de gevolgen voor de organisatorische inrichting vroegtijdig bekend en kon hierop gereageerd worden.

Op dit moment gebeurt het plannen met behulp van Excel sheets. Deze worden met pen en papier ingevuld en later ingevoerd in de Excel sheets. Met de introductie van het nieuwe planningssysteem zal dit verandert gaan worden. Alle planningen zullen direct van achter de pc in het planningssysteem dienen te worden gemaakt. Zo zullen er meerdere medewerkers toegang krijgen tot het planningssysteem. Deze zullen met behulp van rechten verschillende delen van het systeem kunnen bezoeken en planningen kunnen wijzigen. Belangrijk is dat men niet meer op papier gaat plannen zodat het planningssysteem inconsistent raakt met de werkelijkheid. Bijkomend voordeel aan deze manier van plannen is dat de planning buiten de werkplek kan worden ingezien en ingevuld kan worden (bijvoorbeeld thuiswerken) tevens zijn alle planningen centraal opgeslagen.

BonaPlan zal naar de klant toe een extra service zijn. Status opvragen met betrekking tot projecten dienen niet meer via de administratie of FrontOffice van Bonarius te lopen, maar zoveel mogelijk via de website. Bonarius zal er voor kunnen zorgen dat middels een gebruikersnaam en een wachtwoord de klant de voortgang van zijn project kan volgen. Daarnaast kan de klant (opdrachtgever) de contactgegevens van de bijbehorende uitvoerder en projectleiders op de site opzoeken om zo direct met de juiste mensen in contact te kunnen komen zonder tussenkomst van de administratie of Frontoffice, deze zal dan ook ontlast worden.

Het systeem is zo ontwikkeld dat de eindgebruiker, middels een door de ontwikkelaar te schrijven handleiding en een korte introductiesessie, BonaPlan direct kan gebruiken. De opleiding van het planningssysteem zal dus ongeveer één dag in beslag nemen.



## 4.8 Pilotplan

In deze paragraaf wordt het pilotplan uit de definitiestudie besproken. In het pilotplan zijn de verschillende pilots nader toegelicht en word aangeven hoe het ontwerp van deze pilot eruit zal zien. Daarnaast word wederom een verder detaillering van het planning gemaakt. Het pilotplan is opgenomen in dit verslag om de lezer inzicht te verschaffen in de denkwijze die ik heb gevolgd bij het plannen van de pilots. Voor elke pilot heb ik een tabel gemaakt waarin de identificatie, afdekking en schatting van de benodigde tijd is beschreven.

ID	Omschrijving	Prioriteit	Geschatte tijd
Pilot 1	Het ontwerpen en bouwen van de database	Basis	7 dagen
Pilot 2	Het ontwerpen en bouwen van het planningsgedeelte	Basis	20 dagen
Pilot 3	Het ontwerpen en bouwen van het gegevensbeheer	Basis	7 dagen
Pilot 4	Het ontwerpen en bouwen van de overzichten	Basis	10 dagen

Figuur 4.6: Pilot identificatie

De schatting van de benodigde tijd is gebaseerd op eerdere ervaringen tijdens projecten op school. Tevens is er gekeken naar het aantal systeemeisen die worden afgedekt per pilot voor het schatten van de benodigde tijd per pilot.

## 4.9 Verificatie en validatie

In deze paragraaf behandel ik de verificatie en validatie van de definitiestudie. Dit is gedaan door de use-cases tegenover de systeemeisen uitgezet, hieronder een kort fragment:

Systeemeisen / Use-case	1	3 planner 2 (P)	3 Opdrachtgever (O)	3 Viewer (v)	3 Beheerder (B)	4
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						

Figuur 4.7: Verificatie en validatie

Op deze manier heb ik samen met de opdrachtgever kunnen verifiëren/valideren dat alle systeemeisen zijn ondervangen door de use-cases en daardoor uiteindelijk alle functionaliteiten van het systeem ontwikkeld zouden worden. Daarnaast zijn de systeemeisen af gezet tegenover de pilots, hieronder een kort fragment:

Systeemeisen / Pilots	1	3 planner 2 (P)	3 Opdrachtgever (o)	3 Viewer (v)	3 Beheerder (b)	4	5	6	7	8	9
Database											
Planningsysteem											
Gegevensbeheer											
Overzichten											

Figuur 4.8: Verificatie en validatie

Op deze manier heb ik samen met de opdrachtgever kunnen verifiëren/valideren dat alle systeemeisen door de pilots zijn ondervangen. Na dit alles gedaan te hebben kon ik beginnen aan de pilotontwikkeling van de verschillende pilots.

## 5 Pilotontwikkeling

In dit hoofdstuk word de procesgang van de pilotontwikkeling beschreven. In het ontwikkelscenario van de definitiestudie is besloten de te bouwen delen van het systeem op te delen in vier pilots, namelijk de database, het planningssysteem, het gegevensbeheer en de overzichten. Per pilot werd beschreven wat er gedaan moest worden, hoe dit werd aangepakt, de problemen die ik ben tegengekomen en de besluiten / oplossingen welke genomen zijn.

### 5.1 Pilot 1: De database

Tijdens deze pilot is er een database voor BonaPlan ontwikkeld. Deze database moest aan verschillende eisen voldoen welke zijn terug te vinden in de systeemeisen in de externe bijlage: definitiestudie. De geplande tijd voor deze pilot was geschat op een week, conform het pilotplan in de definitiestudie. Deze tijd is gebaseerd op ervaringen uit andere projecten en naar aanleiding van het klassediagram welke was ontwikkeld in de definitiestudie.

In deze pilot zijn de volgende onderdelen gerealiseerd:

- Rapport pilotontwikkelplan database:
  - Plan van aanpak;
  - Functionele structuur;
  - Pilot ontwikkelplan;
  - Verificatie en validatie.
- Database BonaPlan.

Als basis en ter ondersteuning van de bouw van de database is er een rapport pilotontwikkelplan opgesteld. Aan de hand van dit rapport kan de database worden gebouwd. Wanneer bijvoorbeeld de database verloren gaat (door een systeem crash) kan de database herbouwd worden aan de hand van dit document.

Aan het begin van deze pilot is er begonnen met een plan van aanpak. In het plan van aanpak zijn achtereenvolgens de reikwijdte, de resultaten en de planning beschreven. De planning was het belangrijkste onderdeel van het plan van aanpak. Door de eerder beschreven manier van plannen, moest via deze manier de overkoepelende planning verder gedetailleerd worden.

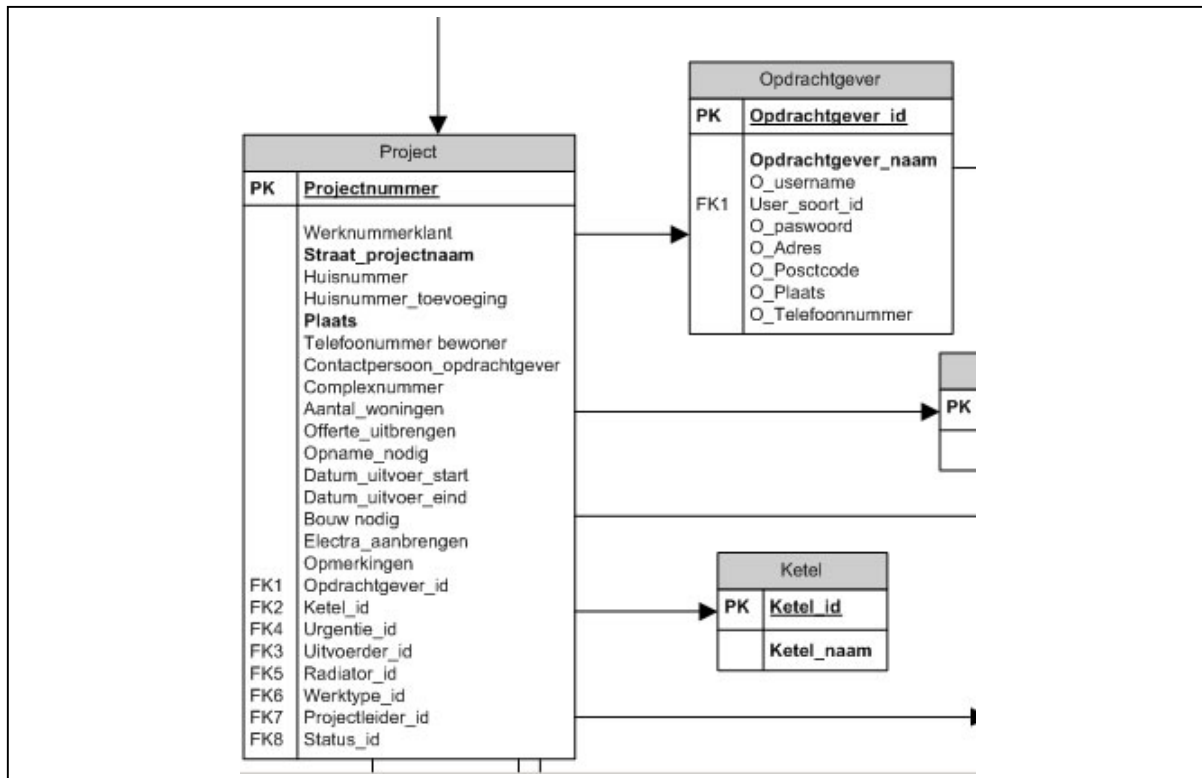
#### 5.1.1 Functionele structuur pilot database

Nadat het plan van aanpak was afgerond kon er worden begonnen met de functionele structuur van de pilot. De functionele structuur bestond uit een klassediagram, een relationeel representatie model en een relationeel implementatiemodel. Aan de hand van deze drie modellen kon de database volledig gebouwd worden. Deze modellen vond ik zeer geschikt voor het modelleren van de te ontwikkelen database aangezien zij de database duidelijk conceptueel weergeven en deze goed door de opdrachtgever te begrijpen waren. Deze modellen zijn opgesteld aan de hand van het klassediagram welke is opgenomen in het systeemconcept van de definitiestudie.

Het "klassediagram" dat gemaakt is in dit onderdeel verschilt op enkele punten van het klassediagram uit de definitiestudie en de regels die UML stelt aan het klassediagram. In dit klassediagram zijn namelijk de Primary keys (identificerende attributen van een klasse) en de foreign keys (de afgeleide attributen in een klasse) opgenomen in het klassediagram. De reden dat er voor deze vorm van een "klassediagram" is gekozen zijn:

1. Het creëren van de database word zo vergemakkelijkt (in een vroeg stadium zijn de primary en foreign keys bekend);
2. Het communiceren met de opdrachtgever met betrekking tot de database ging makkelijker omdat de opdrachtgever in één diagram kon zien wat de klassen en attributen zijn binnen de database en door welk attribuut iedere klasse geïdentificeerd kan werd.

Een fragment uit dit "klassediagram" is hieronder te zien.



Figuur 5.1: Fragment klassediagram

Uit dit klassediagram volgt het volgende onderdeel van de functionele structuur, te weten het relationeel representatiemodel (RRM), dit model is een tussenstap om zo makkelijker tot een relationeel implementatiemodel (RIM) te kunnen komen en om te kijken of de database correct genormaliseerd is. Hieronder is een fragment van het RRM te bezichtigen.

User\_soort (**User\_soort\_id**, Soort\_user)

Opdrachtgever (**Opdrachtgever\_id**, Opdrachtgever\_naam, O\_username, O\_paswoord, *User\_soort\_id*, O\_Adres, O\_Postcode, O\_Plaats, O\_Telefoonnummer)

Vreemde Sleutel User\_soort\_id refereert aan User\_soort\_id in User\_soort (Null allowed)

Figuur 5.2: Fragment relationeel implementatiemodel

In het bovenstaande voorbeeld zijn de **Primary keys** dik gedrukt, daarnaast zijn de *Foreignkeys* *schuin gedrukt*.

In dit voorbeeld valt tevens te zien dat een opdrachtgever van een bepaald user\_soort kan zijn. In een tabel ziet dat er als volgt uit (zie figuur 5.3).

User_soort ID	Soort_user
1	Beheerder
2	Planner
3	Viewer
4	Opdrachtgever

Opdracht gever_id	Opdracht gever_naam	O_ username	O_ paswoord	User_ soort_id	O_ Adres	O_ Plaats	O_ Postcode	O_Telefoon nummer
1	BAM	BAM	****	4	Lissenvaart	Zoetermeer	-	1234567890

**Figuur 5.3: Tabel User\_soort & tabel opdrachtgever**

Door de Foreign key koppeling in opdrachtgever kan een opdrachtgever van een bepaalde user\_soort zijn (in dit geval Opdrachtgever). Dit user\_soort wordt gebruikt in het login gedeelte om zo de gebruiker binnen het goede gedeelte van het systeem te laten binnen komen en daar de juiste rechten aan te kunnen toekennen.

Na het RMM volgt het RIM, hieronder is wederom een kort fragment ter illustratie te zien.

```
CREATE TABLE Opdrachtgever (
  Opdrachtgever_id          int(11)                NOT NULL auto_increment,
  Opdrachtgever_naam        varchar (50)            NOT NULL
  O_username                 varchar(25)             default NULL,
  O_paswoord                 varchar(25)             default NULL,
  O_Adres                    varchar(50)             default NULL,
  O_Postcode                 varchar(50)             default NULL,
  O_Plaats                   varchar(50)             default NULL,
  O_Telefoonnummer           varchar(11)             default NULL,
  User_soort_id              int (5)                 default NULL,

  FOREIGN KEY User_soort_id REFERENCES User_soort_id in User_sort,
  PRIMARY KEY Opdrachtgever_id)
```

**Figuur 5.4: Fragment relationeel implementatiemodel**

Het bovenstaande RIM werd ontwikkeld aan de hand van het eerder beschreven RRM en bevatte daarom ook geen nieuwe klassen/ of attributen. In dit geval wordt de klasse opdrachtgever met zijn bijbehorende attributen aangemaakt.

### 5.1.2 Pilotontwikkelplan database

Nadat het plan van aanpak en de functionele structuur waren gemaakt kon er begonnen worden met het pilotontwikkelplan. In dit onderdeel heb ik ervoor gekozen om het volgende te beschrijven:

- Beschrijving van de klassen;
- Beschrijving van de attributen;
- Planning van de pilot;
- Invoering van de pilot;
- Testen van de database.

De beschrijving van de klassen en de attributen zijn er om voor de ontwikkelaar en de opdrachtgever inzicht te geven in welke gegevens er allemaal in de database worden opgeslagen.

Naast deze beschrijving is de bouw van de database ingepland. Voor de gehele database (Ontwikkeling en Bouw) stond een week gepland, daarnaast is tijdens de definitiestudie al onderzoek gedaan naar wat het systeem allemaal aan gegevens diende te bevatten. Tijdens de bouw van de pilot leek de week die gepland stond meer dan genoeg tijd voor de bouw van de database. Deze week was gebaseerd op de gedachte dat de database twee iteraties zou doorlopen maar naar aanleiding van de grote verandering m.b.t. tot workshop één (zie paragraaf 5.2) heeft database een extra iteratie doorlopen met het gevolg dat de ingeplande tijd hard nodig was.

Nadat de database ontwikkeld was, moest deze ingevoerd worden. Het invoeren is gedaan middels het RIM en de Front-End voor MySQL: PHPMyAdmin. PHPMyAdmin biedt de mogelijkheid om een RIM direct in te voeren in een MySQL database. Het invoeren van de database was dan ook een snelle en makkelijke taak aangezien PHPMyAdmin al was geïmplementeerd tijdens de definitiestudie.

Nadat de database geïmplementeerd was zijn er voor alle tabellen twee rijen aan testgegevens met behulp van PHPMyAdmin ingevoegd. Vervolgens was het tijd om de database te testen. Dit is gedaan met behulp van een script dat de tabellen uit de database toont met de ingevulde testgegevens. Hiermee werd dus gekeken of interactie met de database mogelijk was en of de ingevulde testgegevens daadwerkelijk op te halen waren. Het script zag er als volgt uit.

```
<?php

//connectie met de database
$hostname = "localhost";
$username = "root";
$password = "*****";
$db = "bonaplan";

$global_dbh = mysql_connect($hostname, $username, $password);
mysql_select_db($db, $global_dbh);

//functie om tabellen uit te lezen
function display_db_table($tablename, $connection)
{
    $query_string = "select * from $tablename";
    $result_id = mysql_query($query_string, $connection);
    $column_count = mysql_num_fields($result_id);

    print("<TABLE BORDER=1>\n");
    while ($row = mysql_fetch_row($result_id))
    {
        print("<TR ALIGN=LEFT VALIGN=TOP>");
        for ($column_num = 0;
            $column_num < $column_count;
            $column_num++)
            print("<TD>$row[$column_num]</TD>\n");
        print("</TR>\n");
    }
    print("</TABLE>\n");
}
//bouw van de pagina
?>

<HTML>
<HEAD>
<TITLE>Database overzicht</TITLE>
</HEAD>
<BODY>

<TABLE><TR><TD>
<?php display_db_table("ketel", $global_dbh); ?>
</TD><TD>
<?php display_db_table("monteur", $global_dbh); ?>
</TD><TD>
enz...
```

**Figuur 5.5: Testcode database**

Een korte uitleg voor deze code is op zijn plaats. Binnen de code is commentaar geplaatst (// xx) aan de hand van dit commentaar zal de code doorlopen worden.

#### **//connectie met de database**

Aan de hand van deze code word er een connectie gemaakt met de database. Voor een connectie met een MySQL database zijn de gegevens "hostname" (plaats van database), "username" (gebruikersnaam voor de database), "password" (paswoord voor de database) en "db" (naam van de database) benodigd. Daarna wordt middels de functie mysql\_connect de daadwerkelijke connectie met de database gemaakt.

### //Functie om tabellen uit te lezen

Met deze functie worden alle attributen van een tabel opgevraagd en geplaatst in een tabel.

### //Bouw van de pagina

Door vervolgens de functie die hierboven gedefinieerd is aan te roepen voor iedere tabel in de database zullen alle gegevens uit de database op het scherm getoond worden en kan de conclusie getrokken worden dat de database al dan niet te bevragen valt en of er een connectie met database kan worden gemaakt.

### 5.1.3 Verificatie en validatie pilot Database

De verificatie en validatie van de database bood de mogelijkheid om te controleren of alles was ontwikkeld wat ontwikkeld diende te worden. Dit was belangrijk omdat ik graag met een goed gevoel verder wil gaan met de ontwikkeling van de volgende pilot en omdat de volgende pilots afhankelijk waren van deze pilot. Verificatie en validatie is uitgevoerd door te kijken met de opdrachtgever of alle informatie in de database kan worden opgeslagen (is alle informatie die dient te worden opgeslagen opgenomen in de klassen en bijbehorende attributen). Daarnaast is er gekeken of de klassen en attributen overeenkomen met het klassediagram, het RRM en RIM. Door dit te doen werd gecontroleerd of de modellen en de uiteindelijke database consistent met elkaar waren. Hieronder is een fragment te zien van deze verificatie en validatie.

Klasse	Attribuut	In klassendiagram	In R.M.M.	In R.I.M.
Uitvoerder	Uitvoerder_id	x	x	x
	Uitvoerder_naam	x	x	x
	Uitvoerder_tel	x	x	x
	Uitvoerder_mobiel	x	x	x
Monteur	Monteur_id	x	x	x
	Monteur_naam	x	x	x
	Monteur_tel	x	x	x
	Soort	x	x	x

Figuur 5.6: Verificatie en validatie database

## 5.2 Pilot 2: Planningssysteem

Tijdens deze pilot is het planningsgedeelte van BonaPlan gebouwd. Het planningsgedeelte moest aan verschillende eisen voldoen welke zijn terug te vinden in de systeemeisen in de definitiestudie. Het planningssysteem is het belangrijkste onderdeel van BonaPlan en omvat de gehele functionaliteit voor het plannen van projecten. Daarnaast moeten deze planningen natuurlijk bekeken kunnen worden.

Deze pilot werd parallel ontwikkeld met pilot drie: gegevensbeheer. Hier is voor gekozen omdat deze twee pilots nauw verbonden met elkaar zijn en omdat pilot twee, pilot drie nodig heeft voor het testen. Parallel moet in dit geval geïnterpreteerd worden in de vorm van het werken aan twee pilots op een dag in een willekeurige volgorde, dit omdat het project door mijzelf alleen ontwikkeld werd zodat letterlijk parallel ontwerpen onmogelijk is. De onderdelen van pilot drie zijn dan ook ontwikkeld op het moment dat ik ze tijdens het planningssysteem nodig had.

In de pilot zijn de volgende onderdelen gerealiseerd:

- Rapport pilotontwikkelplan planningssysteem;
  - Plan van aanpak;
  - Functionele structuur;
  - Organisatorische inrichting;
  - Pilot ontwikkelplan;
  - Verificatie en validatie.
- Pilot twee: Planningssysteem.

Als basis en ter ondersteuning van de bouw van het planningssysteem is er een rapport pilotontwikkelplan opgesteld. In dit rapport is het planningssysteem ontworpen. Aan de hand van dit rapport is vervolgens het planningssysteem gebouwd.

Aan het begin van deze pilot is er begonnen met een plan van aanpak. In het plan van aanpak zijn achtereenvolgens de reikwijdte, de resultaten en de planning beschreven. De planning was het belangrijkste onderdeel van het plan van aanpak, door de eerder beschreven manier van plannen, moest via deze manier de overkoepelende planning verder gedetailleerd worden.

Tijdens de definitiestudie is er bepaald dat het ontwikkelen van BonaPlan twee iteratie in beslag zou nemen namelijk:

1. De eerste iteratie voor het opzetten van BonaPlan;
2. De tweede iteratie voor aanpassingen wensen en eisen van de opdrachtgever.

Aan het eind van de pilots twee en drie is daar verandering in gekomen. Aangezien het planningsgedeelte de kernfunctionaliteit omvat van BonaPlan, had ik besloten om nadat de eerste iteratie van het planningssysteem en gegevensbeheer was afgerond een workshop te houden. Deze workshop is dan ook gehouden in de negende week van het afstuderen (13-07-2004). Deze workshop is gehouden met de eindgebruiker (tevens bedrijfsmentor) dhr. Molenkamp. Dit is gedaan doormiddel van het uit voeren van de volgende opdrachten:

- Toevoegen, wijzigen en verwijderen van een projectleider;
- Toevoegen, wijzigen en verwijderen van een project.

Daarnaast is er een demonstratie door mij gegeven over het plannen van een project met behulp van het systeem.

Deze workshop leverde veel reactie en input op voor een volgende iteratie, zelfs zoveel dat er besloten is om een extra iteratie uit te voeren op het project en wel direct vanaf de dag van de workshop. De belangrijkste verandering n.a.v. deze workshop was dat het plannen van projecten niet in dagen moest gebeuren maar met weeknummers: Voer een startweek in (bijvoorbeeld 46) en een eindweek (47) en geeft daarbij aan op welke dagen (maandag, dinsdag, woensdag, donderdag, vrijdag, zaterdag of zondag) dit project door monteur x dient te worden uitgevoerd. Nadat de gebruiker op de knop plannen drukt zal het programma voor week 46 & 47 op de geselecteerde dag(en) het project en de monteur plannen. Wil men één week of zelfs één dag invullen voor een klus dan is de startweek hetzelfde als de eindweek en geef je dus bijvoorbeeld alleen maandag aan. (zo creëer je dus een ééndaagse klus).



De bovenstaande situatie levert de onderstaande schermschets op.

Projectnaam	<input type="text"/>	ma di wo do vr za zo
Monteur	<input type="text" value="listbox"/>	<input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/> <input type="radio"/>
Startweek	<input type="text"/>	
Eindweek	<input type="text"/>	<input type="button" value="Plannen"/>

Figuur 5.7: Schermschets naar aanleiding van workshop 1

Door deze wijziging is BonaPlan dus uiteindelijk ontwikkeld aan de hand van drie iteratie's

1. Iteratie 1: Definitiestudie, pilot 1, pilot 2 en pilot 3 (geen hele iteratie);
2. Iteratie 2: Definitiestudie, pilot 1, pilot 2, pilot 3 en pilot 4 (gehele iteratie);
3. Iteratie 3: Definitiestudie, pilot 1, pilot 2, pilot 3 en pilot 4 (gehele iteratie).

Deze veranderingen brachten duidelijke veranderingen met betrekking tot de planning met zich mee. Naar aanleiding van deze veranderingen is besloten dat de invoering zou worden uitgesteld tot de week na de afstudeerperiode, wel is er nog getest binnen de afstudeerperiode. Meer hiervoor valt te lezen in hoofdstuk 6: invoering.

Aan het eind van de tweede iteratie is er wederom een workshop gehouden, te weten workshop twee. Deze workshop is gehouden om zo de laatste op en aanmerkingen, wensen en eisen boven water te halen als input voor de derde en laatste iteratie, deze workshop is te vinden in de externe bijlage: workshop twee.

De rest van dit verslag zal ingaan op de laatste versies van het ontwerp van de verschillende pilots dus na workshop één en twee. Daarnaast zal de bouw van de verschillende pilots worden behandeld.

Veruit het belangrijkste onderdeel van het rapport pilotontwikkelplan planningssysteem is de functionele structuur van de pilot, deze zal dan ook verder behandeld worden.

### 5.2.1 Functionele structuur pilot planningssysteem

Voor de functionele structuur van pilot heb ik gekozen om de volgende onderdelen te schrijven:

- Gebruikte use-cases uit de definitiestudie;
- Menustructuur;
- Schetsen van de schermen;
- Navigatieschema.

De use-cases zijn in dit rapport naar voren gehaald om de koppeling met de functionaliteiten van deze pilotontwikkeling naar voren te brengen (welke functionaliteiten omvat deze pilot). De menustructuur laat de structuur en hiërarchie van het planningssysteem zien. De schetsen van de schermen (plus de bijbehorende uitleg) dienden voor de technische en functionele aspecten en lieten tevens de interactie tussen de gebruiker en het systeem zien. Tot slot liet het navigatieschema de manier van navigeren door de pilot zien.

Aangezien de gebruikte use-cases een simpele opsomming was welke te vinden is in de externe bijlage (Rapport pilotontwikkeling planningssysteem) en de schetsen van de schermen met de daarbij behorende uitleg de belangrijkste leidraad tijdens de ontwikkeling van deze pilot zijn geweest zal ik in deze paragraaf direct verder worden gegaan met het behandelen van de schetsen van de schermen.

#### Schetsen van de schermen

In deze subparagraaf behandel ik de schetsen van de schermen ( plus de bijbehorende toelichting) Hier zal duidelijk worden hoe de schetsen van de schermen de use-cases hebben kunnen ondervangen.



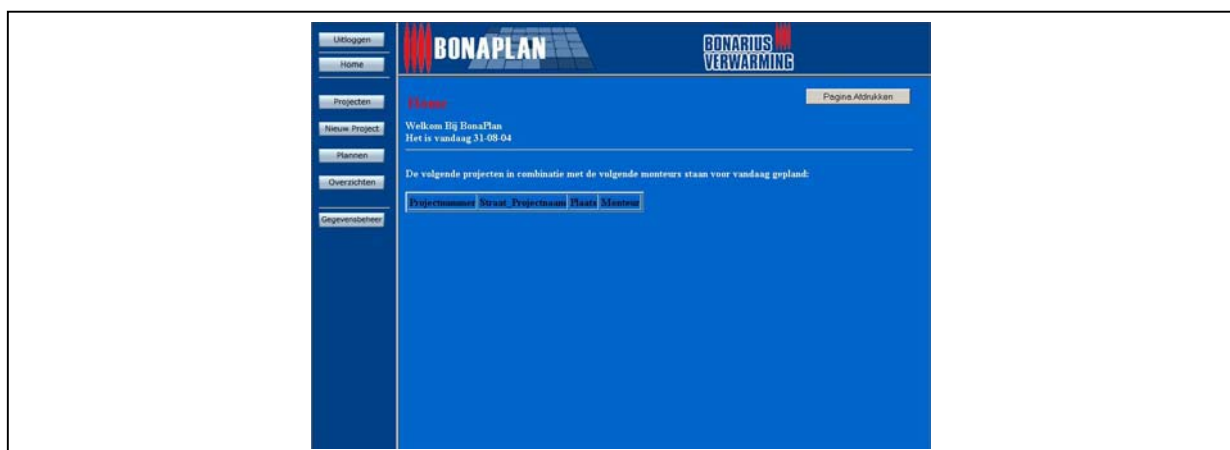
De pilot bestond uit de volgende pilotdelen, de pilotdelen zijn tevens de functionaliteiten die deze pilot bevat:

1. Inloggen/uitloggen: dit deel betreft de functionaliteiten voor de inlog en uitlog functionaliteiten van het systeem;
2. Project toevoegen: dit deel betreft de functionaliteiten voor het toevoegen van een project aan het systeem;
3. Project wijzigen: dit deel betreft de functionaliteiten voor het wijzigen van een project binnen het systeem;
4. Project verwijderen: dit deel betreft de functionaliteiten voor het verwijderen van een project uit het systeem;
5. Gegevens toewijzen aan een project: dit betreft de functionaliteiten voor het toewijzen van gegevens aan een project (programmeer technisch betekent dit dat alle listboxen dynamisch moeten worden gemaakt in de schermen project toevoegen & project wijzigen);
6. Project globaal bekijken: dit betreft de functionaliteiten voor het bekijken van een overzicht met globale project gegevens;
7. Project detail bekijken: dit betreft de functionaliteiten voor het bekijken van de detail gegevens van een project;
8. Monteur en project plannen: dit deel betreft de functionaliteiten voor het plannen van projecten en project met monteurs;
9. Gepland project verwijderen: dit deel betreft de functionaliteiten voor het verwijderen van een gepland project;
10. Monteur van gepland project per dag verwijderen: dit deel betreft de functionaliteiten voor het verwijderen van een monteur per project per dag;
11. Monteur van gepland project verwijderen: dit deel betreft de functionaliteiten voor het verwijderen van een monteur van een project;
12. Geplande monteur vervangen voor andere monteur: dit deel betreft de functionaliteiten voor het vervangen van een monteur op een gepland project.

De pilot is in delen opgedeeld omdat de gehele pilot nog steeds te groot is om in één keer te ontwikkelen. De delen zijn zo gekozen dat ze één functionaliteit of een groep van verwante functionaliteiten omvatten.

Eerst zal de basis van planningssysteem worden getoond. Daarna zullen de functionaliteiten inloggen, project toevoegen, en project plannen van het planningssysteem worden getoond. Dit zal worden gedaan middels een schets van het scherm, een screenshot van het uiteindelijke resultaat en een impressie van de broncode, tevens zullen hierbij de keuzes en eventuele problemen en oplossingen beschreven worden welke naar voren zijn gekomen tijdens de bouw van deze functionaliteiten.

Het planningssysteem is opgebouwd uit een frameset welke hieronder te zien is.



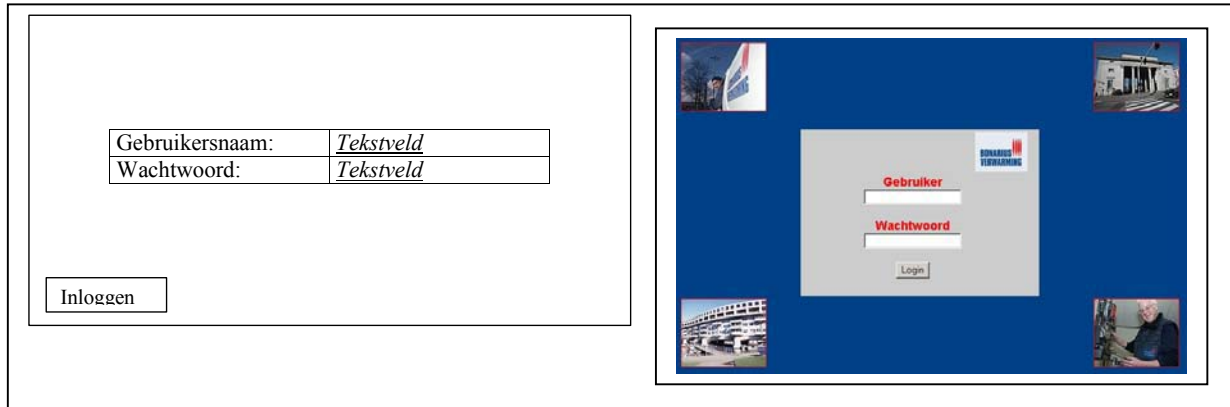
**Figuur 5.8: Frameset BonaPlan**

Het voordeel van het gebruik van een frameset was dat het door het hele programma mogelijk blijft om naar vast te punten te kunnen navigeren. Zo blijft het overzicht binnen het planningssysteem behouden. De vaste punten binnen het systeem waren: projecten, nieuw project, plannen, overzichten en gegevens beheer. Dit alles zijn de belangrijkste zaken van het programma BonaPlan. Tijdens deze

pilot hield ik me bezig met het plannen en de projecten, deze zijn tenslotte benodigd voordat er gepland kan worden.

Het gehele systeem BonaPlan en dus ook het planningssysteem bevindt zich achter een inlogscherm. Middels een gebruikersnaam en een wachtwoord wordt door het systeem bepaald of de gebruiker toegang heeft tot het systeem en wat zijn rechten zijn (dit gebeurt aan de hand van het attribuut user\_soort)

De schets voor dit scherm en het uiteindelijke resultaat was als volgt.



**Figuur 5.9: Schermschets en resultaat inloggen**

Het uiteindelijke scherm is gemaakt met behulp van Dreamweaver. Door met Dreamweaver de lay-out van dit scherm te ontwikkelen was dit erg snel gedaan en moest alleen nog de functionaliteit voor het inloggen aan dit scherm te worden toegevoegd

Wanneer men op de knop inloggen drukt zal er doormiddel van een script door het systeem de gebruikersnaam en wachtwoord worden gecontroleerd en de juiste rechten aan de gebruiker worden toegekend. Een fragment van de achterliggende code vindt u in figuur 5.10.

```
// check of de user op de 'login' knop heeft gedrukt en of $loginnaam en $paswoord niet leeg zijn
if ($HTTP_POST_VARS["loginnaam"] && $HTTP_POST_VARS["paswoord"])
{
    /* haal login gegevens uit database en check de
    gegevens met de ingevoerde gegevens */

    $sql_db = "SELECT Systeem_gebruiker_id, Username, Paswoord, User_soort_id FROM $table WHERE Username =
'$HTTP_POST_VARS[loginnaam]' AND Paswoord = '$HTTP_POST_VARS[paswoord]'";
    $res_db = mysql_query($sql_db);

    // check of er een overeenkomst is met ingevoerde en uitgelezen bestanden
    if (mysql_num_rows($res_db) >= 1)
    {
        // zet de values van database in een array
        $row = mysql_fetch_array($res_db);

        // zet de sessions
        $sess_id = $row['Systeem_gebruiker_id'];
        $sess_loginnaam = $row['Username'];
        $user_soort = $row['User_soort_id'];

        // registreer ze
        session_register('sess_id');
```

**Figuur 5.10: Fragment code inlogfunctie**

In deze code worden de velden gebruikersnaam en wachtwoord uitgelezen en vergeleken met de waarde uit de database. Kloppen deze dan gaat de rest van het script zijn werk doen om vervolgens als gebruiker zijnde op de juiste pagina met de juiste rechten BonaPlan binnen te komen.

Een andere functionaliteit van het planningssysteem is het toevoegen van een project. De moeilijkheid hiervan was dat het project verschillende gegevens bevat uit andere tabellen (vandaar dat de pilot 3

parallel met deze pilot is ontwikkeld). Dit is dan ook een lastig punt geweest tijdens het realiseren van deze functionaliteit. Bijkomend voordeel was dat het principe achter deze functionaliteit ook gebruikt werd tijdens de pilot gegevensbeheer, daardoor kon deze pilot sneller ontwikkeld worden. Hieronder ziet u de schets van een project toevoegen.

Projectnummer:	<u>Tekstveld</u>	Opmerkingen:	<u>Tekstvak</u>
Werknummer klant:	<u>Tekstveld</u>		
Straat / projectnaam:	<u>Tekstveld</u>	Opdrachtgever:	<u>Listbox</u>
Huisnummer	<u>Tekstveld</u>	Ketel:	<u>Listbox</u>
Huisnummer toevoeging:	<u>Tekstveld</u>	Urgentie:	<u>Listbox</u>
Plaats:	<u>Tekstveld</u>	Uitvoerder:	<u>Listbox</u>
Telefoonnummer bewoner:	<u>Tekstveld</u>	Radiator:	<u>Listbox</u>
Contactpersoon opdrachtgever:	<u>Tekstveld</u>	Werktype:	<u>Listbox</u>
Complexnummer:	<u>Tekstveld</u>	Projectleider:	<u>Listbox</u>
Aantal woningen:	<u>Tekstveld</u>	Status:	<u>Listbox</u>
Offerte uitbrengen:	<u>Listbox</u>		
Opname nodig:	<u>Listbox</u>		
Bouw nodig:	<u>Listbox</u>		
Elektra aanbrengen:	<u>Listbox</u>		

toevoegen

Figuur 5.11: Schermschets project toevoegen

Het uiteindelijk ontwikkelde scherm ziet er als volgt uit:

**Figuur 5.12: Screenshot project toevoegen**

De gegevens welke uit een andere tabel (dan de project tabel) worden toegevoegd zijn de listboxen welke zich bevinden aan de linker kant van het scherm. Deze listboxen worden dus gevuld aan de hand van de inhoud van de database (dynamische listboxen).

Om de informatie vanuit een andere tabel te kunnen toevoegen aan een project heb ik gebruik, gemaakt van de PHP functie `mysql_fetch_array()` en een SQL query, deze zijn hieronder te zien

```
//ketel
$sql2 = "SELECT Ketel_id, Ketel_naam FROM Ketel ORDER BY Ketel_id";
$result2 = mysql_query($sql2);

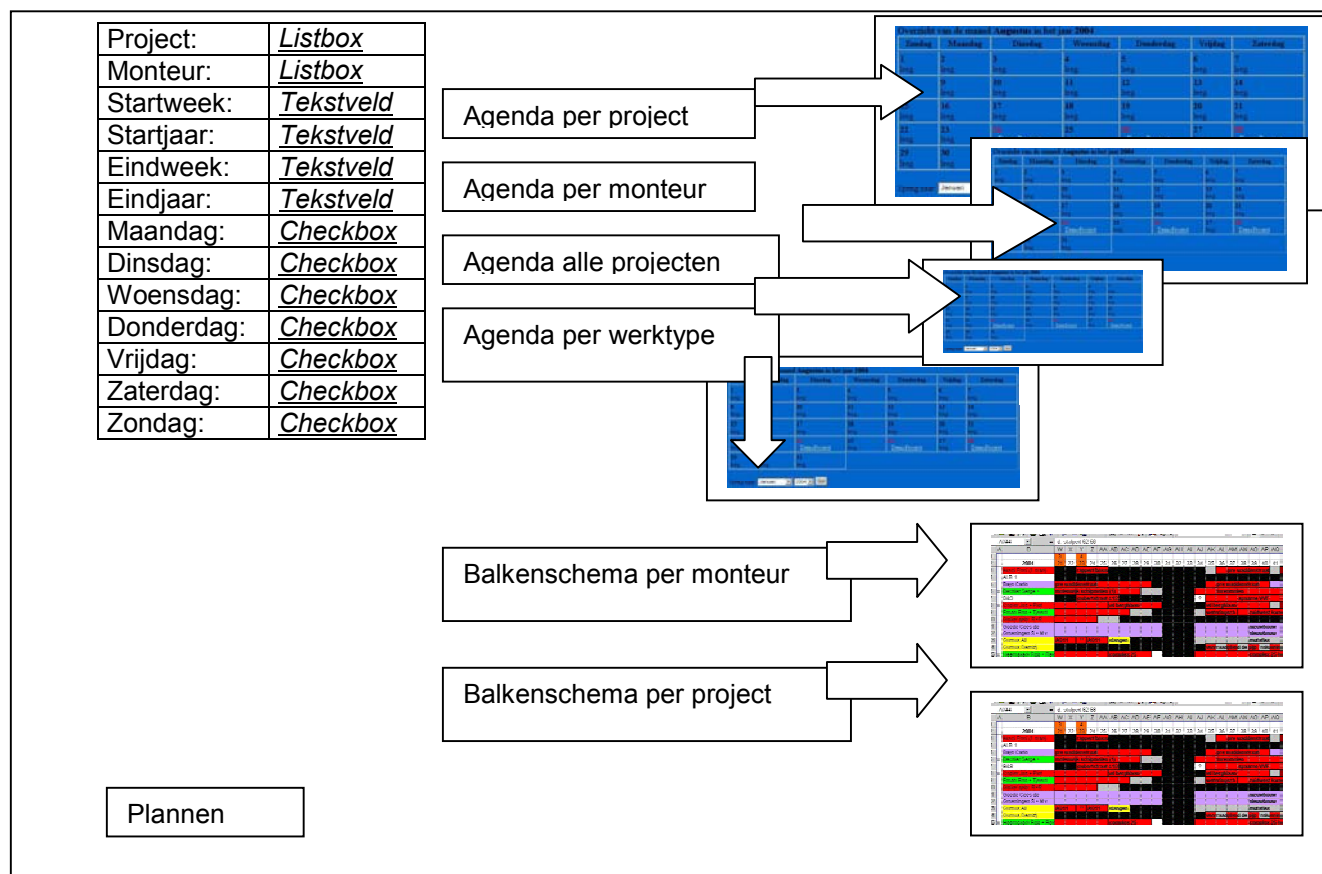
-----
echo "      <td><select name=\"ketel\">;
              while($ketel = mysql_fetch_array($result2))
              {
                  print("<OPTION VALUE=\"\$ketel[0]\">\$ketel[1]\n");
              }
echo "      </select></td>";
```

**Figuur 5.13: Fragment code project toevoegen**

`mysql_fetch_array()` is een bestaande functie binnen PHP en zorgt ervoor dat alle uitkomsten van de query (in dit geval de query `sql2`) in een array worden geplaatst. Vervolgens zolang er nog waarden in de array zitten wordt de waarde `<OPTION VALUE=\"\$ketel[0]\">\$ketel[1]\n`; geprint. Dit is de HTML code voor een listbox. Middels deze code word de listbox in "project toevoegen" dus gevuld met de gegevens die in de tabel ketel staan.

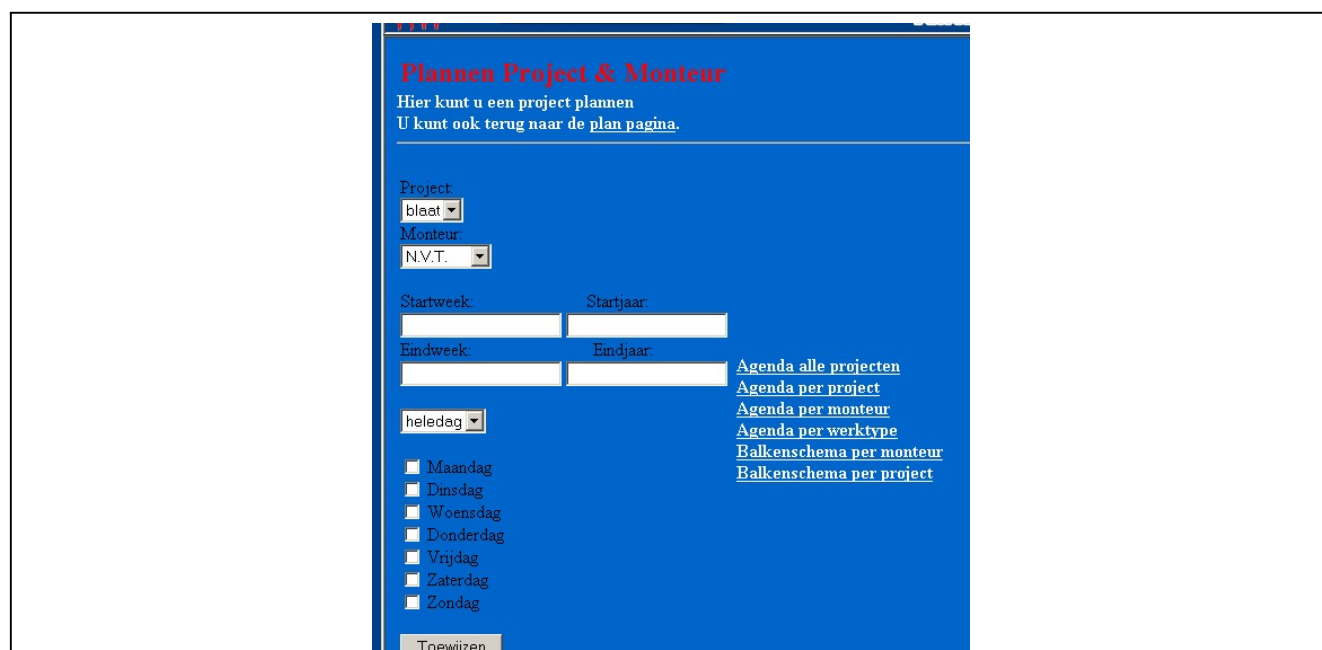
Aangezien dit principe door het gehele systeem gebruikt word was dit een belangrijke stap tijdens de ontwikkeling van BonaPlan. Doordat ik deze functionaliteit al vroeg heb kunnen programmeren heeft mij dit veel tijd bespaard tijdens de verdere ontwikkeling van BonaPlan.

Nadat de functionaliteit project toevoegen in de pilot planningssysteem was toegevoegd, kon er verder gegaan worden met de plan functionaliteit. Voor deze functionaliteit is in navolging van de use-case een schermshots gemaakt, deze zag er als volgt uit.



Figuur 5.14: Schermschets plan functionaliteit

Het uiteindelijk ontwikkelde scherm ziet er als volgt uit:



Figuur 5.15: Screenshot plan functionaliteit

Het ontwerpen van deze functionaliteit was een tijdrovende en lastige klus. Dit had te maken met het feit dat er gepland dient te worden met weeknummers en dat er gepland moet kunnen worden voor een periode (van week .. tot week..). Tijdens mijn informatiewinning naar weeknummer en het plannen hiervan kwam ik erachter dat er een ISO-8601 norm bestaat voor notatie van data en tijd. De definitie van deze norm is als volgt:

*International Standard ISO 8601 specifies numeric representations of date and time. This standard notation helps to avoid confusion in international communication caused by the many different national notations and increases the portability of computer user interfaces. In addition, these formats have several important advantages for computer usage compared to other traditional date and time notations. The time notation described here is already the de-facto standard in almost all countries and the date notation is becoming increasingly popular.*

In deze ISO norm komt ondermeer naar voren hoe het weeknummer bepaald kan worden. Middels een script was het mogelijk om via een weeknummer en het jaar van dat weeknummer de datum van de eerste maandag van die week te bereken. Dit script zag er als volgt uit:

```
function weektoday($week,$year){
    $day = strtotime($year."-01-01");
    $dayofweek = date("w",$day);
    if($dayofweek != 5 && $dayofweek != 6 && $dayofweek != 0){
        while(1 != date("w",$day)){
            $day = mktime(0,0,0,date("m", $day),date("d", $day)-1,date("Y", $day));
        }
    } else{
        while(1 != date("w",$day)){
            $day = mktime(0,0,0,date("m", $day),date("d", $day)+1,date("Y", $day));
        }
    }
    $day = mktime(0,0,0,date("m", $day),date("d", $day)+7*$week-7,date("Y", $day));
    return $day;
}
```

**Figuur 5.16: Script voor omreken weeknummers**

Het enig nadeel was dat de uitkomst van deze functie het Unix formaat van die eerste maandag van dat weeknummer weergeeft. Wat is nu het Unix formaat? Het Unix formaat is het aantal seconden dat verstreken is sinds 1 januari 1970. Deze tijd diende dan ook weer omgezet te worden naar een tijd met een jaar-maand-dag. Dit kan middels de functies:

- \$yy = date("Y", \$Unixtime) : Dit levert het bijbehorende jaar van de Unix-time;
- \$mm = date("m", \$Unixtime) : Dit levert de bijbehorende maand van de Unix-time;
- \$dd = date("d", \$Unixtime) : Dit levert de bijbehorende dag van de Unix-time;

Na dit alles uitgezocht en geschreven te hebben kon de planfunctie geschreven worden. Een fragment uit de code, welke met behulp van een PHP editor is geschreven, van de planfunctie is te zien in figuur 5.17.

```
case "2": //eerstemaandag + 1 dag = dus dinsdag in unix tijd + 86400
    $EersteDinsdag = $EersteMaandag + 86400;
    $yy = date("Y", $EersteDinsdag);
    $mm = date("m", $EersteDinsdag);
    $dd = date("d", $EersteDinsdag);
    $datum = "$yy$mm$dd";

    $sql = "INSERT INTO `werkt_aan` ('ID', `datum`, `Projectnummer`, `Monteur_id`)
VALUES ('', '$datum', '$projectnummer', '$monteur')";
    $result = mysql_query($sql);

    $i=0;
    While($AantalWeken !=$i){
        $EersteDinsdag2 = $EersteDinsdag + ((1 + $i) * 604800);
        $yy = date("Y", $EersteDinsdag2);
        $mm = date("m", $EersteDinsdag2);
        $dd = date("d", $EersteDinsdag2);
        $datum2 = "$yy$mm$dd";

        $sql = "INSERT INTO `werkt_aan` ('ID', `datum`, `Projectnummer`, `Monteur_id`)
VALUES ('', '$datum2', '$projectnummer', '$monteur')";
        $result = mysql_query($sql);
        $i++;
    }

    break;
```

**Figuur 5.17: Script van plan functie**

De uitleg van de bovenstaande code zal middels een voorbeeld worden uitgelegd. Als voorbeeld gaan we er vanuit dat de gebruiker een project alleen op dinsdag wil inplannen voor een duur van twee weken. Wanneer de gebruiker op de knop plannen drukt, rekent het systeem aan de hand van de startweek + startjaar de eerste maandag van deze week uit in Unix formaat tevens berekent het systeem het aantal weken dat het project dient te duren. Daarna komt het bovenstaande fragment om de hoek kijken. Eerst wordt de Unix-tijd van de eerste dinsdag bepaald. Dit is dus de Unix-tijd die berekend is + het aantal seconden van 1dag, dit is dus eerstemaandag + (60\*60\*24 = 86400) die datum wordt omgezet in een jaar, maand en dag. Deze word daarna weggeschreven in de database tezamen met het projectnummer en monteur id. Wanneer het project over meerdere weken gepland wordt, wordt de eerste dinsdag Unix-time aangevuld met het aantal seconden van een week 604800 (86400 \* 7) en schrijft het systeem deze datum weer weg naar de database.

Het tweede onderdeel van de functionaliteit plannen is het kunnen bekijken van de geplande gegevens in een agenda. In de schermshots van de functionaliteit plannen viel al eerder dit scherm te zien.

Overzicht van de maand Augustus in het jaar 2004						
Zondag	Maandag	Dinsdag	Woensdag	Donderdag	Vrijdag	Zaterdag
1 leeg	2 leeg	3 leeg	4 leeg	5 leeg	6 leeg	7 leeg
8 leeg	9 leeg	10 leeg	11 leeg	12 leeg	13 leeg	14 leeg
15 leeg	16 leeg	17 leeg	18 leeg	19 leeg	20 leeg	21 leeg
22 leeg	23 leeg	24 DemoProject	25 leeg	26 DemoProject	27 leeg	28 DemoProject
29 leeg	30 leeg	31 leeg				

Spring naar:

**Figuur 5.18: De Agenda**



Dit scherm is in verschillende variaties aan te roepen:

- Agenda per monteur;
- Agenda per project;
- Agenda alle projecten;
- Agenda per werkttype.

Een fragment uit de achterliggende code volgt hieronder.

```
<?php
function gepland_of_niet($counter,$datum)
{
    global $xtra;

    $query = mysql_query("SELECT * FROM werkt_aan WHERE datum='$datum'");

    $aantal = mysql_num_rows($query);
    if ($aantal==0)
    {
        $stuff = "<td valign=top $xtra><B>$counter</B>.<BR>leeg.</td>";
        return $stuff;
    }
    else
    {
        $stuff = "<td valign=top bgcolor=#0066CC $xtra><B><font color=red>$counter</B>.";
        while ($row = mysql_fetch_object($query))
        {
            $Projectnummer = $row->Projectnummer;
            if (strlen($Projectnummer) > 15)
            {
                $Projectnummer = substr($Projectnummer,0,15)."...";
            }
            $sql8 = mysql_query("SELECT Monteur_naam FROM monteur WHERE Monteur_id=$row->Monteur_id");
            $fetch8 = mysql_fetch_assoc($sql8);
            $stuff .= "<BR><font color='\"#FFFFFF\"'>$Projectnummer, $fetch8[Monteur_naam]</font></a>";
            //
        }

        $stuff .= "</font></td>";
        return $stuff;
        //
    }
}
```

**Figuur 5.19: Code van agenda**

Deze code is wederom geschreven met behulp van een PHP editor. Het bovenstaande schema zorgt ervoor dat de afspraken uit de database worden opgehaald en worden getoond in de agenda. De agenda moet in verschillende varianten kunnen worden aangeroepen. Dit valt te doen door simpelweg de query aan te passen binnen dit script (\$query = mysql\_query("SELECT \* FROM werkt\_aan WHERE datum='\$datum'")).

## 5.2.2 Organisatorische inrichting pilot planningssysteem

In de organisatorische inrichting staat de huidige manier van plannen, de organisatorische gevolgen, het globale opleidingsplan en de specificatie van de gebruikersdocumentatie opgenomen. Deze zijn afgeleid van het analyse gedeelte welke zich binnen de definitiestudie bevindt. De organisatorische gevolgen voor Bonarius zijn voornamelijk dat de manier van plannen zal veranderen. Planningen zullen niet meer op papier moeten worden gedaan maar direct in BonaPlan, dit om zo het planningssysteem consistent te houden aan de werkelijkheid. De opleidingen van deze pilot zal bestaan uit een demonstratie van ongeveer een half uur, daarnaast zal er een handleiding geschreven worden voor dit systeem. Aangezien ik nog enkele maanden blijf na de afstudeerperiode zal er tevens in geval van nood beroep op mij gedaan kunnen worden in deze maanden. Voor informatie over de organisatorische inrichting verwijs ik u naar de externe bijlage (rapport pilotontwikkelplan planningssysteem).



### 5.2.3 Pilotontwikkelplan pilot planningssysteem

Nadat het plan van aanpak, de functionele structuur en de organisatorische inrichting waren beschreven kon er begonnen worden met het pilotontwikkelplan. In dit onderdeel heb ik ervoor gekozen om het volgende te beschrijven:

- Specificatie van de pilotdelen;
- Planning van de pilotdelen;
- Testprocedure;
- Invoering.

De belangrijkste onderdelen van het pilotplan waren de specificatie van de pilotdelen en de planning hiervan. Aangezien de gehele pilot te groot was om in één keer te ontwikkelen is de pilot in delen opgedeeld, deze delen zijn zo gekozen dat ze één functionaliteit of groep van samenhangende functionaliteiten ondervangen.

Aan het begin van het rapport viel al een planning te zien van deze pilot (ontwikkeling en bouw) het aantal dagen voor de bouw van deze pilot is in het pilotplan opgedeeld over de verschillende pilotdelen en deze zijn in een zogenaamde time-box gezet.

Naam pilotdeel	Benodigde tijd	Prioriteit
Inloggen/uitloggen	1 dag	Basis
Project toevoegen	1 dag	Basis
Project wijzigen	1 dag	Basis
Project verwijderen	1 dag	Basis
Gegevens toewijzen aan een project	1 dag	Basis
Project globaal bekijken	½ dag	Basis
Project detail bekijken	½ dag	Basis
Monteur en project plannen	4 dagen	Basis
Gepland project verwijderen	1 dag	Basis
Monteur van gepland project per dag verwijderen	1 dag	Basis
Monteur van gepland project verwijderen	1 dag	Basis
Agenda alle projecten	4 dagen	Basis
Agenda per monteur	½ dag	Basis
Agenda per project	½ dag	Basis
Agenda per werktipe	1 dag	Basis
Geplande monteur vervangen voor andere monteur	2 dagen	Comfort

**Figuur 5.20: Time-box planningssysteem**

Voor verder informatie over het pilotontwikkelplan verwijs ik u naar de externe bijlagen (rapport pilotontwikkelplan planningssysteem).

### 5.2.4 Verificatie en validatie pilot planningssysteem

In deze paragraaf behandel ik de verificatie en validatie van de pilot planningssysteem. Dit is gedaan door de use-cases tegenover de functionele systeemeisen uitgezet, hieronder een kort fragment:

	1	2	3	4
		Planner	Opdrachtgever	Viewer
				Beleider
U1				
U2				
U3				
U4				
U5				
U36				
U37				
U38				
U39				
U40				
U41				
U42				
U43				
U44				
U45				
U46				
U47				
U48				
U49				
U50				

**Figuur 5.21: Verificatie en validatie**

Op deze manier heb ik samen met de opdrachtgever kunnen verifiëren/valideren dat alle systeemeisen zijn ondervangen door de use-cases en daardoor uiteindelijk alle functionaliteiten van het systeem ontwikkeld zouden worden. Daarnaast zijn de schermsschetsen uitgezet tegenover de use-cases.

	1	2	3	4	5	6	7	8	9	10	11
U1											
U2											
U3											
U4											
U5											
U36											
U37											
U38											
U39											
U40											
U41											
U42											
U43											
U44											
U45											
U46											
U47											
U48											
U49											
U50											

*N.b. in de rijen staan de nummers van de use cases en in de kolommen de nummers van de schermsschetsen.*

**Figuur 5.22: Verificatie en validatie**

Op deze manier heb ik samen met de opdrachtgever kunnen verifiëren/valideren dat de schermsschetsen alle use-cases ondervangen en zo dus tijdens deze pilot alle functionaliteiten zouden worden ontwikkeld.

Kortom kan gezegd worden dat deze pilot een groot onderdeel van het programma BonaPlan was. Daarnaast dient in gedachten gehouden te worden dat deze pilot parallel ontwikkeld is aan pilot drie: gegevensbeheer, welke in de volgende paragraaf behandeld zal worden.

### 5.3 Pilot 3: Gegevensbeheer

Tijdens deze pilot is het gegevensbeheer gedeelte van BonaPlan gebouwd. Het gegevensbeheer gedeelte moest aan verschillende eisen voldoen welke zijn terug te vinden in de systeemeisen in de definitiestudie. Gegevensbeheer is de ondersteuning en aanvulling op het belangrijkste onderdeel van BonaPlan: het planningssysteem. Gegevensbeheer omvat het complete beheer van de volgende gegevens:

- Werktype;
- Status;
- Radiator;
- Projectleider;
- Uitvoerder;
- Urgentie;
- Ketel;
- Opdrachtgever;
- Monteur;
- Systeemgebruiker (deze kan alleen door de Beheerder beheert worden).

Onder beheer wordt verstaan, het toevoegen, wijzigen en verwijderen van de gegevens, tevens is er in deze pilot een totaal overzicht per gegeven opgenomen.

In deze pilot zijn de volgende onderdelen gerealiseerd:

- Rapport pilotontwikkelplan gegevensbeheer;
  - Plan van aanpak;
  - Functionele structuur;
  - Organisatorische inrichting;
  - Pilot ontwikkelplan;
  - Verificatie en validatie.
- Pilot drie: Gegevensbeheer.

Als basis en ter ondersteuning van de bouw van het gegevensbeheer is er een rapport pilotontwikkelplan opgesteld. In dit rapport is het gegevensbeheer ontworpen. Aan de hand van dit rapport is vervolgens deze pilot gebouwd.

Aan het begin van deze pilot is er begonnen met een plan van aanpak. In het plan van aanpak zijn achtereenvolgens de reikwijdte, de resultaten en de planning beschreven. De planning was het belangrijkste onderdeel van het plan van aanpak, door de eerder beschreven manier van plannen, moest via deze manier de overkoepelende planning verder gedetailleerd worden.

De organisatorische inrichting, pilot ontwikkelplan en de verificatie en validatie van deze pilot zullen niet besproken worden. Hier is voor gekozen omdat deze dezelfde informatie en/of op dezelfde manier zijn uitgevoerd zoals in pilot twee: planningssysteem is beschreven.

Daardoor zal er nu direct verder gegaan worden met veruit het belangrijkste onderdeel van het rapport pilotontwikkelplan gegevensbeheer: de functionele structuur van de pilot.

#### 5.3.1 Functionele structuur pilot gegevensbeheer

Voor de functionele structuur van pilot heb ik gekozen om de volgende onderdelen te schrijven:

- Gebruikte use-cases uit de definitiestudie;
- Menustructuur;
- Schetsen van de schermen;
- Navigatieschema.

De use-cases zijn in dit rapport naar voren gehaald om zo de koppeling tussen de definitiestudie en de pilotontwikkeling duidelijk te houden. De menustructuur laat de structuur en hiërarchie van het gegevensbeheer zien. De schetsen van de schermen (+ de bijbehorende uitleg) dienen voor de technische en functionele aspecten en laten tevens de interactie tussen de gebruiker en het systeem zien. Tot slot laat het navigatieschema de manier van navigeren door de pilot zien.

Aangezien de gebruikte use-cases een simpele opsomming is welke te vinden is in de externe bijlage (Rapport pilotontwikkeling gegevensbeheer) en de schetsen van de schermen met de daarbij behorende uitleg de belangrijkste leidraad tijdens de ontwikkeling van deze pilot zijn geweest zal ik in deze paragraaf direct verder gaan met de schetsen van de schermen.

### **Schetsen van de schermen**

In deze sub paragraaf behandel ik de schetsen van de schermen (plus de bijbehorende toelichting) hier zal duidelijk worden hoe de schetsen van de schermen de use-cases kunnen ondervangen.

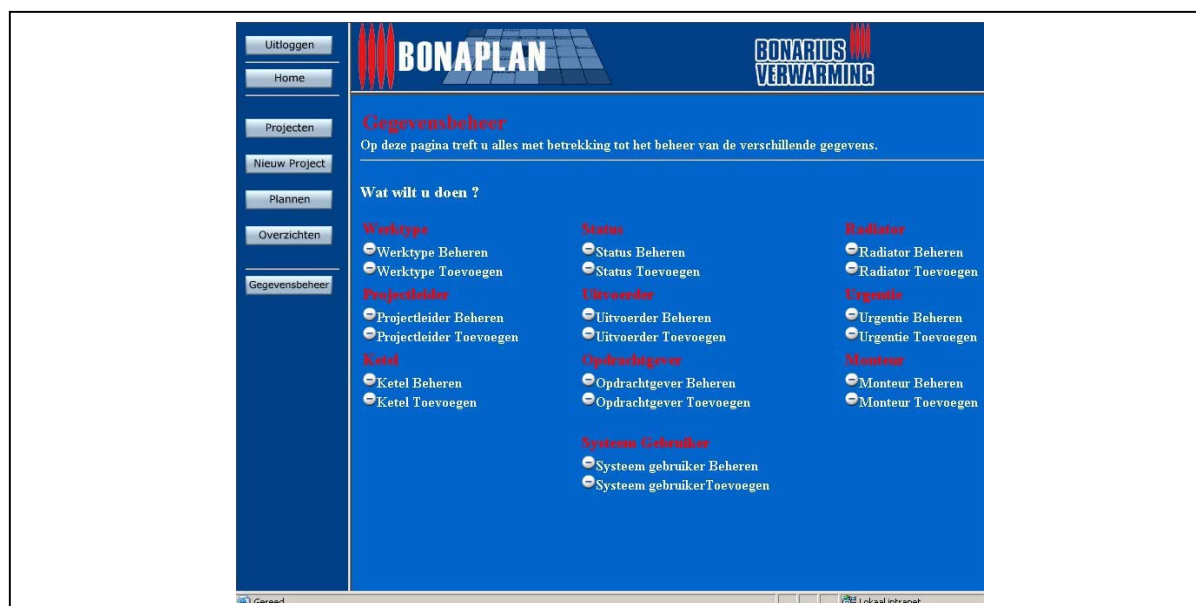
De pilot gegevensbeheer bestond uit de volgende pilotdelen:

1. Ketel beheer: in dit onderdeel kan binnen het systeem een nieuwe ketel toegevoegd worden en een bestaande gewijzigd of verwijderd worden. Tevens kan er een globaal overzicht van alle ketels worden bekeken;
2. Werktype beheer: in dit onderdeel kan binnen het systeem een nieuw werkttype toegevoegd worden en een bestaande gewijzigd of verwijderd worden. Tevens kan er een globaal overzicht van alle werktypes worden bekeken;
3. Status beheer: in dit onderdeel kan binnen het systeem een nieuwe status toegevoegd worden en een bestaande gewijzigd of verwijderd worden. Tevens kan er een globaal overzicht van alle statussen worden bekeken;
4. Opdrachtgever beheer: in dit onderdeel kan binnen het systeem een nieuwe Opdrachtgever toegevoegd worden en een bestaande gewijzigd of verwijderd worden. Tevens kan er een globaal overzicht van alle opdrachtgevers worden bekeken;
5. Radiator beheer: in dit onderdeel kan binnen het systeem een nieuwe radiator toegevoegd worden en een bestaande gewijzigd of verwijderd worden. Tevens kan er een globaal overzicht van alle radiatoren worden bekeken;
6. Projectleider beheer: in dit onderdeel kan binnen het systeem een nieuwe projectleider toegevoegd worden en een bestaande gewijzigd of verwijderd worden. Tevens kan er een globaal overzicht van alle projectleiders worden bekeken;
7. Uitvoerder beheer: in dit onderdeel kan binnen het systeem een nieuwe uitvoerder toegevoegd worden en een bestaande gewijzigd of verwijderd worden. Tevens kan er een globaal overzicht van alle uitvoerders worden bekeken;
8. Urgentie beheer: in dit onderdeel kan binnen het systeem een nieuwe urgentie toegevoegd worden en een bestaande gewijzigd of verwijderd worden. Tevens kan er een globaal overzicht van alle urgenties worden bekeken;
9. Systeem\_gebruiker beheer: in dit onderdeel kan binnen het systeem een nieuwe Systeem gebruiker toegevoegd worden en een bestaande gewijzigd of verwijderd worden. Tevens kan er een globaal overzicht van alle systeem gebruikers worden bekeken.

De pilot is in delen opgedeeld omdat de gehele pilot nog steeds te groot is om in één keer te ontwikkelen. De delen zijn zo gekozen dat ze één functionaliteit of een groep van verwante functionaliteiten omvatten.

Ter toelichting van deze pilot zal de functionaliteit opdrachtgever beheer nader worden toegelicht. Dit zal worden gedaan middels een schets van het scherm, een screenshot van het uiteindelijke resultaat en een impressie van de broncode, tevens zullen hierbij de keuzes en eventuele problemen en oplossingen beschreven worden welke naar voren zijn gekomen tijdens de bouw van deze functionaliteit.

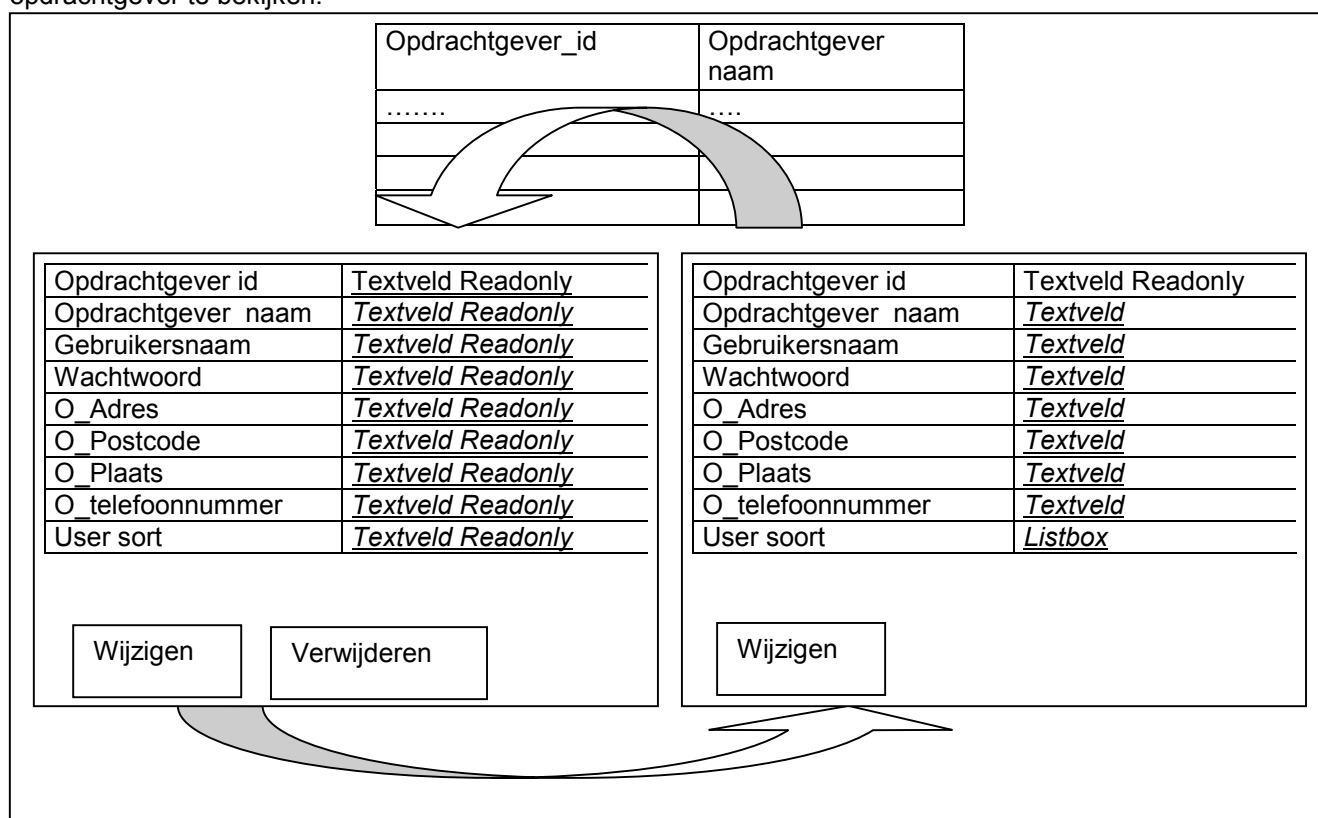
Het scherm van de pilot gegevensbeheer is van te voren niet geschetst. Dit scherm is toegevoegd om in één opslag te kunnen zien welke gegevens beheert kunnen worden. Dit scherm heeft dan ook niet meer functionaliteit dan een overzicht creëren en de mogelijk om vanuit daar de gegevens te gaan beheren, het scherm is er uiteindelijk als volgt uit komen te zien.



Figuur 5.23: Verzameling gegevensbeheer

Middels de schetsen van de schermen zal meteen de werking van het systeem worden uitgelegd. Deze schetsen van de schermen zijn ontwikkeld in samenwerking met de opdrachtgever en eindgebruiker en zijn gebaseerd op de systeemeisen en de use-cases. Deze schermsschetsen zijn gemaakt en aangepast tijdens de verschillende interviews en workshop één en twee.

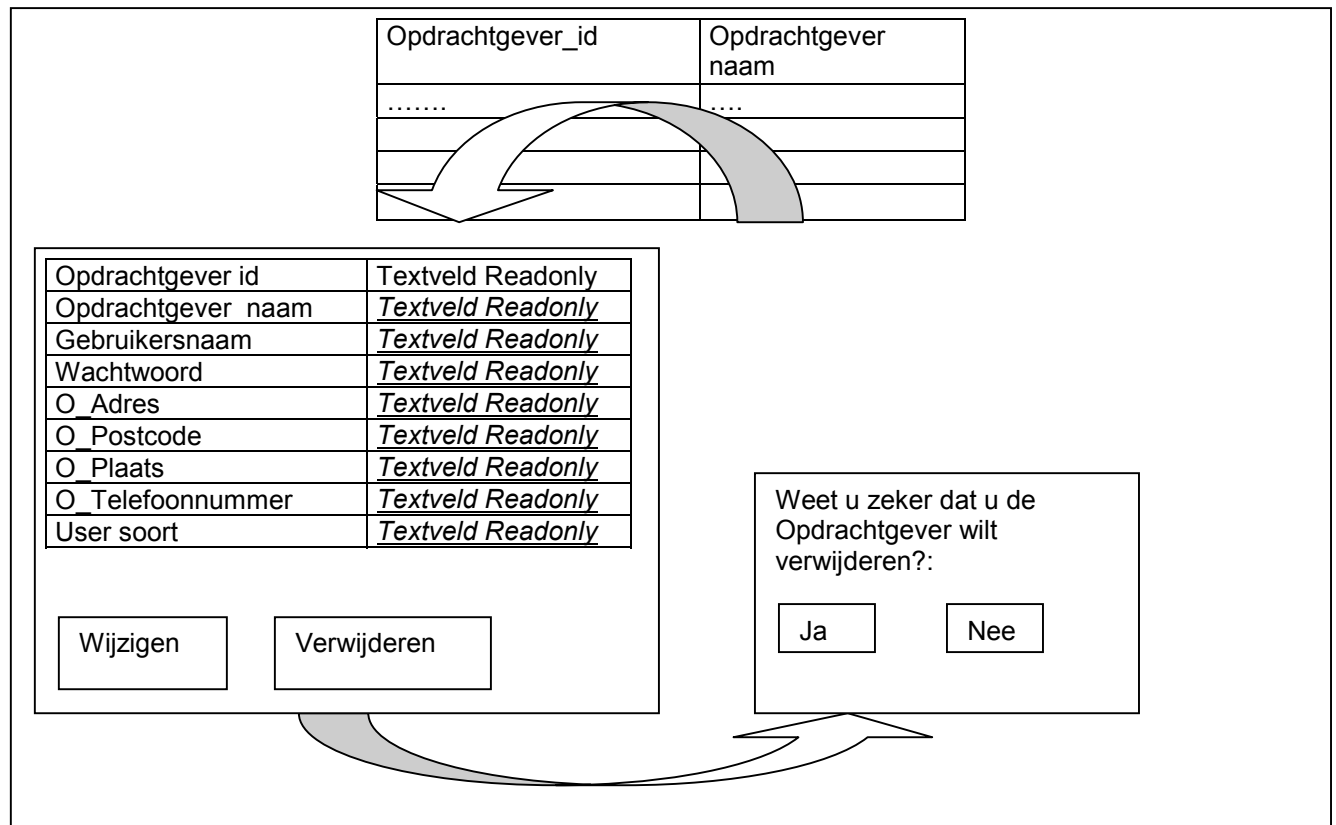
Wanneer de gebruiker kiest voor de opties opdrachtgever beheren zal hij uitkomen in een overzicht van alle opdrachtgevers. Het "Id" van de opdrachtgever dient als link om de detail gegevens van de opdrachtgever te bekijken.



Figuur 5.24: Schermsschets opdrachtgever wijzigen

Vanuit dit detailscherm zal de gebruiker kunnen kiezen tussen de opties wijzigen of verwijderen. Wanneer de gebruiker op de optie wijzigen klikt zal hij de mogelijkheid krijgen om de gegevens te wijzigen. Zijn de gegevens gewijzigd en drukt de gebruiker op de knop opslaan dan zal het systeem een melding geven dat de gegevens zijn opgeslagen. Lukt het niet dan zal het systeem daar ook een melding over geven, na het al dan wel of niet wijzigen van een opdrachtgever krijgt de gebruiker de mogelijkheid om terug te keren naar het overzicht van alle opdrachtgevers

Wanneer de gebruiker voor de op optie verwijderen, wordt de gebruiker om bevestiging gevraagd. Druk de gebruiker op "ja", dan zal de opdrachtgever uit het systeem verwijderd worden en keert de gebruiker terug in het overzicht van alle opdrachtgevers. Druk de gebruiker op "nee", dan zal hij terugkeren in het overzicht van alle opdrachtgevers.



**Figuur 5.25: Schermschets opdrachtgever verwijderen**

Wanneer de gebruiker kiest voor de opties toevoegen zal hij uitkomen in het scherm uit figuur 5.26 waar hij een opdrachtgever kan toevoegen. De gebruiker vult de benodigde gegevens in en drukt op de knop toevoegen. Het systeem zal een melding geven wanneer de opdrachtgever succesvol is opgeslagen. Lukt het niet dan zal het systeem daar ook een melding over geven, na het al dan wel of niet toevoegen van een opdrachtgever krijgt de gebruiker de mogelijkheid om terug te keren naar het overzicht van alle opdrachtgevers.

Opdrachtgever id	Textveld Readonly
Opdrachtgever naam	<u>Textveld</u>
Gebruikersnaam	<u>Textveld</u>
Wachtwoord	<u>Textveld</u>
O Adres	<u>Textveld</u>
O Postcode	<u>Textveld</u>
O Plaats	<u>Textveld</u>
O Telefoonnummer	<u>Textveld</u>
User soort	<u>Listbox</u>

Toevoegen

**Figuur 5.26: Schermschets opdrachtgever toevoegen**

Bij de ontwikkeling van de pilot ben ik op één structureel probleem gestuit. Dit probleem dook op wanneer er een gegeven gewijzigd diende te worden welke gegevens uit een andere tabel gebruikt (Foreign key). Dit probleem zal uitgelegd worden aan de hand van het wijzigen van een opdrachtgever.

Zoals eerder vermeld (Project toevoegen in pilot planningssysteem) word er in dit systeem gebruik gemaakt van dynamische listboxen. Wanneer de gebruiker een opdrachtgever wil wijzigen word er een scherm getoond waarin de gegevens gewijzigd kunnen worden, maar wanneer er zich in dit scherm een listbox bevind, onthoud deze niet de gewenste waarde. Wanneer een gebruiker dan alle gegevens wijzigt van een opdrachtgever, behalve de listbox (in het geval van opdracht gever user\_soort) en deze wijzigingen opslaat, dan wordt het attribuut uit de listbox gewijzigd in het eerste attribuut uit de listbox. In het geval van bijvoorbeeld de opdrachtgever kan het dan voorkomen dat de opdrachtgever de rechten van de beheerder krijgt. Dit mag dus absoluut niet gebeuren. Helaas heb ik dit probleem niet kunnen oplossen daarom heb ik voor een noodoplossing gekozen. Deze zal ik beschrijven aan de hand van screenshots uit het programma.

Wanneer de gebruiker op de knop opdrachtgever beheren drukt zal hij (zoals de eerdere schermschetsen tonen) in het scherm terecht komen waarin een lijst van alle opdrachtgevers met het ID en de naam van de opdrachtgever word getoond. Wanneer de gebruiker dan op het id van een opdrachtgever drukt komt hij uit in de detail gegevens van deze opdrachtgever. Hier kan de gebruiker kiezen voor de opties wijzigen. In het hierop volgende scherm kunnen de gegevens gewijzigd worden op het ene gegeven, user\_soort (de Foreign key), na. Wanneer de gebruiker nu de wijzigen zou opslaan blijft het user\_soort van de opdrachtgever gehandhaafd. Wil de gebruiker het user soort toch wijzigen dan kan dit gedaan worden door op de w (welke zich achter het user soort bevind) te drukken en daar het user soort te wijzigen.

**Opdrachtgever wijzigen**

Op deze pagina kunt u de gegevens wijzigen.  
U kunt ook terug naar de [opdrachtgever pagina](#).

Opdrachtgever naam	BAM Woningbouw	opslaan
Opdrachtgever username	BAM	
Opdrachtgever paswoord	*****	
Opdrachtgever Adres		
Opdrachtgever postcode		
Opdrachtgever plaats		
Opdrachtgever Tel		
User_soort	Opdrachtgever	<u>w</u>

**Figuur 5.27: Screenshot opdrachtgever wijzigen**



Door voor deze oplossing te kiezen werd een hoop leed voorkomen. Helaas bracht dit wel een extra paar klikken in het programma met zich mee. Daar kwam nog bij dat deze oplossing ook bij project wijzigen moest worden toegevoegd.

Nadat de pilot planningssysteem en de pilot gegevensbeheer waren afgerond was het tijd om naar de laatste pilot te gaan: Pilot Overzichten.

## 5.4 Pilot 4: Overzichten

Tijdens deze pilot zijn de overzichten welke binnen BonaPlan aanwezig zijn gebouwd. Zoals eerder gemeld zijn er binnen BonaPlan verschillende actoren (user soorten) gedefinieerd: Een beheerder, planner, viewer, en opdrachtgever. De pilot overzichten is in twee delen op te delen:

- Overzichten voor de viewer en planner;
- Overzichten voor de opdrachtgever.

De zogenoemde viewer heeft de rechten om alle overzichten binnen het systeem te bekijken. De viewer zal dan ook alleen personen binnen Bonarius mogen zijn. Daarnaast is er nog de opdrachtgever. De overzichten voor de opdrachtgever is een soort van extra gedeelte van het systeem. Dit gedeelte zorgt ervoor dat de volgende doelstelling uit de opdrachtschrijving (zie interne bijlage opdrachtschrijving) gerealiseerd wordt.

*Tevens dient het planningssysteem meer inzicht te geven aan opdrachtgevers (klanten) zodat zij ten alle tijden de status van een project kunnen bekijken. Dit is onderdeel van de verdere professionalisering van Bonarius Verwarming. Ook wil men middels deze functie van BonaPlan de service van Bonarius Verwarming proberen te ontlasten (minder status navragen).*

In deze pilot zijn de volgende onderdelen gerealiseerd:

- Rapport pilotontwikkelplan overzichten;
  - Plan van aanpak;
  - Functionele structuur;
  - Organisatorische inrichting;
  - Pilot ontwikkelplan;
  - Verificatie en validatie.
- Pilot vier: Overzichten.

Als basis en ter ondersteuning van de bouw van de overzichten is er een rapport pilotontwikkelplan opgesteld. In dit rapport zijn de overzichten ontworpen. Aan de hand van dit rapport is vervolgens deze pilot gebouwd.

Aan het begin van deze pilot is er begonnen met een plan van aanpak. In het plan van aanpak zijn achtereenvolgens de reikwijdte, de resultaten en de planning beschreven. De planning was het belangrijkste onderdeel van het plan van aanpak, door de eerder beschreven manier van plannen, moest via deze manier de overkoepelende planning verder gedetailleerd worden.

De organisatorische inrichting, pilot ontwikkelplan en de verificatie en validatie van deze pilot zullen niet besproken worden. Hier is voor gekozen omdat deze dezelfde informatie en/of op dezelfde manier zijn uitgevoerd zoals in pilot twee: planningssysteem is beschreven. Daardoor zal er nu direct verder gegaan worden met veruit het belangrijkste onderdeel van het rapport pilotontwikkelplan gegevensbeheer: de functionele structuur van de pilot.

In tegenstelling tot de vorige pilot heeft de pilot overzichten alleen de twee geplande iteraties doorlopen. De eerste iteratie was de opzet van de overzichten, de tweede iteratie betrof de wijzigingen die in workshop twee naar boven kwamen en het “stylen” van de pilot overzichten. Aangezien dat tijdens de workshops de ideeën over de verschillende overzichten die BonaPlan diende te bevatten steeds duidelijker werden, heeft deze pilot vooral in de tweede iteratie extra aandacht gekregen.

Veruit het belangrijkste onderdeel van het rapport pilotontwikkelplan overzichten is de functionele structuur van de pilot, deze zal dan ook verder behandeld worden.

### 5.4.1 Functionele structuur pilot gegevensbeheer

Voor de functionele structuur van pilot heb ik gekozen om de volgende onderdelen te schrijven:

- Gebruikte use-cases uit de definitiestudie;
- Menustructuur;
- Schetsen van de schermen;
- Navigatieschema.

De use-cases zijn in dit rapport naar voren gehaald om zo de koppeling tussen de definitiestudie en de pilotontwikkeling duidelijk te houden. De menustructuur laat de structuur en hiërarchie van het gegevensbeheer zien. De schetsen van de schermen (plus de bijbehorende uitleg) dienen voor de technische en functionele aspecten en laten tevens de interactie tussen de gebruiker en het systeem zien. Tot slot laat het navigatieschema de manier van navigeren door de pilot zien.

Aangezien de gebruikte use-cases een simpele opsomming is welke te vinden was in de externe bijlage (Rapport pilotontwikkeling gegevensbeheer) en de schetsen van de schermen met de daarbij behorende uitleg de belangrijkste leidraad tijdens de ontwikkeling van deze pilot zijn geweest zal in deze paragraaf direct verder gaan met de schetsen van de schermen.

#### Schetsen van de schermen

In deze sub paragraaf behandel ik de schetsen van de schermen (plus de bijbehorende toelichting) hier zal duidelijk worden hoe de schetsen van de schermen de use-cases kunnen ondervangen.

De pilot gegevensbeheer bestond uit de volgende pilotdelen:

1. Overzichten viewer: de overzichten voor de gebruiker welke de rechten van een viewer bezit.
2. Overzichten Opdrachtgever: De overzichten welke belangrijk zijn voor de opdrachtgever.

De pilot is in delen opgedeeld omdat de gehele pilot nog steeds te groot was om in één keer te ontwikkelen. De delen zijn zo gekozen dat ze één functionaliteit of een groep van verwante functionaliteiten omvat. Ter toelichting van deze pilot zullen de twee functionaliteiten nader worden toegelicht middels een voorbeeld. Dit zal worden gedaan middels een schets van het scherm, een screenshot van het uiteindelijke resultaat en een impressie van de broncode, tevens zullen hierbij de keuzes en eventuele problemen en oplossingen beschreven worden welke naar voren zijn gekomen tijdens de bouw van deze functionaliteit.

#### Overzichten voor de viewer

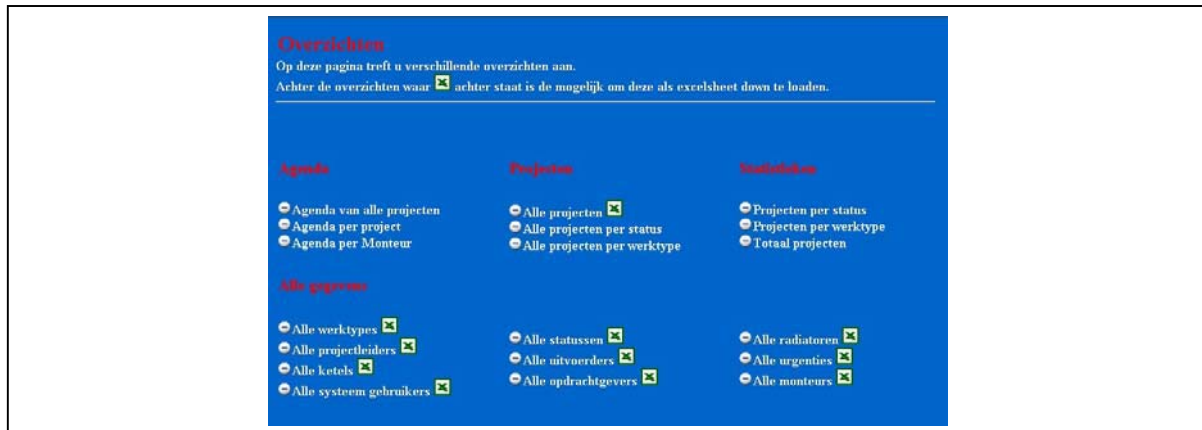
De overzichten van de viewer kunnen tevens bekeken worden door de beheerder en de planner, dit zijn immers allemaal gebruikers welke binnen Bonarius actief zijn. Verschillende overzichten voor dit pilotdeel waren al gemaakt in eerdere pilots.

Zo waren onder de overzichten

1. Agenda per project;
2. Agenda per monteur;
3. Agenda alle projecten;
4. Agenda per werktipe.

Allen al gemaakt tijdens pilot twee: planningssysteem.

Daarnaast zijn tijdens de pilot gegevensbeheer de overzichten gemaakt van alle gegevens. Daardoor zijn deze tezamen gevoegd op één pagina waar de viewer direct na het inloggen op BonaPlan binnenkomt. Dit scherm ziet er als volgt uit:



**Figuur 5.28: Screenshot overzichten BonaPlan**

Het bovenstaande scherm is ontwikkeld met behulp van Dreamweaver. De pagina is een simpele HTML pagina waarop doorverwezen wordt naar de verschillende overzichten. Door voor dit scherm Dreamweaver te gebruiken kon dit scherm snel gemaakt zonder al te veel technische kennis, Dreamweaver maakt namelijk gebruik van het *what you see is what you get* "WYSIWYG" principe waardoor je geen code hoeft te schrijven.

Wat waarschijnlijk direct opvalt in dit scherm zijn de Excel logo's in dit scherm. Wanneer deze logo's achter een overzicht staan is het mogelijk om deze overzichten te downloaden in excel formaat. Deze Excel overzichten zal ik dan ook nader toelichten omdat dit een erg functionele en naar mijn mening mooie functionaliteit van BonaPlan is.

De Excel files worden gemaakt middels een PHP-code. Een fragment van deze code is hieronder te bezichtigen.

```
<?php
header("Content-type: application/octet-stream");
header("Content-Disposition: attachment; filename=alleprojecten.xls");
header("Pragma: no-cache");
header("Expires: 0");
include_once("../inc/dblogin.php");
$rDbConn = connectdb();
$result = mysql_query('select * from project ORDER BY Projectnummer');
$count = mysql_num_fields($result);
for ($i = 0; $i < $count; $i++){
    $header .= mysql_field_name($result, $i)."\t";
}
while($row = mysql_fetch_row($result)){
    $line = "";
    foreach($row as $value){
```

**Figuur 5.29: Fragment Excel code**

In deze PHP-code wordt gebruik gemaakt van de zogenaamde Header functie van PHP. Met deze functie kan onder andere een Excel file worden gegenereerd. In dit geval is de standaard name van deze file alleprojecten.xls. Deze code zorgt er dan ook voor dat alle projecten met alle attributen worden opgehaald uit de database en worden geplaatst in een Excel file (zie hieronder).

	A	B	C	D	E	F	G
1	Projectnummer	Werknummerklant	Straat_projectnaam	Huisnummer	Huisnummer_toevoeging	Plaats	Te
2	200045123		Lissenvaart	70		Zoetermeer	
3	20045369		Andoornvaart	22		Zoetermeer	
4							
5							

Figuur 5.30: Fragment Excel sheet

Deze functionaliteit heb ik ontwikkeld met behulp van informatie op de website <http://www.php.net>. Hier stond namelijk een functionaliteit beschreven dat de header functie kon worden gebruikt voor het maken van PDF files. Na verder onderzoek te hebben gedaan naar deze functionaliteit ben ik erachter gekomen dat dit ook mogelijk was met Excel files.

Nadat deze functionaliteit werkte was het gemakkelijk om dit te hergebruiken voor het maken van allerlei andere overzichten. Door simpel de SQL query (een taal voor het bevragen van een database) `select * from project ORDER BY Projectnummer` kan het overzichten worden aangepast naar wens.

Al tijdens de eerste workshop kwam naar voren dat de opdrachtgever graag een zogenoemde balkenplanning wilde hebben. Aangezien workshop één veel veranderingen met zich meebracht is ervoor gekozen om de balken planning naar de tweede iteratie van de pilot overzichten te schuiven, daardoor is er tijdens de tweede iteratie ook meer aandacht besteed naar de pilot overzichten.

Het balkenschema diende gebaseerd te zijn op het volgende overzicht.

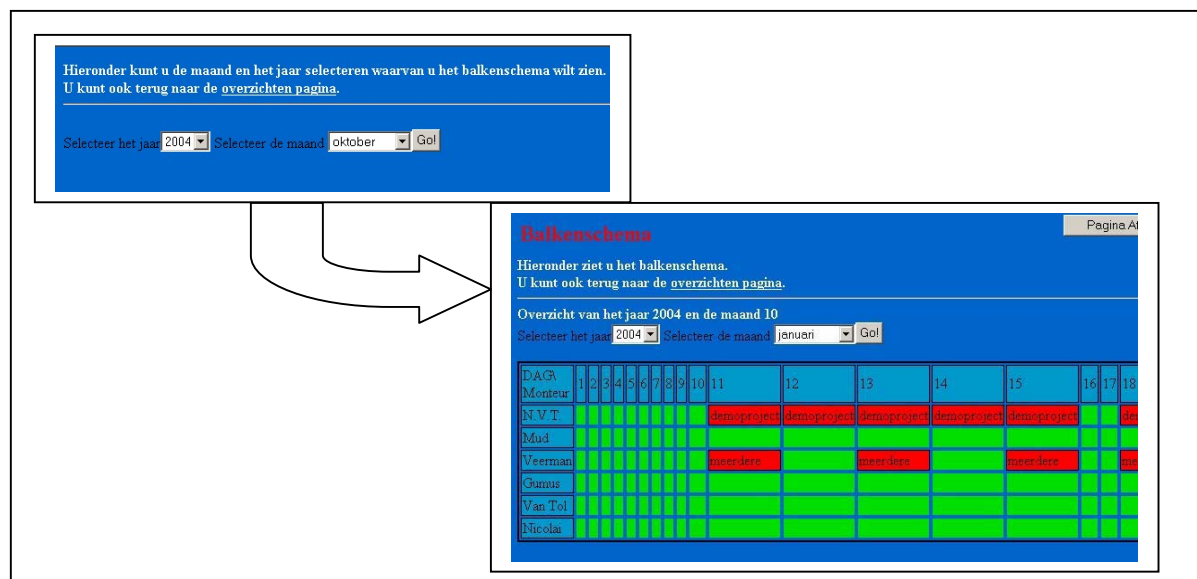
Project	Start Week	End Week	Project Name
Aksoi Enrol (1 man)	21	23	rappertbos
ALB 1	21	23	pre waddenstraat
Bayo Carlo	21	23	pre waddenstraat
Bechler Serge +	21	23	molenwijk schipmolen (1)
B&S	21	23	joubertstraat c155
Copier Jos + Piet	21	23	vd berghlaan
Emah Ron + Tjeerd	21	23	weteringsch
Ginkel geb. R+R	21	23	zuidwest kwad
Goede Cees de	21	23	nieuwbouw
Groeningen S + M v	21	23	nieuwbouw
Gumus Ali	21	23	mutaties
Gumus Derniz	21	23	voorraadpland de pijp / rolwoning
Heemskerk Rob + Rev	21	23	complex 25
IWB supply	21	23	korhoenlaan
Jansen Mark +	21	23	binnenstad
Jong + Koopman	21	23	molenwijk schipmolen (6)
Jong Frank de	21	23	borgerstraat
Jong Michel de + Rona	21	23	verketelingen

Figuur 5.31: Schets balkenschema

Het balkenschema was het meest lastige overzicht welke binnen BonaPlan ontwikkeld diende te worden. In dit overzicht heeft dan ook erg veel tijd gezeten. Het uiteindelijk overzicht diende om één punt te verschillen in vergelijking met de bovenstaande afbeelding. Het schema diende niet op week niveau te zijn maar op dag niveau.

Het idee dat in mijn opkwam toen ik deze afbeelding zag was om een tabel te maken en daar waar een afspraak staat gepland het betreffende vak een kleur te geven met daarin de projectnaam. Dit idee heb ik dan ook gebruikt om het uiteindelijke schema te ontwikkelen. Om het geheel overzichtelijk te houden heb ik ervoor op dit schema per maand te laten generen, tevens komt dit ten goede voor de performance van het systeem.

Hieronder valt eerst het uiteindelijke resultaat te zien hierna zal de achterliggende code nader worden toegelicht.



Figuur 5.32: Balkenschema

Hieronder volgt een fragment van de onderliggende code van het balkenschema

```
<?php
echo "<strong><font color='\#FFFFFF'\>Overzicht van het jaar $jaar en de maand $maand</font></strong>";
echo "<TABLE Border='2' bordercolor='\#000000'\>";

//DEEL 1)
$titelbalk="<TR><TD bgcolor='\#0099CC'\> DAG\ Monteur </TD>";
for($i=1 ; $i<32 ; $i++){
    $titelbalk.="<TD bgcolor='\#0099CC'\>$i</TD>";
}
$titelbalk.="</TR>";
echo $titelbalk;

//database connectie
include_once("../inc/dblogin.php");

$rDbConn = connectdb();

//DEEL 2
$query2= mysql_query("select * from monteur");
while($fetch = mysql_fetch_array($query2)){
    $date= "$jaar$maand"; //$jaar$maand"; //dat zijn dus de maand jaar uit het form

    $html_row="<TR><TD bgcolor='\#0099CC'\>$fetch[Monteur_naam]</TD>";

    $M_id=$fetch[Monteur_id];

    $DagActiviteit=NULL;

    $query= mysql_query("select * from werkt_aan where Monteur_id=$M_id and Datum like '$date%'");
    while($row=mysql_fetch_array($query)){
        $dag = substr($row[Datum], 6, 2);
        $Project=$row[Projectnummer];

        $DagActiviteit[$dag].="$Project "; //met spatie zodat projectnummers erachter niet direct aan elkaar worden
geplakt
    }

    for($i=1 ; $i<32 ; $i++){
        $active=FALSE;
        if(is_array($DagActiviteit)){
            $active=array_key_exists($i,$DagActiviteit);
        }
        if($active){
            $html_row.="<TD bgcolor='red'\>$DagActiviteit[$i]";
        }
        else{
            $html_row.="<TD bgcolor='green'\>";
        }
    }
    $html_row.="</TR>";

    echo $html_row;
}
echo mysql_error();
```

**Figuur 5.33: Code balkeschema**

In deze code word het balkenschema gegenereerd.

Het eerste stuk van de code (//DEEL 1) zorgt voor de benaming van de kolommen van 1 t/m 31 Vervolgens wordt middels het tweede gedeelte van de code (//DEEL 2) ervoor gezorgd dat alle monteurs uit de tabel monteur worden opgehaald en worden geplaatst als naam van de rij binnen de tabel.

De code hierna zorgt ervoor dat van iedere dag van de maand in het jaar (welke geselecteerd zijn in het schermje voor het balkenschema) de afspraken worden opgehaald. Is er op een dag bij een monteur een afspraak dan word de achtergrond van dit hokje rood gekleurd en wordt de naam van het project hierna geplaatst. Is er op de dag bij de monteur geen afspraak dan word de achtergrond groen gekleurd. Door deze code krijgt men dus een schema waarin kan worden afgelezen welke monteur op welke dag is gepland op welk project. Met behulp van dit schema kan tevens worden gekeken welke monteur op welke dag nog niet is ingepland voor een project.



Deze code is geschreven met behulp van een PHP editor. Ik was in staat deze code te schrijven door mijn kennis van tabellen in HTML. De hierboven beschreven code genereert namelijk een HTML tabel met behulp van de database. Nadat dit overzicht was gemaakt kon er verder worden gegaan met het opdrachtgevers gedeelte van BonaPlan.

### Overzichten voor de opdrachtgever

De tweede functionaliteit en tevens het tweede pilotdeel van deze pilot waren de overzichten voor de opdrachtgever. De opdrachtgever logt in met de inlogfunctie welke binnen BonaPlan aanwezig is. Daarna komt de opdrachtgever op een pagina, speciaal voor de opdrachtgevers. Hieronder vind u de schets en het uiteindelijke resultaat


Welkom Opdrachtgever  
Bij Bonaplan

Klik hier om verder te gaan

**DE PROJECTEN VAN OPDRACHTGEVER**

Zoek op projectnaam  
Zoek op complexnummer  
Zoek op werknummer

Projectnummer	Werkn	Projectnaam	Plaats	Complexnummer	Bestand
<a href="#">20004150</a>	12	Lissenvaart	Zoetermeer	23	lets.xls
<a href="#">20043323</a>	135	Andoorn	Zoetermeer	59	Con.doc



Figuur 5.34: Home opdrachtgever

Wanneer de opdrachtgever dus inlogt komt de opdrachtgever op een pagina waar hij persoonlijk begroet word. De naam van de opdrachtgever wordt namelijk aan de hand van de inloggegevens opgehaald uit de database en in het scherm getoond. Na deze pagina komt de opdrachtgever terecht in een scherm waar hij de keuze kan maken voor verschillende overzichten. Tevens zijn deze overzichten te downloaden in Excel formaat en te printen vanaf de website. Daarnaast kan de opdrachtgever zoeken in de projecten op verschillende zoekcriteria. Tijdens de tweede iteratie van deze pilot, naar aanleiding van workshop twee, is er besloten om de bovenstaande twee schetsen samen te voegen tot scherm. Daarnaast is er de mogelijkheid toegevoegd om bestanden (welke aan een project zijn toegevoegd) down te loaden. Tevens heeft de opdrachtgever de mogelijkheid om de detailgegevens van een project te bekijken, daar kan hij onder andere de contact gegevens van de projectleider en de uitvoerder vinden waardoor de opdrachtgever direct contact met deze personen kan opnemen. Met deze mogelijkheid zal de Front-Office en de administratie van Bonarius Verwarming worden ontlast met telefoon gesprekken welke bedoelt zijn voor een opdrachtgever of uitvoerder.

De functionaliteiten van dit pilotdeel waren vlot te ontwikkelen dit omdat er gebruik is gemaakt van de verschillende functies en technieken die al tijdens de andere pilots zijn uitgezocht en geprogrammeerd.

Het uiteindelijke resultaat van dit onderdeel van pilot valt in het onderstaand figuur te zien.

**BONAPLAN** **BONARIUS VERWARMING**

**BAM Woningbouw Welkom Bij BonaPlan®.**

Hieronder vind u uw projecten welke binnen Bonarius bekend zijn. Op deze pagina kunt u zoeken binnen uw projecten op Projectnaam, Complexnummer en uw eigen werknummer. Daarnaast kunt u al uw projecten downloaden in een Excel overzicht door op het excel logo te klikken. Indien er aan een project een bestand is gekoppeld kunt u deze tevens downloaden.

Zoek op Projectnaam    
 Zoek op Complexnummer    
 Zoek op werknummer

Projectnummer	Werknr	Straat Projectnaam	Plaats	Complexnr	Bestand
<a href="#">aapje</a>		<a href="#">aapje</a>			
<a href="#">blaat</a>		<a href="#">blaat</a>			
<a href="#">fiets</a>		<a href="#">fiets</a>			<a href="#">Planning Plandeel 1.xls</a>
<a href="#">opdrachtgeve</a>					
<a href="#">testfile</a>		<a href="#">testfile</a>			<a href="#">alleprojecten.xls</a>

Figuur 5.35: Overzichten opdrachtgever

Met de afronding van deze pilot is BonaPlan in zijn totaliteit ontwikkeld en is deze klaar voor een testsessie, acceptatietest oftewel voor de volgende fase binnen IAD: de invoering.

## 6 Invoering

Na de complete ontwikkeling van BonaPlan was het tijd voor de laatste fase van het project en tevens de laatste fase binnen de ontwikkelmethodiek IAD: de invoering.

Zoals eerder beschreven in de definitiestudie is er voor de project gekozen om het gehele project in één keer in te voeren, het zogenaamde Big-bang-invoeren. Voordat het systeem daadwerkelijk kan worden ingevoerd, zal deze invoering moeten worden gepland.

Zo zal het systeem moeten worden getest, dit zal gedaan worden aan de hand van een testplan, daarnaast dient het gehele systeem in te worden gevoerd wat de nodige invoer van gegevens met zich mee brengt. Daarna zal de opdrachtgever het systeem moeten goedkeuren middels een acceptatie test. Tenslotte zal het systeem worden overgedragen aan Bonarius Verwarming. In dit hoofdstuk zal op de bovenstaande zaken nader worden ingegaan

Zoals eerder vermeld is met name door de wijzigingen met betrekking tot workshop één de invoering uitgesteld. Daardoor is ervoor gekozen om de invoering af te ronden buiten de afstudeerperiode Dit is mogelijk omdat ik na de afstudeer periode nog enkele maanden bij het bedrijf blijft in een vast dienstverband. Tijdens de afstudeerperiode is er voor gekozen om het systeem wel te testen.

### 6.1 Testen van BonaPlan

Het testen van BonaPlan is gebeurt middels een testplan welke is toegevoegd als externe bijlage. Voor het testen van BonaPlan is gekozen voor Black-box testen oftewel het testen op de functionaliteit. Voor white-box testen is bewust niet gekozen aangezien dit het testen op de programmacode betreft en er binnen Bonarius geen kennis aanwezig is met betrekking tot programmacode. Dit zou betekenen dat ikzelf op de code zou moeten testen maar aangezien ik zelf de code heb geschreven ben ik niet objectief te noemen. Tevens is het belangrijkste voor de opdrachtgever dat systeem functioneel werkt, vandaar de keuze voor Black-box testen.

Het testen is gedaan aan de hand van een aantal testcases. Tijdens deze testcases is gekeken of de functionaliteiten naar behoren werken doormiddel van het invoeren, wijzigen en verwijderen van gegevens en dan te kijken of de uitkomsten van deze acties overeen komen met de te verwachte uitkomsten. Tevens is er gekeken of ze in overeenstemming zijn met de functionele en non-functionele systeemeisen. De non-functionele systeemeisen, uit de definitiestudie, zijn hieronder geciteerd.

- Interface eisen zijn de eisen die betrekking hebben op de interface van het planningsprogramma:

  1. Het planningssysteem dient aan te sluiten op de huisstijl en de reeds bestaand website van Bonarius.
  2. Opdrachtgevers dienen zonder uitleg of begeleiding zelf met behulp van de interface de voor hen bestemde overzichten te kunnen bekijken.

Integriteitseisen

De integriteitseisen van het planningssysteem zijn:

  1. Data dient altijd betrouwbaar en up-to-date te zijn zolang deze goed ingevoerd worden door de gebruiker.
  2. Performance eisen

Performance eisen zijn de eisen die betrekking hebben op de performance, ook wel snelheid genoemd, van het planningssysteem.

  1. Elke functionaliteit van het planningsysteem dient binnen maximaal 6 seconden verwerkt te worden.

Operationele eisen

De operationele eisen voor het systeem zijn:

  2. Het systeem dient toegankelijk te zijn met Microsoft Internet Explorer (vanaf versie 5.0).

**Figuur 6.1: Non-funtionele eisen**

### 6.2 Invoeren

De complete invoering zal pas na de afstudeer periode worden voltooid. Inmiddels is het systeem wel al operationeel gemaakt en is er document geschreven waarin staat vermeld hoe het systeem operationeel kan worden gemaakt (zij externe bijlage: Operationeel maken BonaPlan). Daarnaast is er begonnen aan een proefsessie. Tijdens deze proefsessie zal de bestaande planning naast de planning middels BonaPlan worden gedraaid. In deze periode zal worden gekeken of het systeem voldoende functionaliteiten bezit om de volledige planning over te nemen. Eventuele kritische wijzigingen zullen tijdens deze periode nog worden aangepast. Na deze fase zal er over worden gegaan tot het wederom uitvoeren van een testplan, een acceptatietest en niet te vergeten tot de daadwerkelijke overdracht. Helaas valt dit buiten de afstudeerperiode.

## 7 Evaluatie

In de voorgaande hoofdstukken heb de activiteiten, keuzes, technieken en methodes die tijdens mijn afstudeer periode naar voren zijn gekomen behandeld. In deze evaluatie wordt teruggeblikt op de afstudeerperiode. Dit zal gedaan worden aan de hand van het proces en de producten welke ik tijdens deze afstudeerperiode heb doorlopen en opgeleverd. Tenslotte zal deze evaluatie worden beëindigd met de geboekt leerwinsten en natuurlijk een conclusie.

### 7.1 Proces

In deze paragraaf wordt het proces geëvalueerd.

#### **Opdrachtgever, begeleiding en samenwerking**

Tijdens deze afstudeer periode heb ik de beschikking gehad over twee opdrachtgevers die tevens als bedrijfsmentor dienden. De opdrachtgevers hadden de taken duidelijke verdeeld. Voor technische aspecten kon ik terecht bij dhr. Dee en voor functionele aspecten kon ik terecht bij dhr. Molenkamp. Deze scheiding is mij goed bevallen. Ik wist precies met welke vragen ik bij welk persoon moest zijn. Wel moest ik erg opletten dat de opdrachtgevers dezelfde eisen stelden.

De begeleiding door de twee opdrachtgevers, vond ik erg goed. Door de wekelijkse gesprekken welke elke vrijdag plaats vonden wisten wij van elkaar precies waar we mee bezig waren. Verder kon ik altijd terecht met een vraag tussendoor. Daarnaast zijn de gehouden workshops goed ontvangen en hebben de opdrachtgevers en ik hier veel profijt van gehad.

De samenwerking met de collega's is mij erg goed bevallen. Er werd altijd tijd voor mij gemaakt en ik kon erg goed met mijn collega's opschieten. Ik heb dan ook een hele leerzame en gezellige tijd gehad bij Bonarius.

#### **Planning en de manier van plannen**

Terwijl ik een planningssysteem voor Bonarius heb gemaakt is het plannen van mijn eigen werkzaamheden niet de makkelijkste klus geweest. Ik vond het erg moeilijk om een project van dit formaat en over een tijdsbestek van 18 weken in zijn geheel in te plannen. Tijdens het afstuderen heb ik zoals eerder te lezen gekozen voor een bepaalde manier van plannen. Deze manier is deels bevallen. De manier van plannen is bevallen omdat deze erg flexibel was en makkelijk aan te passen. Omdat er op dag niveau werd gepland kon er makkelijk met uurtjes heen en weer geschoven worden. Deze manier van plannen heeft ook zijn nadelen. Doordat de planning steeds verder gedetailleerd werd (van globale planning tot de detaillering in de pilotontwikkelpannen) was het overzicht aan het begin van dit project niet goed genoeg, deze werd dan ook pas in de loop van het project duidelijker en naar mijn gevoel in een te laat stadium. In de toekomst zal ik dan ook op een andere manier gaan plannen.

De globale planning welke in eerste instantie voor dit project is opgesteld kwam redelijk overeen met de werkelijkheid, door een onverwachte extra iteratie is helaas de invoering uitgesteld tot na de afstudeerperiode terwijl deze wel was gepland voor binnen de afstudeerperiode. Doordat ik nog enkele maanden bij het bedrijf blijf kan ik dit op een professionele manier oplossen zodat iedereen tevreden kan zijn.

#### **Methodiek**

De methodiek IAD die ik gebruik heb tijdens dit project is een bekende en voor mij vertrouwde methodiek. Dit is dan ook erg handig geweest tijdens het project ik hoefde zo geen nieuwe methodiek te leren en dat scheelde weer een hoop tijd. IAD heeft mijn geholpen op het project gestructureerde te houden en ervoor gezorgd dat er veel contact en input van de opdrachtgevers is geweest in dit project. Toch zal ik de volgende keer bij eenzelfde soort project uitkijken naar een andere methodiek. Ik ben namelijk van mening dat IAD beter van pas komt bij projecten waar gewerkt wordt met projectteams.

#### **Definitiestudie**

De definitiestudie is een belangrijk document geweest tijdens dit project. Bij de start van dit project waren voor de opdrachtgever en ikzelf de wensen en eisen voor het systeem onduidelijk. Door het schrijven van de definitiestudie zijn de wensen en eisen van de opdrachtgever boven water gekomen en wist ik wat er uiteindelijk gemaakt moest worden. De definitiestudie is dan ook tijdens het gehele

project bijgewerkt en up-to-date gehouden. Om zo een goede overeenkomst te houden met de opdrachtgever.

### **Pilotontwikkeling**

De verschillende pilotontwikkelrapporten zijn tijdens de pilotontwikkelfase van grote waarde geweest. In deze rapporten is uitgewerkt hoe het systeem er uiteindelijk uit moet komen te zien. Deze rapporten vormden dan ook de basis voor het daadwerkelijk coderen van het systeem. Tijdens de ontwikkeling is er twee keer gebruik gemaakt van workshops. Deze workshops hebben gezorgd voor een grote inbreng van de opdrachtgever en eindgebruikers in het systeem, daarnaast zijn deze workshops erg makkelijk geweest ter controle; liggen de opdrachtgever en ik nog op dezelfde lijn. Tijdens de ontwikkeling is er twee keer gebruik gemaakt van workshops. Deze workshops hebben gezorgd voor een grote inbreng van de opdrachtgever en eindgebruikers in het systeem, daarnaast zijn deze workshop erg makkelijk geweest ter controle: liggen de opdrachtgever en ik nog op één lijn. De drie iteraties welke zijn uitgevoerd tijdens deze ontwikkeling zijn ook erg goed bevallen. Het systeem groeide van functionele delen tot een compleet functioneel prettig uitziend geheel. Bijkomend voordeel was dat de opdrachtgever als het ware het systeem zag groeien, waardoor er een goede controle mogelijk was tijdens de ontwikkeling van de pilots.

### **Invoering**

Helaas is de invoering iets wat binnen de afstudeerperiode nog niet is uitgevoerd. Naar aanleiding van een workshop kwam er zo een grote verandering naar voren dat de invoering door tijdgebrek niet meer kon worden uitgevoerd. Wel is het systeem getest en wordt er op dit moment een testperiode gedraaid waarin BonaPlan naast de bestaande planning wordt gedraaid. Om zo te kijken of het systeem aan de functionaliteiten voldoet en in staat is om de gehele planning over te nemen. In de periode na het afstuderen zal BonaPlan compleet worden ingevoerd, geaccepteerd en worden overgedragen.

## **7.2 Product**

In deze paragraaf zullen de verschillende producten worden geëvalueerd.

### **Definitiestudie**

De definitiestudie als product zijnde vind ik een erg functioneel en verzorgd uitziend document. Dit document is zoals vermeld erg waardevol voor mijzelf en voor de opdrachtgever geweest. Daarnaast is het ook bruikbaar als materiaal voor een eventuele andere ontwikkelaar welke BonaPlan zou willen uitbreiden. De kwaliteit van dit document is gewaarborgd door verificatie en validatie technieken welke gebruikt zijn tijdens dit document. Tevens heeft de opdrachtgever het document meerdere malen gelezen en er een positief oordeel over uitgesproken.

### **Pilotontwikkeling**

De verschillende pilotontwikkelrapporten zijn minder interessant voor de opdrachtgever. De ontwerpen zijn bijvoorbeeld erg interessant voor mensen die een uitbreiding willen maken op BonaPlan. Daarnaast hebben deze documenten mij gediend als basis en bouwplan voor de ontwikkeling van de pilots. De documenten zien er naar mijn mening verzorgd uit en zijn zoals gezegd erg bruikbaar geweest.

### **BonaPlan**

Het uiteindelijk programma is toch iets waar ik en de opdrachtgever veel waarde aan hechten. BonaPlan is een compleet en functioneel systeem geworden. Tijdens het testen zijn er nog geen problemen naar boven gekomen. Het invoeren van projecten en aanvullende gegevens is een simpele vanzelfsprekende bezigheid. De overzichten die BonaPlan genereert zorgen niet alleen voor tijdswinst, maar voorkomen ook inconsistentie door fouten die een mens kan maken. Daarnaast is het opdrachtgevers (klanten) gedeelte een mooie service naar de klanten toe welke de administratie en Front-Office werk uit handen moet nemen. Tevens draagt dit gedeelte bij aan de klantgerichtheid welke Bonarius wil uitstralen naar zijn klanten.

Helaas is er binnen BonaPlan nog wel één onopgelost probleem (het wijzig probleem uit paragraaf 5.3.1). Dit probleem is bekend en opgelost met een noodoplossing. Deze noodoplossing is nog niet als vervelend ervaren, maar het is wenselijk dat dit probleem opgelost wordt, hier zal dan ook na de afstudeerperiode aan gewerkt worden.

Al met al is BonaPlan een functionele en mooie applicatie geworden die voor Bonarius tijd zal gaan besparen in verschillende delen van de organisatie. Daarnaast zal de betrouwbaarheid van de verschillende planningen omhoog gaan aangezien de factor mens op enkele plaatsen is geëlimineerd uit het planproces.

### **7.3 Geboekte leerwinst**

Tijdens het afstuderen heb ik veel kennis opgedaan. Deze kennis zijn niet alleen zaken die direct te maken hebben met mijn vakgebied maar ook bijvoorbeeld op communicatief gebied. De volgende leerwinst zij ten opzichte van mijn werkgebied behaald

- Verbetering kennis van HTML;
- Programmeren in PHP in combinatie gebruik MySQL database;
- Toepassen van ontwikkelmethode in werkomstandigheden;
- Interviewen, workshops;
- Opzetten van een webserver (IIS, PHP & MySQL).

Daarnaast heb ik leren werken binnen een organisatie waar ikzelf de kennis ben, hierdoor moest ik dus alles zelf uitzoeken. Dit is dan ook erg leerzaam geweest. Ook op het gebied van effectieve communicatie heb ik een hoop geleerd en heb ik gemerkt dat informele communicatie een zeer belangrijke rol kan spelen in de communicatie tussen de opdrachtgever en ikzelf.

### **7.4 Conclusie**

Binnen Bonarius heb ik een plezierige tijd gehad. Er is door mij dan ook met veel plezier aan de opdracht gewerkt. De resultaten van de afstudeer periode zijn voor zowel de opdrachtgever als ikzelf van goede kwaliteit te noemen. De opdrachtgever is tevreden over het systeem. Ikzelf ben erg tevreden over het proces en de behaalde resultaten, voor Bonarius en mijzelf is het afstuderen dan ook geslaagd.

## Figuren lijst

Figuur 1.1: Organogram.....	8
Figuur 3.1: Iteratie strategie Big-Bang-invoeren .....	12
Figuur 3.2: Fuikprocedure .....	13
Figuur 3.3: Planning .....	14
Figuur 3.4: Tussenoplossing planning .....	15
Figuur 3.5: Detailplanning (deelplanning).....	15
Figuur 3.6: Workshop 1.....	16
Figuur 4.1: Probleem en doelstelling .....	18
Figuur 4.2: Pilotstrategie .....	19
Figuur 4.3: Fragment systeemeisen BonaPlan .....	20
Figuur 4.4: Use-case project toevoegen .....	21
Figuur 4.5: Klassediagram.....	22
Figuur 4.6: Pilot identificatie .....	25
Figuur 4.7: Verificatie en validatie .....	25
Figuur 4.8: Verificatie en validatie .....	25
Figuur 5.1: Fragment klassediagram.....	27
Figuur 5.2: Fragment relationeel implementatiemodel.....	27
Figuur 5.3: Tabel User_soort & tabel opdrachtgever .....	28
Figuur 5.4: Fragment relationeel implementatiemodel.....	28
Figuur 5.5: Testcode database.....	29
Figuur 5.6: Verificatie en validatie database.....	30
Figuur 5.7: Schermschets naar aanleiding van workshop 1.....	32
Figuur 5.8: Frameset BonaPlan.....	33
Figuur 5.9: Schermschets en resultaat inloggen .....	34
Figuur 5.10: Fragment code inlogfunctie.....	34
Figuur 5.11: Schermschets project toevoegen .....	35
Figuur 5.12: Screenshot project toevoegen .....	36
Figuur 5.13: Fragment code project toevoegen .....	36
Figuur 5.14: Schermschets plan functionaliteit .....	37
Figuur 5.15: Screenshot plan functionaliteit .....	37
Figuur 5.16: Script voor omreken weeknummers .....	38
Figuur 5.17: Script van plan functie .....	39
Figuur 5.18: De Agenda .....	39
Figuur 5.19: Code van agenda .....	40
Figuur 5.20: Time-box planningssysteem.....	41
Figuur 5.21: Verificatie en validatie .....	42
Figuur 5.22: Verificatie en validatie .....	42
Figuur 5.23: Verzameling gegevensbeheer .....	45
Figuur 5.24: Schermschets opdrachtgever wijzigen .....	45
Figuur 5.25: Schermschets opdrachtgever verwijderen .....	46
Figuur 5.26: Schermschets opdrachtgever toevoegen .....	47
Figuur 5.27: Screenshot opdrachtgever wijzigen .....	47
Figuur 5.28: Screenshot overzichten BonaPlan.....	51
Figuur 5.29: Fragment Excel code.....	51
Figuur 5.30: Fragment Excel sheet.....	52
Figuur 5.31: Schets balkenschema .....	52
Figuur 5.32: Balkenschema .....	53
Figuur 5.33: Code balkeschema.....	54
Figuur 5.34: Home opdrachtgever .....	55
Figuur 5.35: Overzichten opdrachtgever .....	56
Figuur 6.1: Non-funtionele eisen .....	57



## Verklarende woordenlijst

**Actor:** Een actor is een entiteit die buiten het systeem staat en die direct communiceert met het systeem.

**Black box testen:** Een testtechniek waarbij er geen kennis van de code van een informatiesysteem wordt vereist. Er wordt alleen gekeken hoe, bij een bepaalde invoer, het product zich gedraagt en wat de uitvoer is.

**Definitiestudie:** De eerste fase uit de ontwikkelmethode IAD waarin de doelen en beperkingen van het te ontwikkelen systeem worden geanalyseerd.

**Dreamweaver:** Een 'What You See Is what You Get' editor voor het ontwikkelen van websites.

**HTML:** Hyper Text Markup Language, een opmaaktaal voor het ontwikkelen van websites.

**IAD:** Iterative Application Design, een ontwikkelmethode waarin het informatiesysteem tot stand komt via een aantal snel uitgevoerd iteraties van de complete ontwikkelcyclus.

**Iteratie:** Er is sprake van een iteratie, wanneer een fase uit de ontwikkelmethode IAD meerdere malen wordt uitgevoerd.

**Kernelpilot:** Een niet functionele subset van het totale systeem welke eerst ontwikkeld dient te worden omdat volgende pilots van deze pilot afhankelijk zijn.

**Klassediagram:** Een model dat alle klassen en attributen van de te ontwikkelen database bevat en de onderlinge relaties tussen de verschillende klassen laat zien.

**Menustructuur:** Een model ter verduidelijking van de hiërarchie van de schermen waarin te zien is waar de verschillende schermen zich in de hiërarchie van het totale systeem bevinden.

**MySQL:** Een relationeel 'open source' database managementsysteem.

**Navigatieschema:** Een schema ter verduidelijking van de schermen waarin onderlinge relaties en navigatiemogelijkheden tussen de schermen staan weergegeven.

**PHP:** (Hypertext PreProcessor), een open source scripttaal en engine.

**Pilot:** Een functioneel operationele subset van het totale systeem.

**Pilotontwikkeling:** De tweede fase uit de ontwikkelmethode IAD waarin de verschillende systeemonderdelen (pilots) daadwerkelijk worden ontwikkelt.

**RIM:** (Relationeel Implementatie Model), een model waarmee een relationele database kan worden geïmplementeerd.

**RRM:** (Relationeel Representatie Model), een model waarmee een relationele database wordt gerepresenteerd.

**UML:** Unified Modeling Language, een objectgeoriënteerde taal, die een set technieken aanlevert om het beoogde systeem te kunnen modelleren.

**Use case:** Een beschrijving van een reeks interacties tussen één of meer actoren en het systeem. Het gezichtspunt hierbij is het externe gedrag van het systeem, vanuit het gezichtspunt van de gebruiker.

**Webserver:** Een locatie die online benaderbaar is en verantwoordelijk is voor het beantwoorden van verzoeken om webpagina's. In dit verslag staat deze gelijk aan ontwikkelomgeving.

## Literatuurlijst

### Boeken:

- Tim Converse & Joyce Park, Het complete handboek PHP 4, Den Haag: Academic Services, december 2003
- Eleanor Pascoe-Samson, [Organisatie, besturing en informatie](#), Deventer: Kluwer Bedrijfs-informatie, september 1999
- R.J.H. Tolido, IAD Het evolutionair ontwikkelen van informatiesystemen, Utrecht: Academic Services, oktober 2001
- Jos Warmer & Anneke Kleppe, Praktisch UML, Amsterdam: Addison Wesley Longman Nederland BV., 1999

### Handleidingen:

- Moduleboek inleiding systeemontwikkeling, Haagse Hogeschool, sector informatica
- Moduleboek Rapportagetechnieken, Haagse hogeschool
- Modulehandleiding SO-deel IP-841, Haagse Hogeschool, sector informatica
- Moduleboek inleiding databases, Haagse hogeschool, sector informatica

### Websites:

- ISO-8601, <http://www.cl.cam.ac.uk/~mgk25/iso-time.html>
- PHP informatieve website, [www.phpfreakz.nl](http://www.phpfreakz.nl)
- PHP homepage, [www.php.net](http://www.php.net)
- Testen, <http://www.ictbeheergroep.amsterdam.nl/ictbeheergroep/publicaties/VISIE%20OP%20TESTEN.pdf>

## **INTERNE BIJLAGEN**

### **Bijlage 1 : Opdrachtschrijving**

## Opdrachtgever en uitvoerende

### Opdrachtgever

De opdrachtgever van dit project is Bonarius Verwarming. Binnen Bonarius Verwarming zijn ter begeleiding en ondersteuning twee contactpersonen/bedrijfsmentors aangewezen:

M.C.J. Dee	<a href="mailto:Dee@bonarius.com">Dee@bonarius.com</a>	tel: 020-4074904
G.C. Molenkamp	<a href="mailto:Molenkamp@bonarius.com">Molenkamp@bonarius.com</a>	tel: 020-4074923

Tijdens kantooruren zijn dhr. Dee en dhr. Molenkamp te vinden in het complex van Bonarius Verwarming op de Zwanenburgerdijk 291 te Zwanenburg.

### Uitvoerende

Dit project wordt uitgevoerd door Nicolai Roos. Nicolai Roos studeert informatica en informatiekunde aan de Haagse Hogeschool. Dit project wordt door hem uitgevoerd in het kader van een afstudeeropdracht.

Nicolai Roos	<a href="mailto:Roos@bonarius.com">Roos@bonarius.com</a> <a href="mailto:20007220@student.hhs.nl">20007220@student.hhs.nl</a>	tel : 06-14276478
--------------	--	-------------------

## Inleiding en omgevingsschets

Bonarius verwarming heeft in de afgelopen 50 jaar bijna 100.000 verwarmingsinstallaties aangelegd. Voor bijna 60.000 huishoudens in de Noordelijke Randstad verzorgt Bonarius het periodiek- en storingsonderhoud van cv-installaties. De Bonarius bedrijven hebben zich gespecialiseerd in individuele verwarmingsinstallaties voor de particuliere markt, voornamelijk huurwoningen, maar ook koopwoningen. In de meeste gevallen gaat het om traditionele HR-ketels, maar in toenemende mate eveneens om vernieuwende, energiezuinige systemen voor zonne-energie, mechanische ventilatie en toepassingen van warmtepompen.

Bonarius behoort tot de oudste bedrijven van Nederland (vestigingsjaar 1785). De afgelopen jaren heeft Bonarius een periode van sterke groei doorgemaakt. Dit heeft geleid tot een verandering in de organisatie, met accenten op een verdere professionalisering en een servicegerichte werkwijze. Tot dit veranderingsproces behoort ook de verhuizing, in maart 2004, naar een nieuwe locatie in Zwanenburg welke op 1 april jongstleden officieel is geopend.

Het werk van bonarius valt grofweg in te delen in drie taken:

#### 1. Service:

Bonarius heeft met haar grotere klanten (woningcorporaties en institutionele beleggers) veelal een all-in contract afgesloten, binnen deze overeenkomsten zijn twee zaken geregeld;

##### Onderhoud:

Cv-ketels dienen, conform de geldende voorschriften, periodiek te worden onderhouden. Hierbij wordt de ketel schoongemaakt en geïnspecteerd en waar nodig onderdelen vervangen.

##### Storingen:

Als er iets mankeert aan de cv-installatie nemen de bewoners rechtstreeks contact op met Bonarius die er vervolgens voor zorgt dat er een monteur ter plaatse komt repareren.

#### 2. Het vervangen van ketels:

Op het moment dat een servicemonteur constateert dat een cv-ketel niet meer te repareren is of dat deze niet meer veilig werkt dient deze te worden vervangen. Bonarius vervangt de ketel en voert tevens de benodigde bouwkundige voorzieningen uit.

#### 3. Het plaatsen van volledige cv-installaties:

Wanneer een woning muteert van huurder zal de woningcorporatie veelal besluiten om, als de woning nog niet is voorzien van cv, hier alsnog cv in aan te laten leggen. Ook als een zittende huurder graag cv wil hebben dan kan deze, na acceptatie van bijbehorende huurverhoging, geplaatst worden. Bonarius plaatst de installatie en voert tevens de benodigde bouwkundige voorzieningen uit.

## Probleemstelling

De grote groei van Bonarius verwarming heeft er voor gezorgd dat er behoefte is aan een andere manier van plannen. Waarbij in het kader van klantgericht werken deze via internet voor opdrachtgevers (klanten) beschikbaar moet zijn voor status navragen. Op dit moment worden alle planningen gemaakt in Excel. Alle overzichten worden handmatig gemaakt en weer ingevoerd in Excel. Dit is niet alleen een tijdrovende klus maar ook erg omslachtig.

## Doelstelling van de opdracht

Het doel van de afstudeeropdracht is het ontwerpen, ontwikkelen en implementeren van een online planningssysteem voor Bonarius verwarming (BonaPlan). Het planningssysteem moet ervoor zorgen dat mensen en middelen efficiënt, en minder omslachtig, kunnen worden ingepland voor de verschillende projecten.

Tevens dient het planningssysteem meer inzicht te geven aan opdrachtgevers (klanten) zodat zij ten alle tijden de status van een project kunnen bekijken. Dit is onderdeel van de verdere professionalisering van Bonarius verwarming. Ook wil men middels deze functie van BonaPlan de service van Bonarius verwarming proberen te ontlasten (minder status navragen).

## Uitgangssituatie

### Benodigde software

De volgende software zal worden gebruikt:

- **MS Office pakket:**  
Het MS Office pakket zal worden gebruikt voor het ontwerp van het planningssysteem en het schrijven van verslagen.
- **PHP editor:**  
Het planningssysteem zal worden geschreven in PHP. Om makkelijke te kunnen ontwikkelen met PHP zal gebruikt worden gemaakt van een PHP editor, dit om syntax fouten te voorkomen en het ontwikkelen te vergemakkelijken.
- **MySQL Database:**  
De database welke de basis van het project zal vormen zal worden gemaakt in MySQL. Dit is een welbekende, gratis en snelle combinatie met PHP.
- **Webserver:**  
Binnen bonarius is een server aanwezig welke kan worden ingericht als webserver. Voor de webserver zal Internet Information Server (ISS: webserver programma), PHP software(ondersteuning voor PHP) en MySQL (database) benodigd zijn.

### Benodigde hardware

Binnen Bonarius zal er een computer ter beschikking zijn met daarop een compleet pakket van Operatingsysteem (Windows 2000) tekstverwerkingsoftware (Office 2000), webbrowser en een internetverbinding.

Daarnaast zal tijdens dit project gebruik worden gemaakt van een webserver-computer. Deze is reeds in het bezit van Bonarius en biedt genoeg capaciteit voor de toekomstige functies. Deze dient wel nog ingericht te worden, dit zal tijdens het project worden gedaan.

### Aanwezige ideeën

Het enige idee met betrekking tot het planningssysteem is dat dit web-based dient te zijn en dat het gecreëerd zou moeten kunnen worden door middel van een combinatie van PHP en MySQL.

## Concrete werkzaamheden

### Uit te voeren activiteiten

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

- Opstellen plan van aanpak.
- Definitiestudie:
  - Opstellen plan van aanpak definitiestudie.
  - Definieren ontwikkelscenario.
  - Definieren systeemeisen.
  - Bepalen systeemconcept.
  - Beschouwen van technische structuur.
  - Beschouwen van organisatorische inrichting.
  - Opstellen van pilotplan.
- Pilotontwikkeling:
  - Opstellen plan van aanpak (per pilot)
  - Specificeren functionele structuur pilot.
  - Specificeren globaal-technische structuur pilot.
  - Specificeren globaal-organisatorische inrichting.
  - Opstellen pilotontwikkelplan.
  - Bouw software bouweenheden.
  - Integreer bouweenheden.
  - Beoordeel en test pilotdelen.
- Invoering:
  - Opstellen plan van aanpak.
  - Opstellen invoeringsprocedures planningssysteem.
  - Invoeren planningssysteem.
  - Samenstellen handleidingen planningssysteem samen.
  - Uitvoeren acceptatietest.
  - Overdragen Planningssysteem.

### Te hanteren methodieken en technieken

Bij de ontwikkeling van het Planningssysteem zal gebruikt worden gemaakt van de ontwikkelmethodiek Iterative Application Design (IAD).

Te gebruiken technieken zijn:

- **UML:**  
Dit zal gebruikt worden om het systeem nader te specificeren en een goede basis te vormen voor de bouw van het systeem. Denk bijvoorbeeld aan USE-CASES en Klassediagrammen
- **Interviewtechnieken:**  
Bijvoorbeeld het stellen van open en gesloten vragen, LSD (luisteren, samenvatten en doorvragen). Deze zullen gebruikt worden om overeenstemming te bereiken met de opdrachtgever en meningen van gebruikers te achterhalen.
- **Timeboxing:**  
Voor het plannen van de pilots.

### Op te leveren producten

De volgende producten zullen opgeleverd worden:

1. Plan van aanpak.
2. Definitiestudie.
3. Pilotontwikkelplannen.
4. Planningssysteem (BonaPlan).
5. Testrapport.
6. Invoerings rapport.
7. Handleiding.
8. Systeem Documentatie.

## **Bijlage 2 : Plan van aanpak**



## Opdrachtomschrijving

Deze staat in bijlage 1

### Afbakening opdracht

Dit project omvat een de ontwikkeling van een online planningssysteem voor Bonarius Verwarming. Het online planningssysteem zal de volgende planningen omvatten

1. Ketel vervangingen.
2. Plaatsen van volledige cv installaties.
3. Service (onderhoud en storingen).

Daarnaast zullen medewerkers van Bonarius worden toegevoegd in dit systeem, om deze te kunnen toekennen aan de projecten. Aangezien dit systeem geen personeelsbestand dient te worden zal van de medewerkers alleen de voornaam, achternaam en het telefoonnummer worden opgeslagen.

### Randvoorwaarden

Om het project te kunnen voltooien dient de opdrachtgever nauw betrokken te zijn met de gehele ontwikkeling van het planningssysteem, dit om het beoogde systeem compleet aan de wensen en eisen van de opdrachtgever te laten voldoen. Tevens is er door de nauwe betrokkenheid van de opdrachtgever een constante controle op de uitvoering van het project.

Het online planningssysteem dient na oplevering onderhoudbaar te zijn. Verder dient het zonder al te veel voorkennis te gebruiken te zijn.

### Risicofactoren

Het teveel "hooi op onze vork" nemen kan een risico zijn voor dit project. Hierdoor kan we overrompeld worden door het werk wat verricht dient te worden, waardoor we dit project niet tot een goed eindresultaat kunnen brengen. Dit risico moeten we zien te voorkomen door onze opdracht van te voren goed af te bakenen en ons hier aan te houden.

### Projectorganisatie

Gedurende dit project is er geen sprake van een projectorganisatie aangezien het gehele project door 1 persoon zal worden uitgevoerd. Dit persoon is verantwoordelijk voor alle aspecten van het project.

### Wijze van rapporteren

Rapportage zal plaatsvinden in de vorm van wekelijkse contactmomenten met de bedrijfsmentors. Voorafgaand aan het contactmoment zal er mondeling een afspraak worden gemaakt wat betreft de tijd en aard van het contactmoment.

Daarnaast staan de bedrijfsmentors de gehele week open voor vragen en controle.

### Benodigde mensen/ middelen

#### Opdrachtgever

De opdrachtgever van dit project is Bonarius Verwarming. Binnen Bonarius Verwarming zijn ter begeleiding en ondersteuning twee contactpersonen/bedrijfsmentors aangewezen:

M.C.J. Dee	<a href="mailto:Dee@bonarius.com">Dee@bonarius.com</a>	tel: 020-4074904
G.C. Molenkamp	<a href="mailto:Molenkamp@bonarius.com">Molenkamp@bonarius.com</a>	tel: 020-4074923

Tijden kantooruren zijn dhr. Dee en dhr. Molen kamp te vinden in het complex van Bonarius Verwarming op de Zwanenburgerdijk 291 te Zwanenburg.

#### Uitvoerende

Dit project wordt uitgevoerd door Nicolai Roos. Nicolai Roos studeert informatica en informatiekunde aan de Haagse Hogeschool. Dit project word door hem uitgevoerd in het kader van een afstudeeropdracht.

Nicolai Roos	<a href="mailto:Roos@bonarius.com">Roos@bonarius.com</a> <a href="mailto:20007220@student.hhs.nl">20007220@student.hhs.nl</a>	tel : 06-14276478
--------------	--	-------------------

### Hardware (middelen)

Binnen Bonarius zal er een computer ter beschikking zijn met daarop een compleet pakket van Operatingssysteem (Windows 2000) tekstverwerkingsoftware (Office 2000), webbrowser en een internetverbinding.

Daarnaast zal tijdens dit project gebruik worden gemaakt van een webserver-computer. Deze is reeds in het bezit van Bonarius en biedt genoeg capaciteit voor de toekomstige functies. Deze dient wel nog ingericht te worden, dit zal tijdens het project worden gedaan.

### Globale Planning

Voor dit project staan 15 weken ter beschikking dit komt neer op een totaal van 75 dagen oftewel 600 uur.

Hieronder staat een planning vermeld

Wat direct opvalt is dat de werkzaamheden nog niet zijn ingedeeld in data, maar in dagen dit heeft te maken met het feit dat pas met de definitiestudie bekend is in welke onderdelen het gehele project wordt gehakt.

Tijdens het begin van de definitiestudie zal de planning dan ook verder gedetailleerd worden.

Activiteit	Dagen
<b>Globaal Plan van aanpak</b>	<b>3</b>
<b>Oriëntatie bedrijf + opdracht</b>	<b>2</b>
<b>Definitiestudie</b>	<b>21</b>
Specificeren plan van aanpak	1
Analyse	1
Opstellen ontwikkelscenario	2
Bepalen systeemeisen	3
Opstellen systeemconcept	5
Opstellen technische structuur	5
• Onderzoek webserver	1
• Onderzoek PHP of ASP	1
• Onderzoek Database architectuur	1
• Bouw webserver	1
Opstellen pilotplan	3
Verificatie en validatie	1
<b>Pilotontwikkeling</b>	<b>44</b>
Specificeren plan van aanpak	2
Bepalen globaal functionele structuur	15
Bepalen globaal organisatorische inrichting	2
Opstellen pilotontwikkelplan	4
Bouwen bouweenheden	20
Verificatie en validatie	1
<b>Testen</b>	<b>5</b>
<b>TOTAAL</b>	<b>75</b>
<b>Afstudeerverslag</b>	<b>15</b>

## Detail planning

Dit is de detailplanning deze is ingedeeld aan de hand van de doorlopen iteratie. Voor een nog diepere detail-planningen kunt u in de verschillende pilotontwikkel rapporten vinden.

### Eerste iteratie

- Definitiestudie ( 15 dagen)
  - Plan van aanpak
  - Analyse
  - Ontwikkelscenario
  - Systeemeisen
  - Systeemconcept
  - Technische structuur
    - Onderzoek naar webserver
    - Onderzoek PHP of ASP
    - Onderzoek database architectuur
    - Bouw webserver
  - Organisatorische inrichting
  - Pilotplan
  - Verificatie en validatie
- Pilot 1: database (5 dagen)
  - Plan van aanpak
  - Functionele structuur
  - Pilotontwikkeling
  - Verificatie en validatie
  - Bouw database
- Pilot 2: Planningssysteem + Pilot 3: Gegevensbeheer (16 dagen)
  - Plan van aanpak
  - Functionele structuur
  - Organisatorische inrichting
  - Pilotontwikkelplan
  - Verificatie en validatie
  - Bouw Planningssysteem + Gegevensbeheer

### Tweede iteratie

- Definitiestudie (4 dagen)
  - Plan van aanpak
  - Ontwikkelscenario
  - Systeemeisen
  - Systeemconcept
  - Technische structuur
  - Organisatorische inrichting
  - Pilotplan
  - Verificatie en validatie
- Pilot 1: database (1 dag)
  - Plan van aanpak
  - Functionele structuur
  - Pilotontwikkeling
  - Verificatie en validatie
  - Bouw database
- Pilot 2: Planningssysteem + Pilot 3: Gegevensbeheer (7 dagen)
  - Plan van aanpak
  - Functionele structuur
  - Organisatorische inrichting
  - Pilotontwikkelplan
  - Verificatie en validatie
  - Bouw Planningssysteem + Gegevensbeheer

- Pilot 4: Overzichten (5 dagen)
  - Plan van aanpak
  - Functionele structuur
  - Organisatorische inrichting
  - Pilotontwikkelplan
  - Verificatie en validatie
  - Bouw overzichten

#### **Derde iteratie**

- Definitiestudie (2 dagen)
  - Plan van aanpak
  - Ontwikkelscenario
  - Systeemeisen
  - Systeemconcept
  - Technische structuur
  - Organisatorische inrichting
  - Pilotplan
  - Verificatie en validatie
- Pilot 1: database (1 dag)
  - Plan van aanpak
  - Functionele structuur
  - Pilotontwikkeling
  - Verificatie en validatie
  - Bouw database
- Pilot 2: Planningssysteem + Pilot 3: Gegevensbeheer (4 dagen)
  - Plan van aanpak
  - Functionele structuur
  - Organisatorische inrichting
  - Pilotontwikkelplan
  - Verificatie en validatie
  - Bouw Planningssysteem + Gegevensbeheer
- Pilot 4: Overzichten (5 dagen)
  - Plan van aanpak
  - Functionele structuur
  - Organisatorische inrichting
  - Pilotontwikkelplan
  - Verificatie en validatie
  - Bouw overzichten
- **Invoering**
  - Testen (5 dagen)

## Beschrijving mijlpaalproducten

De volgende mijlproducten heb ik opgesteld en wil ik opleveren aan de opdrachtgever.

### **Plan van aanpak**

Hierin wordt beschreven hoe ik de opdracht wil gaan aanpakken.

### **Definitiestudie**

Door middel van dit document zal er beschreven worden wat de gevolgen zijn van het opleveren van het planningssysteem. Verder zal er in beschreven worden wat onze verdere stappen zullen zijn tot voltooiing van dit project.

### **Pilotontwikkelplan**

Binnen IAD wordt er gebruik gemaakt van pilots. De opdracht wordt in stukken verdeeld, zodat het overzichtelijker wordt. Elk deel wordt een pilot genoemd en zullen beschreven worden in het Pilotontwikkelplan.

### **Planningssysteem**

Dit zal het gehele werkende planningssysteem voor Bonarius Verwarming zijn.

### **Testrapport**

In dit rapport staat de testen en de conclusies van het planningssysteem van Bonarius beschreven.

### **Invoeringsrapport**

Hierin zal komen te staan op wat voor een manier en hoe we de producten zullen gaan implementeren.

### **Handleiding**

De handleiding van het gehele planningssysteem.

### **Systeemdokumentatie.**

Dit zal alle gemaakte documentatie, keuzes en de source code van het planningssysteem omvatten.

## Kosten/ batenanalyse

### **Kosten voor manuren:**

Dit project wordt uitgevoerd in het kader van een afstudeeropdracht. De vergoeding voor dit project zal 350€ per maand zijn.

### **Kosten voor aanschaf software:**

Er zal gebruik gemaakt worden van aanwezige software of software die gratis is.

### **Kosten voor aanschaf hardware:**

De hardware is al aanwezig en zal dus ook niet aan geschaft hoeven te worden.

## **Bijlage 3: Workshop 1**

# WORKSHOP/ DEMONSTRATIE PROTOTYPE PLANNINGSSYSTEEM

Datum: 13-7-2004

Tijd: C.A 11:00

## AANWEZIGEN

Gebruikers: G.C. Molenkamp  
Ontwikkelaar/programmeur: N.E. Roos

Welkom bij de workshop/ demonstratie van het prototype planningssysteem. In deze workshop willen we een eerste impressie geven van het planningssysteem. De workshop zal bestaan uit het uitvoeren van enkele opdrachten, daarnaast is er ruimte voor vragen, opmerkingen, en wensen en eisen. Door deze workshop kan, waar nodig, het project nog worden bijgestuurd.

## DEEL 1

### Opdracht 1.

Om met het planningssysteem te kunnen plannen, dienen er verschillende gegevens binnen het systeem bekend gemaakt te worden. Een groot gedeelte van dit systeem bestaat dan ook uit gegevensbeheer. Ter introductie van het planningssysteem zullen we enkele opdrachten uitvoeren met betrekking tot het gegevensbeheer.

### Opdracht 1.1

1. Start internet explorer op.
2. Surf naar het volgende adres: <http://fs02/test/bonaplan/bonaplan.html>
3. Druk op de naam BonaPlan.
4. Er verschijnt een login scherm, log in met de volgende gegevens  
Username: nicolai paswoord: nicolai.
5. Druk op de link gegevensbeheer.

#### Vragen, opmerkingen, wensen of eisen m.b.t inlog gedeelte:

Geen extra wensen eisen of opmerkingen m.b.t het inlog gedeelte.

### Opdracht 1.2

Alle gegevensbeheer onderdelen werken op dezelfde manier, daarom zullen we niet alle schermen doorlopen.

6. U bevindt zich in het scherm gegevensbeheer druk op de link PROJECTLEIDER.
7. U ziet een overzicht van de aanwezige projectleider. Druk op het ID om de detail gegevens te zien.
8. Druk op de link terug.
9. Druk op de link nieuwe projectleider en voeg een nieuwe projectleider met de volgende gegevens toe.  
Naam: Nicolai Roos  
Telefoonnummer: 079-3317731  
Na het toevoegen druk u op de link terug.
10. U ziet dat de projectleider is toegevoegd.

#### Vragen, opmerkingen, wensen of eisen m.b.t. toevoegen:

Spreek voor zich niks op toe te voegen.

11. Druk op het ID van de projectleider die u zojuist heeft toegevoegd.
12. Druk op de knop wijzigen.
13. Wijzig de gegevens van de betreffende projectleider en druk op wijzigen.
14. Druk op terug.
15. Druk wederom op de zojuist gewijzigde projectleider.
16. Druk op de link verwijderen en verwijder de projectleider.
17. Druk op OK.
18. Druk op de link HOME.



U heeft nu een projectleider, toegevoegd, gewijzigd en verwijderd.

**Vragen, opmerkingen, wensen of eisen m.b.t. gegevensbeheer:**

De continuïteit van het programma bevat goed, de manier van toevoegen wijzigen en verwijderen hoeft niet aangepast te worden.

**DEEL 2:**

Deel 2 zal bestaan uit het toevoegen, wijzigen en verwijderen van een project  
We gaan er van uit dat u bent ingelogd en u zich bevindt in het keuzescherf.

**Opdracht 2.1**

Projecten, is eigenlijk een vorm van gegevensbeheer, maar aangezien, dit een van de grotere onderdelen van het systeem omvat is deze gescheiden van de rest.

19. U bevindt zich in het centrale keuze scherm druk op de link projecten
20. U ziet een overzicht van de aanwezige projecten. Druk op het ID om de detail gegevens te zien.
21. Druk vervolgens op de link terug.
22. Druk op de link nieuw project en voeg een nieuw project met de volgende gegevens toe  
Projectnummer: 20043125  
Werknummerklant: 123-5696635  
Straat\_projectnaam: Zwaneburgerdijk  
Plaats: Zwanenburg  
Huisnummer: 12  
Werkttype: Mutatie  
Na het toevoegen druk u op de link terug.
23. U ziet dat het project is toegevoegd.

**Vragen, opmerkingen, wensen of eisen m.b.t. toevoegen:**

Aangezien de offertes worden bijgehouden in een ander systeem (SYNTESS), kan dat uit dit systeem verwijderd worden, dit om dubbel werk te voorkomen en geen inconsistente gegevens te creëren.

24. Druk op het idee van het project die u zojuist heeft toegevoegd.
25. Druk op de knop wijzigen.
26. Wijzig de gegevens van het betreffende project en druk op wijzigen
27. Druk op terug.
28. Druk wederom op de zojuist gewijzigde project.
29. Druk op de link verwijderen en verwijder de project.
30. Druk op OK.
31. Druk op de link HOME.

U heeft nu een project, toegevoegd, gewijzigd en verwijderd.

**Vragen, opmerkingen, wensen of eisen m.b.t. gegevensbeheer:**

Tot nu is het gebruik van het program makkelijk en vanzelf sprekend, grafisch zo er nog het een en ander verbeterd kunnen worden.

**DEEL 3:**

Deel 3 zal bestaan uit het plannen van monteurs en projecten een van de belangrijke zaken van het project. Aangezien dit gedeelte nog niet geheel functioneel is zal deze aan een hand van een demonstratie getoond worden. Tijdens en na deze demonstratie is er ruimte voor vragen, opmerkingen wensen en eisen.

**Vragen, opmerkingen, wensen of eisen m.b.t. plannen:**

Het plannen dient op een andere manier te gebeuren: Graag zou ik per week nummer kunnen willen plannen, per dag (zoals het nu gaat) neemt te veel werk met zich mee en daardoor is het werk niet erg efficiënt. Efficiëntie is een van de belangrijkste zaken van het systeem.

Daarnaast zou het wenselijk zijn dat het plannings gedeelte aanroepbaar is vanuit het projecten gedeelte van het systeem.

### **EINDE VAN WORKSHOP/ DEMONSTRATIE BonaPlan**

Hieronder volgen de wensen, eisen en vragen en opmerkingen die tijdens deze workshop/demonstratie naar voren zijn gekomen en de veranderingen die deze met zich mee brengen.

(zie volgende pagina)

Hieronder staan de vragen, opmerkingen wensen en eisen welke een oplossing vereisen en welke dus aangepast dienen te worden.

- **Vragen, opmerkingen, wensen of eisen m.b.t. toevoegen (1) :**  
Aangezien de offertes worden bijgehouden in een ander systeem (SYNTESS), kan dat uit dit systeem verwijderd worden, dit om dubbel werk te voorkomen en geen inconsistente gegevens te creëren.
- **Vragen, opmerkingen, wensen of eisen m.b.t. gegevensbeheer (2) :**  
Tot nu is het gebruik van het program makkelijk en vanzelf sprekend, grafisch zo er nog het een en ander verbeterd kunnen worden.
- **Vragen, opmerkingen, wensen of eisen m.b.t. plannen (3) :**  
Het plannen dient op een andere manier te gebeuren: Graag zou ik per week nummer kunnen willen plannen, per dag (zoals het nu gaat) neemt te veel werk met zich mee en daardoor is het werk niet erg efficiënt. Efficiëntie is een van de belangrijkste zaken van het systeem.

Daarnaast zou het wenselijk zijn dat het plannings gedeelte aanroepbaar is vanuit het projecten gedeelte van het systeem.

## AANPASSINGEN NAAR AANLEIDING VAN DE WORKSHOP DEMONSTRATIE

### 1.

Binnen het projecten gedeelte zullen de opties met betrekking tot offertes worden verwijderd. Dit zal tevens worden doorgevoerd in de database, om zo geen onnodige gegevens op te slaan of lege velden te creëren.

### 2.

Het grafische gedeelte van het systeem dient te worden aangepast, dit zal dan ook later gebeuren tijdens een andere iteratie, voorlopig dient eerst de functionaliteit in orde te zijn. De functionaliteit is immers het belangrijkste van het systeem.

### 3.

De derde en laatste aanpassing die verricht dient te worden is tevens de gene die het meeste werk met zich mee brengt. Hier is de volgende oplossing voor bedacht.

Het overzicht zoals het er nu uitziet moet min of meer behouden blijven wel moeten er weeknummers aan toegevoegd worden

Het idee is als volgt:

#### OVERZICHT VAN DE MAAND JULI 2004

weeknr	MA	DI	WO	DO	VR	ZA	ZO
27	28	29	30	1	2	3	4
28	5	6	7	8	9	10	11

De opdrachtgever wil dus kunnen plannen op weeknummers (net zoals weeknummers in een agenda)

Kortom je voert een startweek in (bijvoorbeeld 46) en een eindweek (47)

en geeft daarbij aan op welke dagen (maandag, dinsdag, woensdag, donderdag, vrijdag)

Dit project door monteur x wordt uitgevoerd!

En het programma vult voor week 46 & 47 op de geselecteerde dag(en) het project en de monteur samen en werkt dit bij in de agenda

Wil men 1 week of zelfs 1 dag invullen voor een klus dan is de startweek hetzelfde als de eindweek en geef je dus bijvoorbeeld alleen maandag aan. (zo creëer je dus een 1 daagse klus)

Je krijgt dan iets in de vorm van dit schermpje

```
-----  
|  
| startweek: [weeknummer] |  
| eindweek: [weeknummer] |  
|  
| MA DI WO DO VRY |  
| o  o  o  o  o |  
|  
|-----|
```

Naar aanleiding van hierboven beschreven aanpassingen is er besloten om alvorens verder te gaan met de bouw van de pilot, eerst de definitiestudie en het bijbehorende documentatie van de pilot planningsysteem opnieuw te doorlopen om zo de documentatie en de werkzaamheden consistent te houden.

## **Bijlage 4: Workshop 2**

## WORKSHOP/ DEMONSTRATIE PROTOTYPE PLANNINGSSYSTEEM

Datum: 24-8-2004

Tijd: C.A 11:00

### AANWEZIGEN

Opdrachtgever:

M.C.J. Dee

Gebruikers:

G.C.Molenkamp, R.L.A. van Cappelle

Ontwikkelaar/programmeur: N.E. Roos

Welkom bij de tweede workshop/ demonstratie van het prototype planningssysteem. In deze workshop willen we een impressie geven van het planningssysteem. De workshop zal bestaan uit het uitvoeren van enkele opdrachten en het demonstreren van functionaliteiten. Daarnaast is er ruimte voor vragen, opmerkingen, en wensen en eisen. Door deze workshop kan, waar nodig, het project nog worden bijgestuurd.

## DEEL 1

### Opdracht 1.

Om met het planningssysteem te kunnen plannen, dienen er verschillende gegevens binnen het systeem bekend gemaakt te worden. Een groot gedeelte van dit systeem bestaat dan ook uit gegevensbeheer. Ter introductie van het planningssysteem zullen we enkele opdrachten uitvoeren met betrekking tot het gegevensbeheer.

### Opdracht 1.1

32. Start internet explorer op.
33. Surf naar het volgende adres: <http://fs02/test/bonanieuw/bonaplan.html>
34. Druk op de naam Bonaplan
35. Er verschijnt een login scherm, log in met de volgende gegevens  
Username: nicolai paswoord: nicolai
36. Druk in het linker frame op de knop gegevens beheer
37. (Demonstratie van verschillen bij inloggen per user soort)

**Vragen, opmerkingen, wensen of eisen m.b.t inlog gedeelte:**

.....  
.....

### Opdracht 1.2

Alle gegevensbeheer onderdelen werken op dezelfde manier, daarom zullen we niet alle schermen doorlopen.

38. U bevindt zich in het scherm gegevensbeheer druk op de link Projectleider beheren
39. U ziet een overzicht van de aanwezige projectleider. Druk op het ID om de detail gegevens te zien.
40. Druk op de link "projectleider pagina"
41. Druk op de link "Gegevensbeheer pagina"
42. Druk op de link nieuwe projectleider en voeg een nieuwe projectleider met de volgende gegevens toe  
Naam: Nicolai Roos  
Telefoonnummer: 079-3317731  
Na het toevoegen druk u op de link "projectleider pagina"
43. U ziet dat de projectleider is toegevoegd

**Vragen, opmerkingen, wensen of eisen m.b.t. toevoegen:**

Zie eind van dit document

44. Druk op het ID van de opdrachtgever die u zojuist heeft toegevoegd.
45. Druk op de knop wijzigen.
46. Wijzig de gegevens van de betreffende opdrachtgever en druk op wijzigen

47. Druk op link "opdrachtgever pagina"
48. Druk wederom op de zojuist gewijzigde opdrachtgever.
49. Druk op de link verwijderen en verwijder de opdrachtgever
50. Druk op OK

U heeft nu een projectleider, toegevoegd, gewijzigd en verwijderd.

**Vragen, opmerkingen, wensen of eisen m.b.t. gegevensbeheer:**

Zie eind van dit document

**DEEL 2:**

Deel 2 zal bestaan uit het toevoegen, wijzigen en verwijderen van een project  
We gaan er van uit dat u bent ingelogd en u zich bevindt in het keuzescherf.

## Opdracht 2.1

Projecten, is eigenlijk een vorm van gegevensbeheer, maar aangezien, dit een van de grotere onderdelen van het systeem omvat is deze gescheiden van de rest

51. U bevindt zich in het centrale keuze scherm druk op de link projecten
52. U ziet een overzicht van de aanwezige projecten. Druk op het ID om de detail gegevens te zien
53. Druk vervolgens op de link terug.
54. Druk op de link nieuw project en voeg een nieuw project met de volgende gegevens toe  
Projectnummer: 20043125  
Werknummerklant: 123-5696635  
Straat\_projectnaam: Zwaneburgerdijk  
Plaats: Zwanenburg  
Huisnummer: 12  
Datum\_aanvraag: 19-07-2004  
Werkttype: Mutatie  
Na het toevoegen druk u op de link "Projecten pagina"
55. U ziet dat het project is toegevoegd

**Vragen, opmerkingen, wensen of eisen m.b.t. toevoegen:**

Zie eind van dit document

56. Druk op het id van het project die u zojuist heeft toegevoegd.
57. Druk op de knop wijzigen.
58. Wijzig de gegevens van het betreffende project en druk op wijzigen
59. Druk op "projecten pagina"
60. Druk wederom op de zojuist gewijzigde project.
61. Druk op de link verwijderen en verwijder de project.
62. Druk op OK

U heeft nu een project, toegevoegd, gewijzigd en verwijderd.

**Vragen, opmerkingen, wensen of eisen m.b.t. gegevensbeheer:**

Zie eind van dit document

**DEEL 3:**

Deel 3 zal bestaan uit het plannen van monteurs en projecten een van de belangrijke zaken van het project. In vergelijking met de vorige workshop is het planningsgedeelte totaal veranderd. Van het plannings gedeelte zal een uitgebreide demonstratie gegeven worden . Natuurlijk is er ruimte voor vragen, opmerkingen, wensen en eisen.

**Vragen, opmerkingen, wensen of eisen m.b.t. plannen:**

Zie eind van dit document



**DEEL 4:**

Discussie / brainstormsessie m.b.t tot het opdrachtgevers gedeelte en de overzichten

**Vragen, opmerkingen, wensen of eisen m.b.t. plannen:**

Zie eind van dit document

**EINDE VAN WORKSHOP/ DEMONSTRATIE BonaPlan**

Hieronder volgen de wensen, eisen en vragen en opmerkingen die tijdens deze workshop/demonstratie naar voren zijn gekomen en de veranderingen die deze met zich mee brengen.

(zie volgende pagina)

1. Benoeming loginnaam / paswoord vervangen door gebruiker / wachtwoord;
2. De positie van de login-gegevens is slecht leesbaar, e.e.a. aanpassen;
3. Foutmelding bij foutieve login slecht leesbaar, zou grote pop-up o.d. moeten worden;
4. Bij keuzeschermen de tekst "wat wilt u doen" verwijderen, staat beetje oenig spreekt nl. voor zich;
5. In de tabel "projectleider" toevoegen veld "telefoonnummer mobiel";
6. Te vaak "klikken" bij aanmelden;
7. Wijzigen in de tabel "soort monteur" "normaal" in "CV";
8. In de tabel "opdrachtgever" de volgende velden toevoegen; Adres, Postcode, Plaats, telefoonnummer;
9. Bij onderhouden gebruikers de tekst "systeem" verwijderen;
10. Bij onderhoud de BUTTON "wijzigen" wijzigen in "opslaan";
- 11. Bij onderhoud (met name project) chronologisch binnen het scherm "doortabben";**
12. In de volgende tabellen de keuze "n.n.b." toevoegen; "offerte uitbrengen", "opname nodig", "bouw nodig", "elektra aanbrengen";
- 13. Bij het plannen van projecten moet de listbox project vervangen worden door een zoekfunctie waar bij gekozen kan worden voor "op nummer", "op naam", "op deel van naam"(bevat);**
14. Display in alle schermen de datum van vandaag + weeknummer;
15. Bij agenda detail tussenstap verwijderen, scheelt 1x klikken;
- 16. Plannen in twee dagdelen (ochtend en middag);**
17. Aanroepen kalender per monteur in het planscherm;
18. Binnen een lopende planning monteur vervangen vanaf datum..;
19. Binnen een lopende planning monteur verwijderen vanaf datum;
20. Extra rapport; kalender per werksoort;
- 21. Rapport "balkenschema".**

Aanvullende wens;

In het scherm onderhoud projecten de <tab> functie vervangen door een <enter> functie.

#### **Prioriteiten;**

**Zeer hoog;**

**Hoog;**

Normaal.

## **Bijlage 5: Interview**

Interviewer: Nicolai Roos  
Geïnterviewde: G.C. Molenkamp

Betreft: Eisen aan BonaPlan

### **Wat verwachten jullie van BonaPlan?**

BonaPlan zal een on-line planningssysteem moeten worden. Dit planningssysteem dient ervoor te zorgen dat het plannen efficiënter gaat en minder werk in beslag neemt. Daarnaast dient BonaPlan een service zijn naar de opdrachtgevers. Een online gedeelte speciaal voor deze opdrachtgevers moet een onderdeel van BonaPlan zijn. Binnen BonaPlan moeten ook overzichten aanwezig waardoor in één oogopslag informatie kan worden bekeken, deze overzichten zullen in de loop van het project worden omschreven (is nog niet bekend).

### **Welke planningen dient BonaPlan te vervangen?**

BonaPlan dient de huidige planningen te vervangen te weten:

Planning tekenkamer indien project bestaande bouw > 1 woning of nieuwbouw;

Planning mutatie indien individuele woning complete CV;

Planning verketelingen via bedrijfsbureau indien verketeling of slangen vervangen.:

Planlijst 2004 (in 2005 is dat planlijst 2005);

Planlijst Ronald (verketelingen);

Balkenplanning.

### **Zijn er eisen met betrekking tot het uiterlijk?**

Het uiterlijk van het systeem dient aan te sluiten op de huisstijl van Bonarius en moet voor zichzelf te spreken

### **Wat vind u een acceptabele tijd voor het laten berekenen van acties door het systeem.?**

Een maximale tijd van 6 seconden vind ik acceptabel.

Moeten er binnen het systeem verschillende gebruiker met verschillende rechter aanwezig zijn?

Ja binnen het systeem moeten verschillende gebruikers zijn

Beheerder

Planner

Viewer (persoon binnen Bonarius die alle overzichten kan zien)

Opdrachtgever (welke alleen in het opdrachtgevers gedeelte mag komen).

### **Welke gegevens dienen er binnen BonaPlan geregistreerd te kunnen worden?**

Naast dat natuurlijk projecten geregistreerd en gepland moeten kunnen worden. Zal BonaPlan nog verschillende andere gegevens moeten registreren.

- Monteurs
- Systeem Gebruikers
- Opdrachtgevers
- Uitvoerder
- Ketels
- Urgenties
- Projectleiders
- Radiatoren
- Werktypes
- Statussen

## **Bijlage 6: Wekelijkse contactmomenten**



## **EXTERNE BIJLAGEN**

- 1. Definitiestudie**
- 2. Rapport Pilotontwikkelplan Database**
- 3. Rapport Pilotontwikkelplan Planningssysteem**
- 4. Rapport Pilotontwikkelplan Gegevensbeheer**
- 5. Rapport Pilotontwikkelplan Overzichten**
- 6. Operationeel maken BonaPlan**
- 7. Testplan**