

Bachelorproject IN3405
Workflow-management in een Content Managementsysteem

Rob Ruigrok
R.J.Ruigrok@student.tudelft.nl
1371746

Jan Pieter Mars
J.P.Mars@student.tudelft.nl
1381075

7 december 2011



ContentPower
informatiebeheersystemen

Voorwoord

Dit is het verslag van het bachelorproject van Jan Pieter Mars en Rob Ruigrok, studenten Technische Informatica aan de Technische Universiteit in Delft. Dit verslag beschrijft het proces dat is doorlopen tijdens het werken aan een stageopdracht bij Content Power in Delft.

Van september tot december 2011 zijn wij bezig geweest met het onderzoeken, ontwerpen en ontwikkelen van een workflow-management systeem in het Content Management Systeem van het bedrijf. Het resulterende product is opgeleverd aan Evenementenbureau Delft, een van de klanten van Content Power. Dit bureau houdt zich bezig met het organiseren van activiteiten en festiviteiten in Delft. Het organiseren van activiteiten is een interactief proces waarbij medewerkers, fondsen en projectleiders betrokken zijn. Voor Evenementenbureau Delft is een workflow-model opgesteld voor het proces van activiteiten organiseren. Eind november heeft van dit model een gebruikerstest met medewerkers van Evenementenbureau Delft plaatsgevonden, en bleek dat het product zeer veel potentie heeft voor een succesvolle toepassing in de praktijk.

Tijdens dit project zijn wij vanuit de TU Delft begeleid door drs. Peter van Nieuwenhuizen, en vanuit Content Power begeleid door ir. Willem van Valkenburg en Jan Niemantsverdriet. Via deze weg willen wij hen bedanken voor de goede begeleiding en adviezen gedurende het project. Verder willen wij via deze weg Susanne Dix en David Lansen van Evenementenbureau Delft bedanken voor het testen van het product. Wij hopen dat gebruik van het product in praktijk succesvol zal zijn. Als laatste willen wij Kees Rodenburg en Ed Rademaker namens VOC (VakOpleiding Carrosseriebedrijf), en Marcel Bon namens NEVI (NEderlandse Vereniging voor Inkopers) bedanken voor hun tijd tijdens het interview. Het heeft ons veel inzicht gegeven in de problemen die in de praktijk een rol spelen, zodat wij daar in het ontwerp rekening mee konden houden.

Delft, december 2011

Samenvatting

Content Power heeft een bestaand Content Managementsysteem (CMS). Dit CMS is een framework waarin modules zijn geïmplementeerd. Het doel van de opdracht is om in het CMS van Content Power een framework en bijbehorende module te bouwen die workflow-management ondersteunt. Workflow-management biedt klanten de mogelijkheid om (sturing in) bedrijfsprocessen te automatiseren, en routinetaken van het personeel uit handen te nemen. Een bedrijfsproces is een keten van activiteiten die een resultaat opleveren. Dat kan een primair proces zijn dat rechtstreeks een product of dienst voor de klant oplevert, maar ook een intern proces dat andere bedrijfsprocessen direct of indirect ondersteunt. Daarvoor moet het CMS zelf kunnen reageren op acties, en nieuwe acties daarop kunnen ondernemen.

Processen in een workflow bestaan uit losse activiteiten. Deze activiteiten kunnen afhankelijk van elkaar zijn. Om de afhankelijke activiteiten met elkaar te koppelen, zijn er trigger-, verwerkings- en communicatiemethodes, die taken uitvoeren en beslissingen nemen om het proces te sturen zodat de koppeling vloeiend verloopt. Via e-mail worden gebruikers betrokken in de activiteiten. Deze e-mails zijn gericht gestuurd naar de verantwoordelijke personen voor de activiteit, en bevatten overzichten, herinneringen, vragen voor nieuwe informatie, of vragen van goed- of afkeuring. De beïnvloeding van processen hangt af van de gebruikers, maar uitvoering en sturing van processen verloopt via het workflow-managementsysteem. Het resultaat van het project is een framework dat bovenstaande functionaliteit volledig ondersteunt, en is getest door zowel medewerkers als klanten van Content Power.

De uitvoering van dit project verliep gefaseerd en volgens het sashimi-model. Het sashimi-model is een doorontwikkeling van het watervalmodel, waarbij fases elkaar gedeeltelijk kunnen overlappen. Meer informatie over het sashimi-model is te vinden in appendix A. Hieronder volgt een opsomming van de belangrijkste fases en bijbehorende deliverables.

- De originele opdrachtomschrijving van Content Power is opgenomen in appendix A. Informatie over de werkwijze van Content Power en over de workflow-theorie is te vinden in sectie 2, en in het oriëntatieverslag (appendix B).
- Documentatie van het systeem, zowel voor gebruikers (gebruiken van de module) als voor het bedrijf (technische details en verdere ontwikkeling), is opgenomen in sectie 3.
- De planning voor uitvoering van het project en de doorlopen fases wordt toegelicht in het Plan van Aanpak in appendix C. Een evaluatie van de planning is opgenomen in sectie 4.
- De eisen aan het workflow-management systeem zijn in de analysefase uitgedacht. Deze eisen zijn vastgesteld door de opgedane kennis uit voorgaande fases. Aan de hand van de interviews zijn usecases opgesteld. De resultaten zijn verwerkt in het Requirements and Analysis Document in appendix D.
- Het ontwerp voor het framework is gespecificeerd in het Technical Design Document (appendix E). Deze bevat een uitgebreid klassen- en databasemodel. Gedurende de implementatie is aan dit ontwerp vastgehouden. Een verslag van het verloop van de implementatie is beschreven in de sectie met evaluaties van fases, sectie 5.
- Softwarekwaliteit is zeer belangrijk, en testen is een belangrijk onderdeel in een softwareontwikkeltraject. Het testplan beschrijft methodieken en procedures die zijn uitgevoerd om de softwarekwaliteit te waarborgen, door middel van tests. Deze worden uitgebreid beschreven in appendices G en F. De resultaten van deze tests staan beschreven in sectie 5.5.
- Gedurende de implementatiefase is de code opgestuurd naar Software Improvement Group (SIG) voor een analyse van de codekwaliteit. Het rapport met aanbevelingen voor het verbeteren van de code is te vinden in appendix H. Onze feedback op deze aanbevelingen wordt beschreven in sectie 5.4.3.

In sectie 6 (ervaringen en aanbevelingen) volgt een opsomming van aanbevelingen en tips voor Content Power. Sectie 7 sluit af met een conclusie over het gehele project.

Inhoudsopgave

1	Introductie	5
2	Opdrachtschrijving en analyses	5
2.1	Bedrijfsprofiel	5
2.2	Probleemstelling	5
2.3	Oriëntatie en analyse	6
2.4	Deliverables	6
3	Documentatie	7
3.1	Documentatie voor eindgebruikers	7
3.1.1	Inleiding	7
3.1.2	Het CMS van Content Power	7
3.1.3	De workflow-module	8
3.1.4	Functionaliteiten	9
3.1.5	Samenvatting	9
3.2	Documentatie voor Content Power	10
3.2.1	Gebruik van de workflow-module	10
3.2.2	Module workflow-processen	10
3.2.3	Nieuwe methode aanmaken	13
3.2.4	Module instanties processen	14
3.2.5	Testsuite	15
4	Werkproces	16
5	Fases van het softwareontwikkeltraject	18
5.1	Oriëntatiefase	18
5.2	Analysefase	18
5.3	Ontwerpfase	18
5.3.1	Functionele eisen	19
5.3.2	Niet-functionele eisen	19
5.4	Implementatiefase	19
5.4.1	Databasemodel	20
5.4.2	Programmacode	20
5.4.3	Codebeoordeling door de Software Improvement Group	20
5.5	Testfase	21
5.5.1	Resultaten van geautomatiseerde tests	22
5.5.2	Resultaten van gebruikerstest	22
5.5.3	Verbeteringen in het workflow-proces	22
5.5.4	Interface	23
5.5.5	Quotes	23
5.6	Afrondingsfase	23
6	Ervaringen en aanbevelingen	24
7	Conclusie	25
	APPENDICES	26
A	Opdrachtschrijving	27
B	Oriëntatieverslag	31
C	Plan van Aanpak	48

D Requirements and Analysis Document	62
E Technical Design and Specification Document	86
F Testplan	104
G Gebruikerstest	116
H Aanbevelingen van Software Improvement Group	122

1 Introductie

Dit verslag beschrijft het verloop van het project en het eindresultaat. In sectie 2 wordt beschreven wat de opdracht inhoudt, worden definities gegeven, en een beknopte samenvatting van ons oriënterend onderzoek. De mogelijkheden van het eindproduct worden beschreven in de documentatie in sectie 3. In sectie 4 wordt per week het werkproces beschreven. Hier worden onder andere de gebruikte resources, de manier waarop wij te werk zijn gegaan, en de gebruikte programmatuur beschreven. In sectie 5 worden alle doorlopen fases geëvalueerd. In deze secties wordt geëvalueerd in hoeverre de concepten en ontwerpen zoals deze zijn opgesteld in het Requirements Analysis Document (appendix D) en Technical Design Document (appendix E) in het eindproduct zijn verwerkt. Ook wordt het eindresultaat geëvalueerd, zowel de technische aspecten in de vorm van diagrammen en modellen, als de front-end in de vorm van screenshots. Ook worden de resultaten van de tests zoals deze zijn beschreven in het Testplan geanalyseerd. Daarna volgt een sectie die ervaringen en aanbevelingen bevat voor toekomstige ontwikkelingen door Content Power. Deze zijn opgenomen in sectie 6). Er wordt afgesloten met een conclusie in sectie 7. Daarna volgt een aantal appendices. Deze appendices zijn geschreven tijdens de verschillende fasen van het software-ontwikkeltraject. In dit verslag zal regelmatig worden gerefereerd naar informatie die in de appendices uitgebreid is beschreven.

2 Opdrachtomschrijving en analyses

Deze sectie bevat de opdracht en probleemstelling, en informatie over Content Power, het bedrijf waar de opdracht wordt uitgevoerd. Ook wordt de aanpak van de oriëntatiefase beschreven. Deze sectie sluit af met een overzicht van deliverables.

2.1 Bedrijfsprofiel

Content Power is opgericht in 2001, het is een bedrijf dat is gespecialiseerd in het ontwikkelen van websites. Op dit moment werkt Content Power met zes personen aan het realiseren van websites voor klanten. De geleverde websites maken gebruik van het door Content Power zelf ontwikkelde Content Managementsysteem (CMS). Dit CMS is de kern van alle geleverde websites. Het CMS vormt een framework om de wensen van de klant te realiseren, en is na jaren van ontwikkeling zeer uitgebreid. Veel klanten van Content Power gebruiken de website als onderdeel van het bedrijfsproces. Voorraadbeheer, webwinkels, online inschrijvingen, ledenadministratie, nieuwsbrieven, intra- en extranetten zijn al regelmatig geïmplementeerd.

2.2 Probleemstelling

Medewerkers van Content Power merken regelmatig op dat klanten behoefte hebben aan een systeem dat zoveel mogelijk taken uit handen kan nemen. Vooral routinetaken kosten in praktijk veel tijd, omdat deze taken tot stand komen door samenwerking van verschillende personen, en zeer afhankelijk zijn van communicatie, en gestuurd diend te worden. Deze routinetaken zijn voorbeelden van workflow-processen. Binnen het bestaande framework van Content Power zal daarvoor een framework en datamodel worden gemaakt waarin workflow-processen kunnen worden gemodelleerd. Het framework wordt vooral toegepast op eenvoudigere processen die dichtbij het beheren van een website of webwinkel liggen. Binnen het CMS van Content Power moet een module komen waarin de processen kunnen worden samengesteld. De module zal voorlopig alleen door Content Power zelf worden gebruikt. Het samenstellen van processen moet zonder fouten kunnen, maar mag enige technische kennis vereisen.

Onderzoek heeft uitgewezen wat de belangrijkste triggers zijn. Triggers worden gebruikt om processen te starten of weer te hervatten. Sommige delen van administratieve processen kunnen worden geautomatiseerd. Voor de eenvoudige handelingen (verwijderen van items, wijzigen van een waarde, etc.) moeten verwerkingsmethodes in het framework en de module worden opgenomen. Uit onderzoek worden de belangrijkste communicatiemiddelen (mail, sms, social media,

etc.) bepaald. Het framework en de module worden ook voorzien van deze communicatiemiddelen. Deze middelen worden gebruikt om, waar nodig, een verantwoordelijke weer bij het proces te betrekken.

In appendix A is de officiële opdrachtomschrijving opgenomen.

2.3 Oriëntatie en analyse

Als start van het project is eerst uitgebreid literatuuronderzoek uitgevoerd naar workflows, workflow-management en andere aan workflow gerelateerde begrippen. In appendix B wordt uitgebreid op dit onderzoek ingegaan.

Als extra analyse zijn de wensen van (potentiële) klanten van Content Power in kaart gebracht. Daarvoor zijn een aantal interviews gehouden met bestaande klanten. Het doel van de interviews was om antwoorden te krijgen op de volgende vragen:

1. Welke administratieve processen vinden er op dit moment bij de bedrijven plaats?
2. Hoe zien deze processen er uit?
3. Waardoor wordt het proces gestart?
4. In welke stappen verloopt het proces?
5. Wie zijn er betrokken bij het proces?
6. Wat is het einddoel van het proces?
7. Welke communicatiemiddelen worden er gebruikt?
8. Wordt het als een voordeel gezien als een systeem deze processen kan begeleiden?

Alle drie de interviews hebben nieuwe inzichten opgeleverd. Op een punt kwamen alle drie de interviews overeen: een groot probleem binnen een organisatie is vaak de communicatie tussen de eigen werknemers. Processen die worden doorlopen in een groot aantal stapjes, verlopen vaak inefficiënt omdat werknemers deadlines vergeten, of laat reageren om een taak uit te voeren. Verder is het gebrek aan transparantie van bedrijfsprocessen vaak een groot probleem. De status van een proces is vaak onbekend, soms zelfs voor de verantwoordelijke persoon van het proces. Voor al deze problemen zou een workflow-managementsysteem uitkomst kunnen bieden. Een belangrijke eis die tijdens de interviews naar voren kwam, was dat het niet voor een extra systeem naast andere systemen in het bedrijf zorgt, maar geïntegreerd moet worden in de bestaande systemen. Omdat klanten al gebruik maken van het CMS voor het beheren van de website, kan aan deze eis worden voldaan.

2.4 Deliverables

De opgeleverde deliverables die zijn beschreven in de opdrachtomschrijving (appendix A) zijn:

- Projectplan met planning. Deze is terug te vinden in het Plan van Aanpak, zie appendix C.
- Basisontwerp met usecases. Het ontwerp is opgenomen in het Requirements Analysis Document (appendix D) en Technical Design Document (appendix E).
- Framework met uitbreidingen (programmeerwerk). Deze code is verweven in de Subversion-repository van het CMS van Content Power.
- Testplan en resultaten. Het testplan is opgenomen in appendix F, de testresultaten volgen later in dit document in sectie 5.5.2.

- Documentatie voor Content Power. Deze documentatie is op een aantal plekken aanwezig. In de workflow-module is achter formulierelden extra help-informatie toegevoegd over de betekenis van het veld. Verder is er op de interne wiki van Content Power een pagina aangemaakt met technische uitleg over de werking van de module, interfaces, en het uitbreiden van het systeem. Verder is er ook documentatie in de code aangebracht, om de methodes en klassen te specificeren.
- Documentatie voor de gebruiker wordt opgenomen in de officiële help-site van CMS4. Deze is te raadplegen op <http://help.contentpower.nl>. Omdat de workflow-module voorlopig alleen door Content Power zal worden gebruikt, is deze documentatie erg beknopt. Klanten zullen alleen binnen een op maat gemaakte oplossing merken dat er op de achtergrond workflow-management zal plaatsvinden. De documentatie zal daarom alleen de mogelijkheden en voordelen beschrijven die workflow-management via het CMS biedt.

Alle gevraagde deliverables zijn opgeleverd, en voor zover mogelijk opgenomen in dit document als appendices.

3 Documentatie

Deze sectie bevat de documentatie. Er is zowel documentatie geschreven voor Content Power, als documentatie voor de eindgebruikers.

3.1 Documentatie voor eindgebruikers

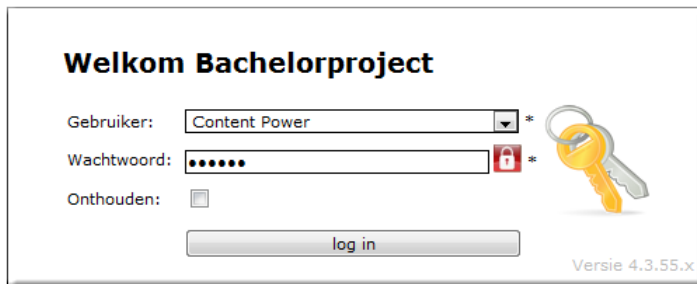
Deze sectie is bedoeld voor de eindgebruikers van de workflow-module. Eindgebruikers zijn altijd klanten van Content Power. In dit document wordt beschreven wat de mogelijkheden zijn van de workflow-module en waar het allemaal voor gebruikt kan worden. De workflow-module kan een grote tijds- en kostenreductie bij bedrijfsprocessen van klanten opleveren. Sectie 3.1.1 - 3.1.5 kan opgenomen worden op de help-site van CMS4 van contentpower: help.contentpower.nl

3.1.1 Inleiding

Het Nederlandse woord voor workflow is "werkstroom", het is een opeenvolging van onderling verbonden stappen. Elk bedrijfsproces kan in verschillende stappen opgedeeld worden. Elke stap bestaat uit een of meer verschillende activiteiten, en deze kunnen vaak automatisch gestuurd worden. De workflow-module die is geïntegreerd in het CMS, biedt de mogelijkheid voor deze automatisering. Door met behulp van de workflow-module bedrijfsprocessen op te delen in verschillende kleine stappen, ontstaat een beter inzicht in de werking van het bedrijfsproces. Daarnaast biedt de workflow-module ook een duidelijk overzicht van de statussen van verschillende processen. Per activiteit is te zien wat de status is, het is daarmee mogelijk om direct te zien welke activiteit nog moet gebeuren. Alle processen kunnen worden getoond in een overzicht. Dit is zeer handig voor projectmanagers binnen een bedrijf.

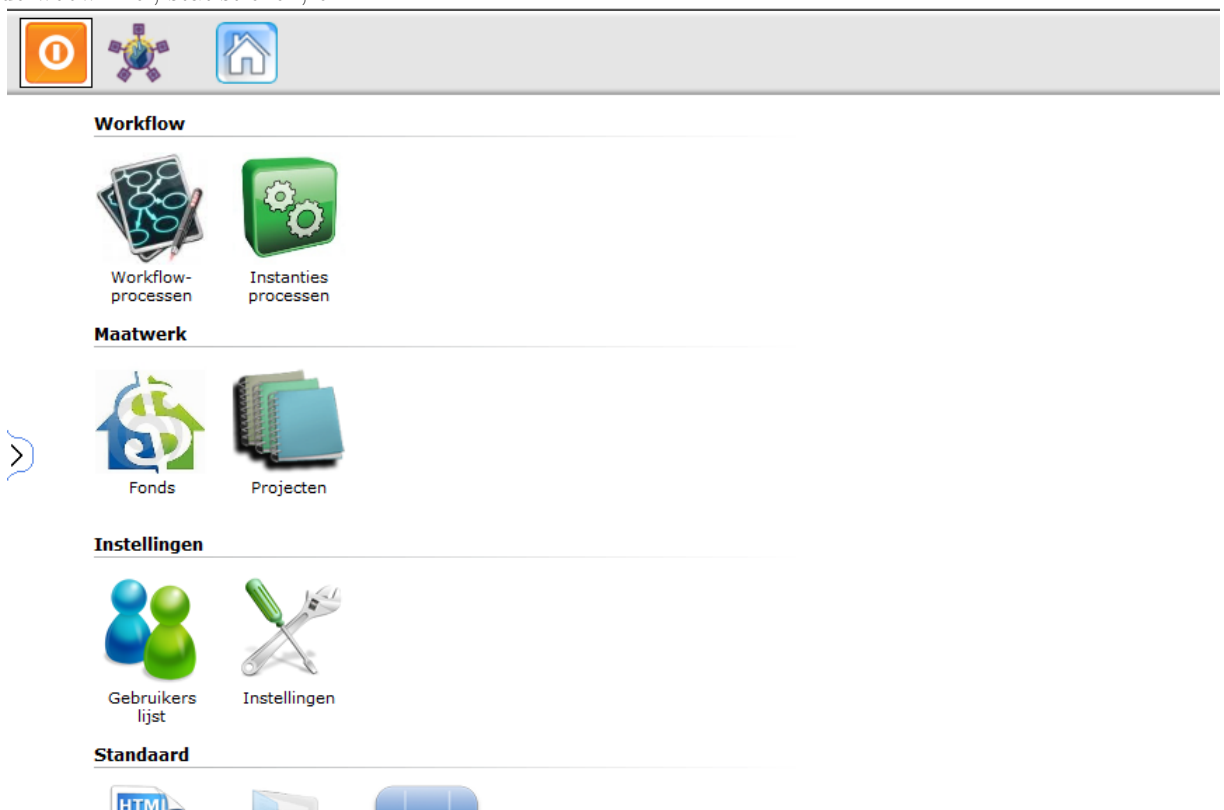
3.1.2 Het CMS van Content Power

Het CMS is het beheersysteem dat klanten gebruiken om alle informatie op de website te beheren (zie figuur 1 en 2). In dit systeem worden alleen modules getoond die voor een ingelogde klant



Figuur 1: Het inlogscherf van het CMS.

van toepassing zijn, zoals bijvoorbeeld het bewerken van pagina's, het plaatsen van producten in de webwinkel, statistieken, enz.

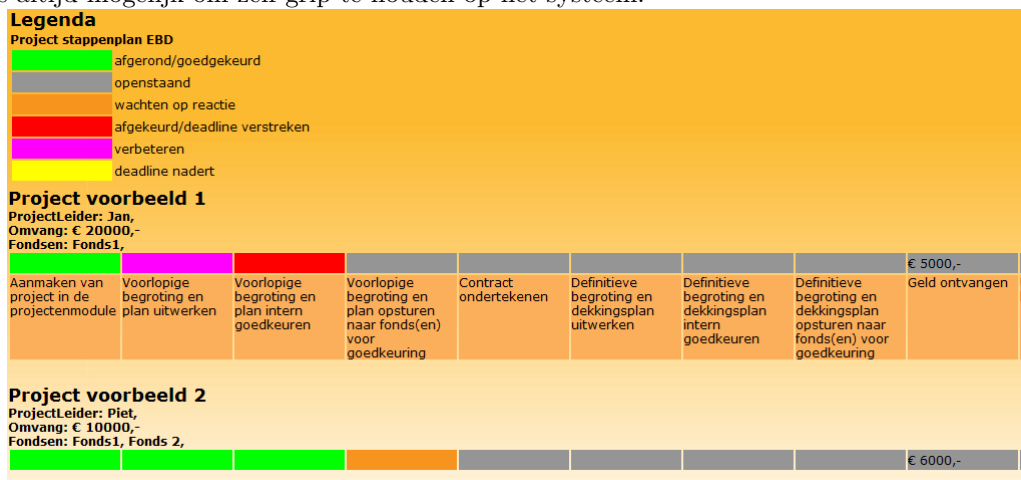


Figuur 2: Het hoofdscherf van het CMS na het inloggen van een gebruiker

3.1.3 De workflow-module

De workflow-module is geïntegreerd in het CMS van Content Power en kan daardoor eenvoudig gebruikt worden door huidige klanten van Content Power. De workflow-module is niet direct zichtbaar voor klanten, omdat Content Power aanvankelijk zelf het workflow-proces modelleert en onderhoudt. Mogelijk komt deze module in een later stadium beschikbaar voor gebruikers. Afhankelijk van het bedrijfsproces worden bepaalde maatwerkmodules getoond. Hier wordt als voorbeeld een module 'projecten' gebruikt, voor een evenementenbureau. Via de module projecten kan een project met een bepaalde naam, verantwoordelijken, deadlines en andere wensen worden aangemaakt. Normaalgesproken is het niet nodig, maar het is handmatig mogelijk om een project te beïnvloeden door vinkjes bij bepaalde activiteiten te zetten en statussen te wijzigen. Het blijft

dus altijd mogelijk om zelf grip te houden op het systeem.



Figuur 3: Voorbeeld overzicht met verschillende projecten

Na het aanmaken van een proces of project verschijnen alle stappen met bijbehorende status direct in een speciaal op maat gemaakt procesoverzicht. Dit overzicht is een webpagina waarvoor ingelogd moet worden. Op deze webpagina worden alle lopende projecten /processen in een enkel overzicht getoond (zie figuur 3).

3.1.4 Functionaliteiten

De workflow-module biedt tal van mogelijkheden en kan naar wens uitgebreid worden. Om een idee te geven wat de mogelijkheden zijn, worden hieronder een aantal bestaande toepassingen als voorbeeld gegeven. Communicatie is een van de belangrijkste onderdelen binnen een bedrijf, want alleen door samenwerking kunnen bedrijfsprocessen goed verlopen. Om de communicatie binnen een bedrijf te ondersteunen en te bevorderen biedt de workflow-module verschillende communicatiemogelijkheden. De belangrijkste hiervan is de e-mail, maar het kan ook een SMS of een Twitter-bericht zijn. Indien er een status gewijzigd wordt, worden de juiste personen direct geïnformeerd. Zo kunnen de juiste werknemers automatisch een e-mail krijgen als er bijvoorbeeld een stap is afgerond, afgekeurd of als er een deadline nadert/verstreken is. Hierdoor is iedereen direct op de hoogte van wat er speelt en weten projectmanagers wat er nog moet gebeuren. Aan een e-mail kunnen links toegevoegd worden, waarmee iets goedgekeurd kan worden of waarbij aangegeven kan worden dat iets afgerond is. Indien er bijvoorbeeld op goedgekeurd in een e-mail geklikt wordt, wordt er automatisch een e-mail verstuurd naar de juiste persoon, met de melding dat het bijbehorende project is goedgekeurd en een vervolgstap ondernomen kan worden. Wanneer op een link wordt geklikt, wordt deze link geopend in de browser in een eenvoudige interface, met feedback wat de klik tot gevolg heeft gehad. Dit om onduidelijkheden te voorkomen (is mijn klik wel aangekomen?). Veel van de functionaliteiten zijn voor de gebruikers niet direct zichtbaar omdat deze functionaliteiten op de achtergrond werken. Een voorbeeld hiervan zijn verwerkingsmethodes die als reactie op een bepaalde wijziging, bijvoorbeeld een statuswijziging, een reeks activiteiten uit het bedrijfsproces automatisch uitvoeren en de feedback van deze acties weer in een e-mail terugcommuniceren. Ook kan er eenvoudig een uitbreiding worden gemaakt waarmee afspraken of deadlines in de agenda's van werknemers toegevoegd worden. Hierdoor is de agenda van elke werknemer altijd actueel en is iedereen binnen een organisatie op de hoogte van deadlines.

3.1.5 Samenvatting

De workflow-module is voor bestaande klanten eenvoudig te gebruiken doordat het geïntegreerd is in het huidige CMS van de klant. De workflow-module kan u meer inzicht geven in de werking

en het verloop van alle bedrijfsprocessen. Daarnaast biedt het uitgebreide ondersteuning voor de communicatie tussen de verschillende afdelingen en medewerkers en kan het een hoop werk automatiseren. Dit alles levert u een hoop tijdswinst en efficiëntie op wat een hoge kostenbesparing kan opleveren. Daarnaast zullen er ook minder irritaties binnen uw bedrijf zijn doordat de communicatie beter verloopt en er daardoor minder miscommunicaties zullen zijn.

3.2 Documentatie voor Content Power

De documentatie voor Content Power is toegevoegd op de gebruikelijke plekken waar Content Power documentatie toevoegt.

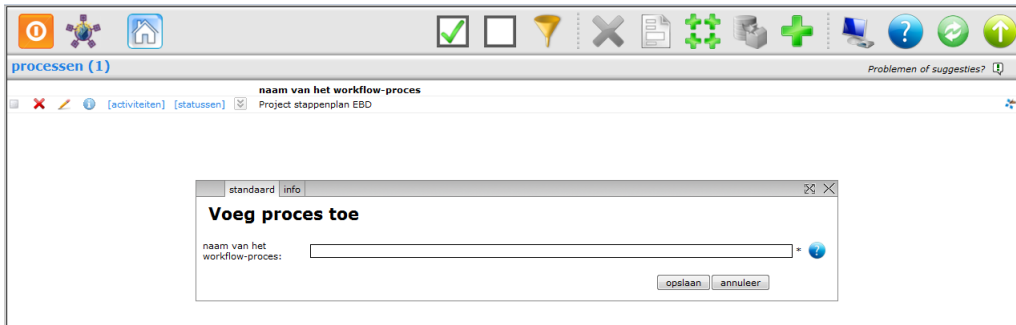
- **Helpknoppen achter formulervelden in CMS** - Deze helpfunctionaliteit is rechtstreeks geïntegreerd in het CMS, en is te gebruiken in alle CMS modules. Deze knoppen zijn vooral bedoeld om snel details over het gebruik van formulieren in het CMS op te vragen. Sommige formulieren, bijvoorbeeld het communicatiemiddel e-mail hebben de mogelijkheid om via een speciale syntaxis speciale waarden in e-mails op te nemen, zoals linkjes met ja/nee-vragen, waarden uit objecten, deadlines plus / minus een aantal dagen, enzovoort. Het snel kunnen raadplegen hoe deze functionaliteit gebruikt kan worden, is zeer handig, en daarom opgenomen in de helpknoppen.
- **PhpDoc Documentatie** - In alle bestanden wordt documentatie bijgehouden. Deze documentatie omvat een beschrijving van de klasse, vervolgens een logboek met wijzigingen per datum, naam en versienummer. Daarna begint de daadwerkelijke klasse, met daarin commentaar bij de velden en methodes (doel van methode, specificatie van parameters, return-waarde, typen, auteur en datum waarop de methode is aangemaakt). Deze documentatie is vooral bedoeld voor programmeurs, zodat zij uitbreidingen via de interfaces, en verbeteringen aan het framework kunnen uitvoeren.
- **Intranet Wiki** - Documentatie over de werking van het framework, het aanmaken van processen, activiteiten, triggers, verwerkings- en communicatiemethodes, mogelijkheden en beperkingen van de workflow-module worden uitgebreid beschreven op de Intranet Wiki. De inhoud van de pagina op de wiki op het moment van schrijven is in sectie 3.2.1 - 3.2.5 opgenomen.

3.2.1 Gebruik van de workflow-module

In het hoofdscherm van het CMS zijn er twee modules bijgekomen onder het kopje workflow (zie figuur 2). In de module 'workflow-processen' kunnen alle verschillende workflow-processen gemodelleerd worden. In de module 'Instanties processen', zijn alle instanties te zien van actieve processen met alle huidige statussen.

3.2.2 Module workflow-processen

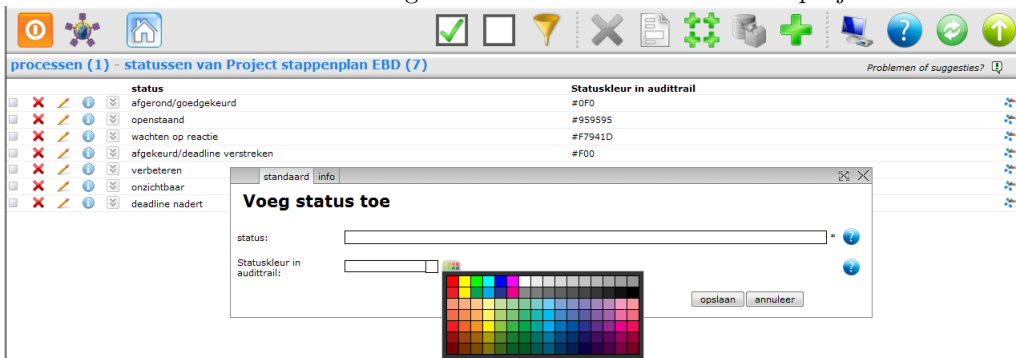
In deze module zijn eerst alle workflow-processen te zien. Via de + knop kan er een nieuw workflow-proces toegevoegd worden (zie figuur 4). Geef dit proces een naam die het bedrijfsproces beschrijft.



Figuur 4: Voeg nieuw proces toe in de module workflow-processen

Vervolgens wordt er een nieuw proces toegevoegd met deze naam, en kan er doorgedrukt worden naar de activiteiten en de statussen van dit proces.

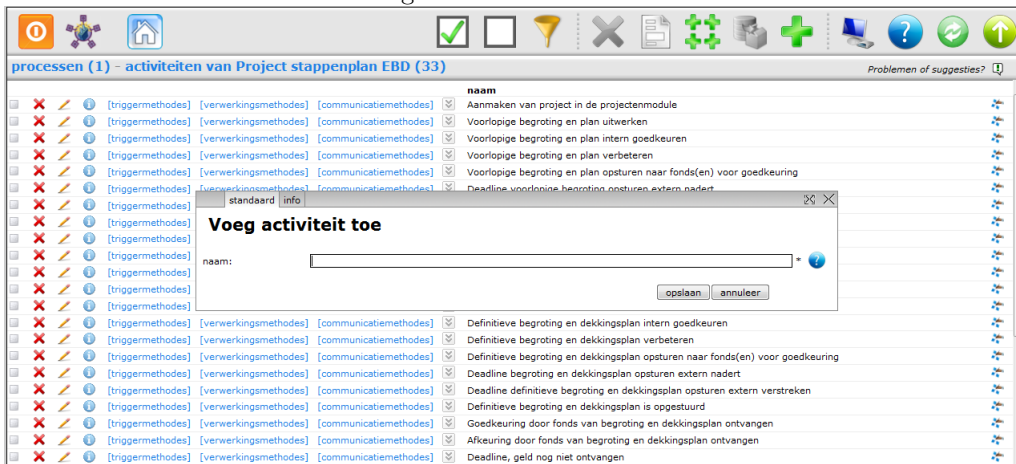
Via statussen kunnen er verschillende statussen toegevoegd worden die nodig zijn voor de activiteiten. Via de + knop kan er een status toegevoegd worden (zie figuur 5). Elke status heeft een naam en een kleur. Deze kleur wordt gebruikt in statusoverzichten van projecten.



Figuur 5: Verschillende statussen toevoegen aan projecten

Er is ook een status 'onzichtbaar' toegevoegd. Deze status wordt gegeven aan alle activiteiten die niet zichtbaar hoeven te zijn in procesoverzichten voor de klant.

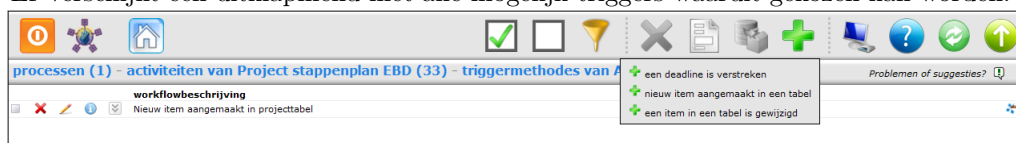
Terug bij de processen kan er doorgedrukt worden naar de verschillende activiteiten. Hier kan weer een activiteit toegevoegd worden door op de + knop te drukken (zie figuur 6). Hierbij is ook alleen maar de naam van een activiteit nodig.



Figuur 6: Voeg activiteit toe

Bij elke activiteit kan er weer doorgelikt worden naar zogenaamde methodes. Er zijn drie verschillende soorten methodes: trigger-, verwerkings- en communicatiemethodes. Nadat er op triggermethodes geklikt is, verschijnt een overzicht van alle triggers die toegevoegd zijn voor de betreffende activiteit.

Door op de + knop te drukken is het mogelijk om nieuwe triggers toe te voegen (zie figuur 7 en 8). Er verschijnt een uitklapmenu met alle mogelijk triggers waaruit gekozen kan worden.

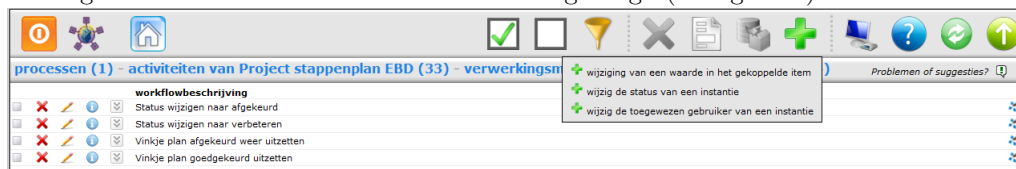


Figuur 7: Voeg een nieuwe triggermethode toe



Figuur 8: Voeg een triggermethode toe die 'afgaat' als er een veld is gewijzigd van het gekoppelde object.

Op een vergelijkbare manier als het toevoegen en wijzigen van triggermethodes, kunnen ook verwerkings- en communicatiemethodes worden toegevoegd (zie figuur 9).



Figuur 9: Verwerking methodes toevoegen

De meeste methodes spreken voor zich, alleen de e-mail vergt nog wat uitleg. Bij het toevoegen van een mailmethode kunnen in de tekst van de e-mail (de body) links en tekstuele waardes van velden toegevoegd worden (zie figuur 10). Dit kan door de volgende regels toe te voegen:

~text|sVeldNaam~, dit levert in de mail de tekstuele waarde die in het veld 'sVeldNaam' in de database van dit object staat.

~link|bVinkje|text van de link~, dit levert een link waarbij na het klikken op de link het veld 'bVinkje' (boolean) op true wordt gezet.

Figuur 10: E-mail instellen

In het tabje feedback is het ook mogelijk om veldwaardes op te vragen in de tekst. De feedbacktekst wordt getoond in de browser zodra een workflow-gebruiker op een link uit een mailtje klikt.

3.2.3 Nieuwe methode aanmaken

In de praktijk zal hoogstwaarschijnlijk de vraag naar extra methodes groot zijn, om daarmee de bedrijfsprocessen van klanten nog beter te modelleren. Hieronder volgen de stappen die nodig zijn om een nieuwe methode toe te voegen.

Stap 1:

In de database moet een nieuwe tabel toegevoegd worden (zie figuur 11). Hierbij is het nodig om de type op 'p' te zetten. Dit zal een workflow-tabel met prefix 'p' opleveren (pNaamVanTabel), de letter 'p' is gekozen omdat de letter 'w' in het CMS al bezet was door widgets, 'p' is een alternatief, en staat voor "processingmethod". Verder moet het formulierveld 'bevat een workflow-methode' (onderaan 1e tabblad) ingesteld worden op het juiste type methode (zie figuur 9). Tot slot moet er bij het kopje extra een vinkje worden gezet bij 'zichtbaar'. Dit zorgt ervoor dat de nieuwe methode ook wordt opgenomen in het uitklapmenu als er op de + knop is gedrukt in de activiteitmethode-module.

Figuur 11: Tabel voor een nieuw type workflow-methode toevoegen

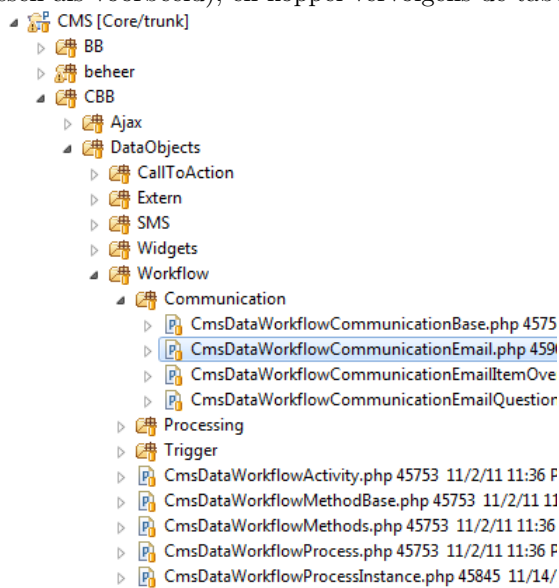
Stap 2:

Nadat er een tabel is toegevoegd kunnen de bijbehorende velden toegevoegd worden die nodig zijn voor de methode (zie figuur 12).

naam	type	leesbare naam	tab
IReceivers	popup met opties	ontvangers van de e-mail	mail
ISender	keuze veld	afzender email-adres	mail
sTitleText	korte tekst	titel van de e-mail	mail
sBodyText	tekst	inhoudelijke tekst van de e-mail	mail
sTitleAfterClick	korte tekst	titel op website na klikken	feedback
sTextAfterClick	HTML	tekst op website na klikken	feedback

Figuur 12: Velden die toegevoegd zijn voor een e-mail methode.

Nu hoeft de methode alleen nog geïmplementeerd te worden in het CMS. Voor alle bestaande methodes is een PHP klasse aangemaakt in het CMS in /CBB/DataObjects/Workflow/... (zie figuur 13). Maak een nieuwe klasse aan als overerving van het bijbehorende type (zie andere klassen als voorbeeld), en koppel vervolgens de tabel in het CMS.



Figuur 13: Map met alle methodes in de code van het CMS

3.2.4 Module instanties processen

Dit is de tweede module in het CMS die direct in verband staat met de workflows. In deze module zijn alle aangemaakte items die een workflow starten terug te zien (zie figuur 14). Het zijn als het ware instanties van het model zoals dat in module 'Workflow processen' is opgesteld.

	tijdstip aangemaakt	proces
[activiteitinstanties]	28 nov 2011 14:56	Project stappenplan EBD
[activiteitinstanties]	28 nov 2011 14:58	Project stappenplan EBD

Figuur 14: Een voorbeeld van procesinstanties na het aanmaken van 2 projecten

Hier kan doorgelinkt worden naar de activiteitinstanties die behoren bij de procesinstantie (zie figuur 15). Er verschijnt een lijst met alle activiteiten als instantie, en de bijbehorende statussen.

procesinstanties (2) - activiteitinstanties van 28 nov 2011 14:56 (33)			
activiteit	status van activiteit	momentopname op	toegewezen gebruiker of rol
Aanmaken van project in de projectenmodule	afgerond/goedgekeurd	28 nov 2011 14:56	David Lansen
Voorlopige begroting en plan uitwerken	verbeteren	28 nov 2011 14:56	David Lansen
Voorlopige begroting en plan intern goedkeuren	afgekeurd/deadline verstreken	28 nov 2011 14:56	David Lansen
Voorlopige begroting en plan opsturen naar fonds(en) voor goedkeuring	openstaand	28 nov 2011 14:56	David Lansen
Voorlopige begroting en plan verbeteren	onzichtbaar	28 nov 2011 14:56	David Lansen
Definitieve begroting en dekkingsplan uitwerken	openstaand	28 nov 2011 14:56	-
Definitieve begroting en dekkingsplan opsturen naar fonds(en) voor goedkeuring	openstaand	28 nov 2011 14:56	-
Goedkeuring door fonds van begroting en dekkingsplan ontvangen	onzichtbaar	28 nov 2011 14:56	-
Afkeuring door fonds van begroting en dekkingsplan ontvangen	onzichtbaar	28 nov 2011 14:56	-
Goedkeuring door fonds van voorlopige begroting en plan ontvangen	onzichtbaar	28 nov 2011 14:56	David Lansen
Afkeuring door fonds van voorlopige begroting en plan ontvangen	onzichtbaar	28 nov 2011 14:56	David Lansen
Contract ondertekenen	openstaand	28 nov 2011 14:56	-
Deadline, geld nog niet ontvangen	onzichtbaar	28 nov 2011 14:56	-
Uitvoering van het project	openstaand	28 nov 2011 14:56	-
Bestedings- verklaring en verslag uitwerken	openstaand	28 nov 2011 14:56	-
Bestedings- verklaring en verslag opsturen naar fonds(en)	openstaand	28 nov 2011 14:56	-
Project afsluiten en archiveren	openstaand	28 nov 2011 14:56	-
-----voor het overzicht-----	onzichtbaar	28 nov 2011 14:56	-
Definitieve begroting en dekkingsplan is opgestuurd	onzichtbaar	28 nov 2011 14:56	-
Voorlopige begroting en plan is opgestuurd naar de fondsen	onzichtbaar	28 nov 2011 14:56	David Lansen
Deadline Contract tekenen	onzichtbaar	28 nov 2011 14:56	-
Definitieve begroting en dekkingsplan intern goedkeuren	openstaand	28 nov 2011 14:56	-

Figuur 15: Een procesinstantie met alle activiteitinstanties in een overzicht.

Op dit moment dient deze module alleen als overzicht voor Content Power, en is deze niet zichtbaar voor de klant. Dit is voornamelijk omdat de klant een eigen interface heeft om inzicht te krijgen in de lopende processen via een webpagina. De module voor instanties in het CMS toont namelijk de processen niet in de juiste volgorde, en verbergt de processen met status 'verborgen' niet. In de toekomst kan Content Power het workflow-framework uitbreiden met een betere weergave van de instanties.

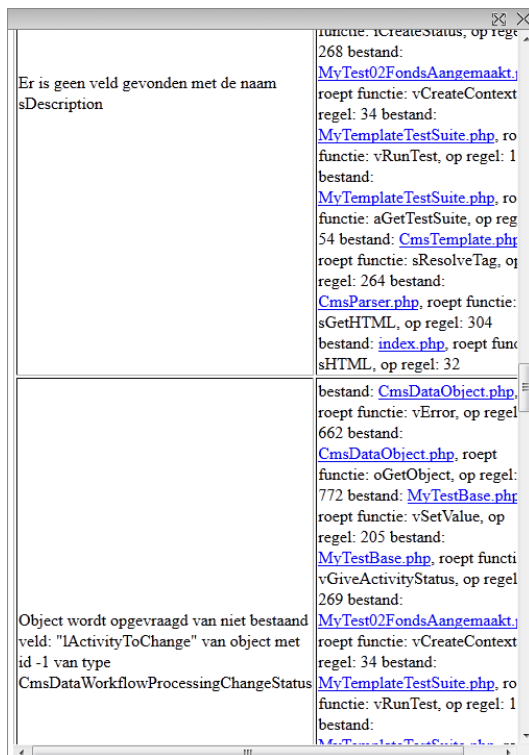
3.2.5 Testsuite

Om het correct functioneren van de code te garanderen is er een testsuite gebouwd. Bij elke nieuwe versie van het CMS zou gecontroleerd moeten worden of alle testen nog slagen. Indien dit niet het geval is kan het CMS van de klanten met een workflow-module niet direct geüpgraded worden, en moet de fout opgelost worden. In figuur 16 is het startscherm van de testsuite te zien.



Figuur 16: Startscherm van de testsuite

Nadat de testsuite uitgevoerd is wordt er aangegeven of de tests geslaagd zijn. Indien er een fout is opgetreden, kan er via de link 'error' een pop-up geopend worden waarin alle fouten en meldingen voor die test te zien zijn (zie figuur 17).



Figuur 17: Pop-up met alle foutmeldingen

4 Werkproces

Deze sectie evalueert de planning per week zoals deze is gespecificeerd in het Plan van Aanpak. Per week is aangegeven wat er in de betreffende week is bereikt.

Week 1

In de eerste week hebben wij kennis gemaakt met Content Power en ons verdiept in de werkwijze. Verder is georiënteerd op het CMS waar de integratie in zal plaatsvinden. Daarnaast hebben wij ons in het probleem verdiept om zo een duidelijk beeld te krijgen van wat er precies ontwikkeld moet worden.

Week 2

Tijdens de tweede week hebben wij ons verdiept in de workflow-theorie. Hiervoor zijn een aantal verschillende papers en boeken geraadpleegd. Een bibliografie is te vinden in appendix B. Met de verkregen inzichten in hoe workflow-processen in elkaar zitten, is daarna begonnen met de voorbereiding voor uitnodigingen voor de interviews. Aan het eind van deze week vierde Content Power het tienjarige bestaan met een symposium. Hier hebben we kennis kunnen maken met verschillende klanten van Content Power, en een aantal klanten uitgenodigd voor een interview.

Week 3

In deze week is het Plan van Aanpak opgesteld en een begin gemaakt met het verzamelen van eisen voor het Requirements and Analysis Document.

Week 4

Op maandagmiddag heeft het eerste interview plaatsgevonden, met Marcel Bon van NEVI. Zie voor meer informatie de sectie interviews in appendix D. Als voorbereiding voor dit interview is een opzet van het huidige proces van NEVI in een schema uitgetekend. Later in deze week is de lijst met eisen definitief bepaald. Deze eisen zijn doorgenomen met de begeleiders vanuit Content Power. Vervolgens is het Requirements and Analysis document verbeterd en afgerond.

Week 5

In deze week is een start gemaakt met het ontwerp. In deze week is eerst het databasemodel gemaakt, en vervolgens een klassendiagram. Op donderdag vond het tweede interview plaats, bij Evenementenbureau Delft. Dit werd tevens de case voor de gebruikerstest. Het interview is terug te lezen in de sectie interviews in appendix D.

Week 6

In deze week is het volledige databasemodel in het CMS geplaatst en is de globale structuur van de modules en het framework geïmplementeerd. Aan het eind van deze week konden we al een workflow maken met een trigger die reageerde op het aanmaken van een item, en vervolgens een e-mail met een klikbare link verstuurd. Dit was een van de eenvoudigste usecases.

Week 7

Op maandag heeft het laatste interview plaatsgevonden bij VOC, zie voor een verslag van dit interview sectie interviews in appendix D. Dit interview gaf nog een aantal nieuwe aandachtspunten. Voor zover dat mogelijk is hebben we hier het ontwerp op aangepast, wat vanwege het sashimi-model toegestaan was. Deze week zijn grote delen van de functionaliteit van het framework geïmplementeerd, en kon de workflow van Evenementenbureau Delft uitgewerkt worden.

Week 8

Op maandag is de code geëxporteerd en doorgestuurd naar de Software Improvement Group. Verder stond deze week in het teken van problemen oplossen en het framework uitbreiden. De workflow van het Evenementenbureau Delft is deze week helemaal afgemaakt, en er is een projectenoverzicht gebouwd. Op donderdag heeft de gebruikerstest plaatsgevonden. Voor een verslag daarvan, zie sectie 5.5.2.

Week 9

In deze week zijn we voornamelijk aan de slag gegaan met het eindverslag. Daarnaast is het systeem verbeterd aan de hand van de feedback tijdens de gebruikerstest.

Week 10

In deze week is het eindverslag afgerond en de gebruikersdocumentatie opgesteld. Deze week worden afspraken gemaakt met de begeleiders over de afronding en presentatie.

5 Fasen van het softwareontwikkeltraject

Deze sectie beschrijft alle fasen die zijn doorlopen tijdens het softwareontwikkeltraject.

5.1 Oriëntatiefase

Tijdens de oriëntatiefase is een analyse gedaan van het volledige softwareontwikkeltraject zoals dat bij Content Power plaatsvindt. In deze fase is verder literatuuronderzoek gedaan naar workflowtheorie. In appendix B wordt hier uitgebreid op ingegaan. Een aantal interessante boeken over dit onderwerp zijn aanwezig in de TU Delft Library, en worden gerefereerd in het oriëntatieverslag (zie appendix B). Met deze kennis, en de begeleiding vanuit Content Power is een goed inzicht verkregen in de werking van workflow-systemen. Deze kennis is in latere fasen toegepast. Dit was allemaal nodig om in de analysefase tot een goed plan te komen voor het ontwerp van de workflow-module. Tijdens de oriëntatiefase hadden we het geluk gehad dat Content Power tien jaar bestond. Om dit te vieren was een symposium georganiseerd, en waren veel grote klanten van het bedrijf aanwezig. Voor ons was dit een mooie gelegenheid om klanten te benaderen met de vraag of wij een interview mochten afnemen. Als afronding van de oriëntatiefase is een Plan van Aanpak opgesteld (zie appendix C). Deze bevat informatie over onze aanpak van het probleem, en een uitgebreide planning als leidraad gedurende het softwareontwikkelproces.

5.2 Analysefase

Tijdens de analysefase zijn de eisen opgesteld waaraan het eindresultaat moet voldoen. Het Requirements and Analysis Document (appendix D) is opgesteld in deze fase. Dit document bevat onder andere de eisen waaraan het te ontwikkelen systeem moet voldoen. Deze eisen zijn opgenomen in een MoSCoW-overzicht. Verder bevat het document een aantal usecases. Tijdens de interviews die plaatsvonden in de analysefase is ons duidelijk geworden welke bedrijfsprocessen en welke problemen er binnen veel bedrijven spelen, en wat de wensen binnen deze bedrijven zijn. Deze wensen zijn voor zover dat mogelijk was verwerkt in de eisen en in het ontwerp. Een van de grote problemen binnen de geïnterviewde bedrijven is de inefficiënte interne / geautomatiseerde communicatie, en het gemis van een overzicht waar iedereen mee bezig is. Tijdens de analysefase is ook een testplan opgesteld. Dit bevat een beschrijving hoe het te bouwen framework getest zal worden. Dit gebeurt door middel van automatische regressietests, en een gebruikerstest. Deze gebruikerstest zal plaatsvinden in de testfase, en een praktijkprobleem van een klant zal worden gemodelleerd in de workflow-module. Na alle interviews is Evenementenbureau Delft (EBD) geselecteerd om deze gebruikerstest uit te voeren. NEVI had geen directe behoefte aan workflowmanagement in het CMS, VOC had een voorstel voor een workflow maar door de grote omvang van deze workflow zou dit niet binnen het tijdsbestek van de projectplanning passen. EBD heeft de wens om het proces van organiseren van projecten te automatiseren. Dit proces is om te zetten naar een workflow die precies past in het plaatje wat wij verwachten binnen 3 weken implementatietijd te kunnen realiseren in de workflow-module. In de analysefase is een usecase opgesteld die het organiseren van projecten bij EBD d.m.v. een workflow beschrijft. In de testfase zal deze usecase in de workflow-module worden ingevoerd, en als gebruikerstest dienen.

5.3 Ontwerpfase

Tijdens de ontwerpfase is een klassendiagram en databasemodel opgesteld. Deze zijn te raadplegen in het Technical Design and Specification Document (appendix E). Om tot de specificaties te komen is de kennis uit voorgaande fasen meegenomen. Ook is georiënteerd op de structuur in bestaande workflow-managementsystemen. In een aantal discussiemomenten met elkaar en met Jan Niemantsverdriet, project manager bij Content Power, zijn wij tot het ontwerp gekomen. Tijdens een van de brainstormsessies ontdekten wij dat onze definitie van een activiteit in een workflowproces niet volledig duidelijk was, en beter gespecificeerd moest worden. Tijdens deze sessies is ook duidelijk geworden dat een activiteit altijd een of meer triggers en communicatiemethodes

moet bevatten, maar nul of meer verwerkingsmethodes hoeft te bevatten. De structuur van de implementatie verandert door dit gegeven niet, maar het speelt wel een rol bij het definiëren van welke stappen uit bedrijfsprocessen tot een enkele workflow-activiteit gerekend kunnen worden. Verder is in de ontwerpfase ook het testplan opgesteld, zie daarvoor appendix F.

De structuur van het ontwerp was voldoende. Er is tijdens de implementatie in het ontwerp weinig veranderd. Alleen de naamgeving van tabellen, klassen en methodes zijn tijdens de implementatie soms veranderd, en de audit-trail is weggelaten. Voor de wijziging in de audittrail, zie sectie 5.4.1.

5.3.1 Functionele eisen

- **Must have:** Alle 'must have's' zijn geïmplementeerd. Alleen de functionaliteit om een gebruiker toe te voegen, wijzigen of verwijderen bleek overbodig. Er is namelijk al een module voor gebruikers aanwezig in het CMS van Content Power. Hierin kunnen we dus al gebruikers en rollen toevoegen die vervolgens in het workflow-model gebruikt kunnen worden.
- **Should have:** Bijna alle 'should have's' zijn geïmplementeerd. Alleen het activiteiten los aan- en uitzetten is niet geïmplementeerd. De reden hiervoor is dat deze functie uiteindelijk niet nodig bleek, en eerder in de categorie won't have hoort.
- **Could have:** De meeste 'could have's' zijn wel geïmplementeerd maar in sommige gevallen op een alternatieve manier. Zo is het bijvoorbeeld wel mogelijk om processen te herstarten of te resetten door een vinkje aan en uit te zetten, maar hiervoor is geen aparte reset knop. Een reset knop is bij de workflow die wij als testcase gebruiken niet nodig en is daarom ook niet geïmplementeerd. Verder is een wachtconditie ook een van de eisen uit de lijst met 'could have's'. Toch blijkt deze overbodig. Het is namelijk mogelijk om een wachtconditie toe te voegen door middel van een deadline-trigger. De activiteit zal dan na het aflopen van de deadline weer worden geactiveerd.
- **Won't have:** Een klein aantal eisen uit de categorie won't have is (soms gedeeltelijk) geïmplementeerd. Maar de openstaande eisen kunnen in toekomstige ontwikkeling door Content Power worden opgepakt.

5.3.2 Niet-functionele eisen

De niet-functionele eisen die in het Requirements and Analysis Document beschreven staan, zijn gehanteerd. Dit is gedaan in het belang van Content Power zodat zij verder kunnen met het ontwikkelen van de workflow-module. Zonder deze eisen had het voor Content Power lastig - zo niet onmogelijk - geworden om de module te gebruiken.

5.4 Implementatiefase

De implementatie bestaat uit twee onafhankelijke gedeeltes. Het databasemodel dient ingevoerd te worden via een bestaande module voor het aanpassen van de databasestructuur in het CMS. Deze structuur wordt in sectie 5.4.1 beschreven. Om vervolgens het databasemodel correct te tonen, en verwerking van workflows mogelijk te maken, is programmacode geschreven. Dit is beschreven in sectie 5.4.2. De combinatie van de twee gedeeltes maakt de huidige functionaliteiten mogelijk. In sectie 5.4.3 bespreken wij de beoordeling van onze code door de Software Improvement Group (SIG). Voor het volledige rapport van SIG verwijzen we naar appendix H.

De implementatiefase verliep voorspoedig. Dankzij de goede voorbereiding in de eerste vijf weken en een goed doordacht ontwerp, kon het framework en de module voor workflow-management vlot geïmplementeerd worden. In de eerste week van de implementatie was het overgrote deel van het framework gebouwd. In de tweede week zijn veel uitbreidingen in de vorm van trigger-, verwerkings- en communicatiemethodes geschreven, en is een testsuite gebouwd voor het uitvoeren van automatische tests. In de derde week is een begin gemaakt met het verwerken van de usecase

van Evenementenbureau Delft (EBD) in de workflow-module. Ook is in deze week een grafische schil gebouwd voor het weergeven van de status van projecten die voorkomen in de usecase van EBD. In de vierde week vond de gebruikerstest plaats. Deze week stond in het teken van verbeteren van de stabiliteit en robuustheid van het workflow-framework. Tijdens de implementatie waren er geen noemenswaardige problemen. Het CMS van Content Power is erg uitgebreid in mogelijkheden, en daarom ook moeilijk om als buitenstaander te doorgronden. Het vertrouwd raken met de methodes die in het CMS worden gebruikt, zoals bijvoorbeeld de objectrelationele mapping van records uit de database, dataobjecten in de PHP-code, content- en viewproviders, het verwerken van query's in het databaseobject, kostte wat tijd. Het zijn stuk voor stuk wel geavanceerde methodes die veelvuldig in onze code worden gebruikt. Om het ontwerp te verwerken in het bestaande systeem, zijn wijzigingen aangebracht in de kern van het CMS. Aangezien dit zeer ingrijpend is, kunnen we toch terugkijken op een succesvolle implementatie dankzij de begeleiding van Jan Niemandsverdriet. Het is gelukt om vrijwel alle functionele eisen te implementeren.

5.4.1 Databasemodel

In de ontwerpfase is het databasemodel uitgedacht. Dit model is opgenomen in het Technical Design and Specification Document (appendix E). Dit model is tijdens de implementatiefase vrijwel gelijk gebleven. Wel is er een wijziging betreffende de tabel `cWfAuditTrail` geweest. Deze tabel bleek niet nodig te zijn in het framework, omdat de benodigde data om een audit-trail te genereren ook bijgehouden wordt in de tabel met activiteitinstanties. Elke instantie heeft een status, die in de audit-trail gebruikt kan worden. Het CMS beschikt over een functie om voor velden de geschiedenis bij te houden. Via deze geschiedenis kan de volledige audit-trail herleid worden. Het CMS is uitgebreid met mogelijkheden om eenvoudig nieuwe trigger-, verwerking- of communicatiemethodes toe te voegen aan het workflow-framework. Deze uitbreidingen kunnen modulair plaatsvinden. Er zijn geen wijzigingen in het bestaande databasemodel vereist. Dit maakt het voor de programmeurs van Content Power aantrekkelijker om in de toekomst nieuwe methodes toe te voegen.

5.4.2 Programmacode

Alle server-side code is geschreven in objectgeoriënteerde PHP (versie 5.3.x). Deze code is geïntegreerd in de bestaande codebase van het CMS, en maakt daarom ook gebruik van alle functionaliteiten uit andere klassen, zoals de database-klassen, database-objecten, errorhandling, datamodellen, instellingen, lay-out, enz. In het oriëntatieverslag (appendix B) is een beschrijving van de werkwijze en codebase opgenomen. Alle client-side code is geschreven in XHTML, CSS, Javascript en het jQuery framework voor Javascript. Deze code wordt door de browsers geïnterpreteerd, en is bedoeld om pagina's van websites visueel mooier te maken. Toch is er gedurende de implementatiefase voornamelijk PHP-code geschreven.

5.4.3 Codebeoordeling door de Software Improvement Group

Onze code scoort bijna 5 sterren op het onderhoudbaarheidsmodel van SIG. In onderstaande paragrafen worden de factoren besproken die ervoor zorgden dat de perfecte score van 5 sterren niet werd gehaald.

- **Unit size:** De test op unit size controleert het percentage code dat bovengemiddeld lang is. Over het algemeen kunnen functies die meer dan 25 regels beslaan opgesplitst worden in functies met minder code. Deze opmerking is ook zeer terecht, en wij zullen deze aanbeveling doorgeven aan Content Power. Vanuit Content Power is aangegeven dat dit stuk code niet gecorrigeerd dient te worden, dit vanwege het feit dat opsplitsing, door de overervende structuur van de bestaande klassen uit het systeem (met honderden overervende methodes) toekomstige uitbreidingen met nieuwe subclasses juist onoverzichtelijker maakt, en het spreekwoord 'door de methodes de klasse niet meer zien'. Deze wijziging zal in de wensenlijst voor een toekomstige refactoring van het CMS opgenomen worden.

- **Unit interfacing:** De test op unit interfacing controleert het percentage code met een bovengemiddeld aantal parameters. Meer parameters duidt vaak op een gebrek aan abstractie, en het kan verwarring geven omdat methodes met veel parameters ook vaak langer worden. Wij begrijpen deze aanbeveling volledig, want er zijn inderdaad veel methodes aanwezig die meerdere parameters accepteren. Zo krijgt bijna elke methode minimaal een array met waarden, en een url-handler mee. Toch is het onhaalbaar en niet rendabel om dit probleem op te lossen in het bestaande systeem. Gedurende de ontwikkeling van het CMS in de afgelopen jaren is het aantal parameters in veel methodes gegroeid door toenemende functionaliteit. Deze evolutie in het systeem heeft tot gevolg dat veel methodes die door ons zijn geïmplementeerd een overerving zijn, of gebruik maken van bestaande methodes. Deze methodes kunnen niet zonder meer worden gecorrigeerd omdat ze al op een grote hoeveelheid plaatsen in de code worden gebruikt. Hierdoor is het doorgeven van de parameters essentieel en onvermijdelijk. PHP 5.3.x ondersteunt bovendien geen overerving van methodes met een verschillend aantal parameters, zoals dat in Java wel mogelijk is. Alleen overerving met parameters die optioneel ingevuld kunnen worden is mogelijk. Voor kleine applicaties programmeert dat erg makkelijk, maar voor grote applicaties levert dat een gebrek aan schaalbaarheid op, doordat er minder abstractie in methodes te bereiken is. Deze wijziging zal in de wensenlijst voor een toekomstige refactoring van het CMS opgenomen worden.
- **Globale variabelen:** Er wordt in de bestaande CMS code veelvuldig gebruik gemaakt van globale variabelen. Deze variabelen horen inderdaad niet thuis in een nette objectgeoriënteerde omgeving. De gebruikte globale variabelen worden slechts gebruikt om paden naar directories te definiëren, en worden niet voor meer zaken gebruikt. Wij raden Content Power dan ook af om meer globale variabelen te gebruiken. Alle klassen van het CMS zijn een overerving van een gemeenschappelijke klasse: `CpErrorObject`. Het is mogelijk om deze klasse uit te breiden met een aantal methodes, zoals `sGetCBBDir()` die de waarde van `global $sCBBDir`; retourneert, en zo ook voor alle andere globale variabelen methodes aan te maken. Op dat moment zijn er geen globals meer nodig in de methodes. Een mooie objectgeoriënteerde oplossing zou zijn om dit in alle bestanden van het CMS aan te passen. Content Power geeft echter toch de voorkeur aan de huidige globals, omdat door de grootte van de codebase van het CMS (140.000 regels code, exclusief alle maatwerkcode van klanten) de ingreep te ingrijpend is. Als wij alleen in onze code deze aanpassing doorvoeren, is de code niet meer consistent met de rest van de codebase. Deze wijziging zal in de wensenlijst voor een toekomstige refactoring van het CMS opgenomen worden.

De hoge score van bijna 5 sterren is erg mooi. Wel zouden wij SIG willen verzoeken om naast de aanbevelingen voor verbeteringen, ook de factoren die voor de hoge score hebben gezorgd te specificeren. Welke factoren maken de code zo "bovengemiddeld onderhoudbaar", zijn er factoren waarvan wij ons niet bewust zijn?

5.5 Testfase

De testfase liep gelijk aan de implementatiefase. Dat betekent dat code die werd geschreven, ook direct getest werd. Tests zijn uitgevoerd zoals beschreven in het testplan (zie appendix F). Tijdens de testfase heeft een gebruikerstest van de workflow-module plaatsgevonden bij Evenementenbureau Delft. De resultaten hiervan zijn te lezen in sectie 5.5.2. Er zijn tijdens deze gebruikerstest een aantal punten naar voren gekomen voor verbeteringen in het workflow-model en in het projectenoverzicht. De workflow-module is standaard niet zichtbaar voor de eindgebruikers. Het beheer van workflow-processen zal in de beginfase door Content Power plaatsvinden. Gebruikers merken alleen de aanwezigheid van de module door de mailtjes die worden verstuurd. Door deadlines en vinkjes te zetten is beïnvloeding van individuele proces-instanties mogelijk, maar het meeste werk gebeurt op de achtergrond. Door onze eigen handmatige tests, bijvoorbeeld bij het aanmaken van een workflow-proces in de workflow-module, kwamen diverse punten voor verbetering boven drijven. Tijdens de implementatiefase is een testsuite gebouwd waarin een aantal regressietests

zijn opgenomen. De resultaten van deze testsuite zijn te lezen in sectie 5.5.1. Hiermee kunnen toekomstige fouten automatisch worden gedetecteerd in stukken code die op dit moment correct werken. Om de kwaliteit van de code te kunnen blijven waarborgen, dient de testsuite bij elke aanpassing in het systeem (eventueel uitgebreid en) succesvol uitgevoerd te worden.

De resultaten van de uitgevoerde tests worden in de volgende secties beschreven.

5.5.1 Resultaten van geautomatiseerde tests

Voor het workflow-framework zijn regressietests geschreven. Deze regressietests hebben veel overeenkomsten met unit-tests, met het verschil dat niet alle losse methodes afzonderlijk worden getest, maar een volledige verzameling van stappen getest wordt. Deze regressietests kunnen aan een testsuite worden toegevoegd, om automatische tests mogelijk te maken. Het CMS beschikte nog niet over een testsuite waarmee het mogelijk is om in 1 keer verschillende tests achtereenvolgens uit te voeren. Alle (doelbewust gemaakte) fouten die tijdens deze tests worden gegenereerd worden afgevangen en zijn na het draaien van de testsuite, per test terug te zien. Hiermee is eenvoudig na te gaan of het workflow-framework nog correct werkt. Tijdens de implementatiefase is de testsuite gemaakt. Voornamelijk door de korte duur van het project heeft de testsuite nog weinig profijt opgeleverd. Dit werd deels veroorzaakt door bugs in de testsuite, en wijzigingen in het databasemodel, waardoor de testsuite weer aangepast moest worden. Een gevonden bug is wel toe te schrijven aan de testsuite. Wellicht dat de testsuite in de toekomst nog veel voordelen zal bieden. Onder andere om het workflow-framework te testen zodra deze ook op andere servers in gebruik genomen zal worden. Ook vinden zo nu en dan updates van PHP plaats op de servers. Het is nuttig om dan de testsuite uit te voeren.

5.5.2 Resultaten van gebruikerstest

Op 17 november 2011 heeft er een gebruikerstest plaatsgevonden op het kantoor van Evenementenbureau Delft (EBD). Deze gebruikerstest is gebaseerd op de workflow van projecten organiseren bij EBD. Meer informatie daarover is te vinden in appendix F. Susanne Dix en David Lansen hebben kennisgemaakt met de workflow-module, en de mogelijkheden uitgeprobeerd. Voorafgaand aan de gebruikerstest is een handleiding overhandigd met uitleg en stappenplannen hoe met de module gewerkt kon worden. Deze gebruikerstest is opgenomen in appendix G. Tijdens deze gebruikerstest zijn een aantal punten genoemd die voor verbetering vatbaar zijn. Verder zijn een hoop complimenten gegeven over het systeem, en over ons inzicht in de processen die bij EBD een rol spelen. Onderstaande secties beschrijven de resultaten van de gebruikerstest.

5.5.3 Verbeteringen in het workflow-proces

- Elke keer als er een document verstuurd wordt naar externen (zoals fondsen) moet het eerst intern door de leidinggevende gecontroleerd en goedgekeurd worden. Dat gebeurt nu bij sommige documenten, maar niet bij alle documenten.
- Het contract opsturen en tekenen kan in een enkele stap gebeuren, omdat er gewoon iemand naar het kantoor van een fonds gaat met het contract, en er dan gelijk getekend wordt.
- Moment van contract tekenen moet naar voren verplaatst worden, het tekenen komt namelijk voordat de definitieve begroting wordt opgesteld.
- Als een mail wordt gestuurd met de mededeling dat een document is afgekeurd, is een link om het document 'verbeterd' te markeren duidelijker dan de huidige mogelijkheid om nog een extra keer op 'ja' te drukken.
- Geld wordt meestal in delen ontvangen, meestal een gedeelte voor de activiteit, en de rest na de activiteit. Dit moet ook in het workflow-proces bijgehouden kunnen worden.

- Vaak worden er veel fondsen tegelijk aangeschreven om bij te dragen aan een project, en zullen er uiteindelijk bijv. maar twee toezeggen. De andere fondsen keuren het project af, en doen niet meer mee. Het workflow-proces moet hier mee kunnen omgaan.

5.5.4 Interface

- Het overzicht van projecten kan nog beter, nu worden telkens alle activiteiten opgesomd bij elk project. Het is mogelijk om een aantal projecten direct onder elkaar te plaatsen zonder telkens de activiteiten te vermelden.
- Het overzicht van projecten past niet in het scherm. Naar beneden en boven scrollen is prima, maar naar links en rechts scrollen is wel vervelend. Het overzicht van projecten moet daarom in de breedte van het scherm passen.

5.5.5 Quotes

- "Fijn dat het allemaal via de mail kan, het hele proces kan namelijk worden doorlopen door op de linkjes in de verschillende e-mails te klikken."
- "Het is leuker geworden dan verwacht doordat het zo grafisch is."
- "Ons probleem is hiermee opgelost, hier kan je een groot publiek mee bedienen die met hetzelfde probleem zitten."
- "Het heeft weinig geld gekost, maar hiermee kan je veel problemen verhelpen."
- "Veel bedrijven hebben een slecht overzicht over de verschillende bedrijfsprocessen. Met deze workflow-module kunnen onze bedrijfsprocessen inzichtelijk en overzichtelijk worden gemaakt. Groot pluspunt is dat in een overzicht te zien is hoe ver taken zijn gevorderd."

5.6 Afrondingsfase

Tijdens de afrondingsfase is feedback die tijdens de gebruikerstest door Evenementenbureau Delft werd gegeven verwerkt in het workflow-model. Ook zijn in de afrondende fase alle deliverables gecontroleerd, verbeterd en afgerond. Ook is er gebruikersdocumentatie geschreven voor Content Power. Deze beschrijft het gebruiken, aanpassen en uitbreiden van de module. Aan de hand van deze documentatie is het voor Content Power mogelijk om zelf workflow-modellen in de module te plaatsen. Hiermee wordt het mogelijk om de workflow-functionaliteit aan te bieden aan klanten. Bovenstaande deliverables, en de presentatie op 14 december 2011 vormen de officiële afronding van het bachelorproject. Content Power zal daarna nog intern tests uitvoeren met het workflow-framework. Nadat het framework stabiel verklaard is, zal het worden opgenomen in een nieuwe CMS-versie. Bij alle klanten met een CMS zal na verloop van tijd door (handmatige of periodieke) update van het CMS het workflow-framework beschikbaar worden. Dit valt buiten de scope van het bachelorproject.

6 Ervaringen en aanbevelingen

Wij zijn zelf zeer tevreden over wat we in minder dan tien weken tijd gepresteerd hebben. Er was ruim voldoende tijd om alle must-, should-, en zelfs could haves te implementeren. Daarnaast hebben we ook een klant blij kunnen maken door het uitwerken van een op maat gemaakte workflow in onze workflow-module. We hebben ook veel nieuwe ervaringen opgedaan. Zo hebben wij de sfeer kunnen proeven van hoe het er in een klein bedrijf aan toe gaat. Het was ook zeer nuttig om een keer het hele softwareontwikkeltraject van begin tot einde te doorlopen. Dit heeft ons nieuwe inzichten opgeleverd in onder andere de kwaliteit en onderhoudbaarheid van een product/code. De workflow-module is qua functionaliteit nu nog vrij beperkt, maar kan door Content Power eenvoudig via interfaces uitgebreid worden met extra functionaliteiten. Veel nieuwe functionaliteiten hoeven pas gebouwd te worden op het moment dat er een klant is waarvoor het interessant is om te gebruiken. Ondanks dat er een stabiel werkend resultaat is opgeleverd, zijn er nog wel een aantal verbeterpunten op te noemen waarmee het systeem nog functioneler gemaakt kan worden. Dit zijn bijvoorbeeld extra trigger-, verwerkings- en communicatiemethodes die voor onze testcase van Evenementenbureau Delft nog niet essentieel waren. Een voorbeeld hiervan is bijvoorbeeld een communicatiemethode die SMS'jes stuurt. Of om deadlines van de projecten te kopiëren naar de digitale agenda van de klant via iCal/vCal. Verder zou de interface van het uitwerken van een workflow-proces in de module overzichtelijker gemaakt kunnen worden door bij de activiteiten ook direct op hetzelfde scherm de onderliggende trigger-, verwerkings- en communicatiemethodes te tonen. Zo is van elke stap direct in te zien wat deze exact doet. Dat maakt het makkelijker om een workflow aan te passen naar de wensen van een klant. Wij raden Content Power aan om de ontwikkeling van het framework door te zetten, zodat de workflows steeds functioneler worden.

7 Conclusie

De gefaseerde aanpak van het project zoals voorgesteld in het sashimi-model is succesvol te noemen. Het heeft geholpen om een goed werkend product af te leveren. Door de overlappende fases was het mogelijk om voortgang te maken met het project zonder in een bepaalde fase te blijven hangen. Zo kon het laatste interview in week 7 plaatsvinden terwijl de analysefase al lang verstreken was. Verder konden kleine verbeteringen uit voorgaande fases worden verwerkt terwijl het project zich al in de volgende fase bevond. Een voorbeeld hiervan is dat er bijvoorbeeld tijdens de eerste week van de implementatie toch kleine denkfouten in het databasemodel voorkwamen, en het ontwerp daarop nog kon worden aangepast met terugwerkende kracht. Gezien de vele documenten en de grote hoeveelheid tijd die gemoeid was met het opstellen hiervan, was dit project niet efficiënt te noemen, en in de praktijk zelfs onrendabel. Dit 124 pagina's tellende verslag zou pas echt tot zijn recht komen als een groter team over een veel langere periode aan de slag gaat met de implementatie. Desalnietemin is dit project erg leerzaam geweest.

APPENDICES

A Opdrachtomschrijving

Opdrachtschrijving - Bachelorproject IN3405
Workflow-management in een Content Managementsysteem

Rob Ruigrok
R.J.Ruigrok@student.tudelft.nl
1371746

Jan Pieter Mars
J.P.Mars@student.tudelft.nl
1381075

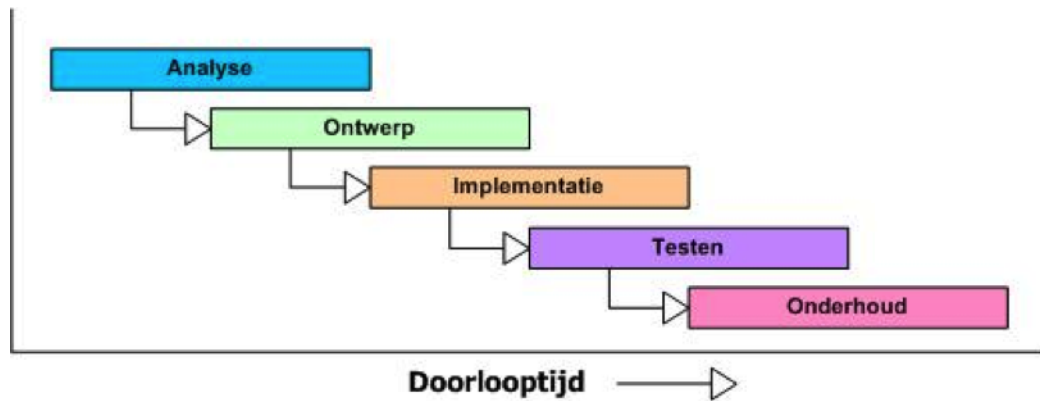
7 december 2011



ContentPower
informatiebeheersystemen

Bachelorproject Content Power Workflow Framework

Bij de uitvoering van dit project wordt gebruik gemaakt van het sashimi-model. Dit is een doorontwikkeling van het watervalmodel.



Figuur 1: Sashimi-model (bron: http://nl.wikipedia.org/wiki/Watervalmethode#Het_.22sashimi.22-model)

De deliverables die opgeleverd worden:

- Projectplan met planning
- Basisontwerp met usecases
- Framework met uitbreidingen (programmeerwerk)
- Testplan en resultaten
- Documentatie voor Content Power
- Documentatie voor de gebruiker

1 Definitiestudie/analyse

Als start van het project moeten de wensen van de (potentiële) klanten van Content Power in kaart worden gebracht. Dit door middel van interviews en/of enquêtes. De volgende vragen dienen in kaart te worden gebracht:

1. Welke administratieve processen vinden er op dit moment bij de bedrijven plaats
2. Hoe zien deze processen er uit:
3. (a) Waardoor wordt het proces gestart
(b) In welke stappen verloopt het proces
(c) Wie zijn er betrokken bij het proces
(d) Wat is het einddoel van het proces
4. Welke communicatiemiddelen worden er gebruikt
5. Wordt het als een voordeel gezien als een systeem deze processen kan begeleiden

2 Opzetten van een framework

Binnen het bestaande framework van Content Power moet een framework en datamodel worden gemaakt waarin deze processen kunnen worden gemodelleerd. Er moet tijdens het bouwen van het framework vooral worden gericht op de eenvoudigere processen die dichtbij het beheren van een website of webwinkel liggen. Binnen het CMS van Content Power moet een module komen waarin de processen kunnen worden samengesteld. De module zal voorlopig alleen door Content Power zelf worden gebruikt. Het samenstellen van processen moet zonder fouten kunnen, maar mag enige technische kennis vereisen.

3 Uitbreiden van het framework

3.1 Triggers

Uit het de inventarisatie worden de belangrijkste triggers (bestelling komt binnen, mail over bepaald onderwerp ontvangen, etc.) bepaald. De triggers worden toegevoegd aan het framework en beschikbaar gemaakt in de module. De triggers zullen worden gebruikt om de processen te starten of weer te hervatten.

3.2 Verwerkingsmethodes

Sommige delen van de administratieve processen kunnen worden geautomatiseerd. Voor de eenvoudige handelingen (verwijderen van items, wijzigen van een waarde, autoreply mail, etc.) moeten elementen in het framework en de module komen.

3.3 Communicatiemiddelen

Uit de inventarisatie worden de belangrijkste communicatiemiddelen (mail, sms, social media, etc.) bepaald. Het framework en de module worden ook voorzien van deze communicatiemiddelen. Deze middelen worden gebruikt om, waar nodig, een verantwoordelijke weer bij het proces te betrekken.

3.4 Doelstelling

Wij verwachten dat de eerste twee punten binnen de gestelde tijd gerealiseerd kunnen worden. Als deze ver genoeg gevorderd zijn, wordt in samenspraak bepaald welke en hoeveel van de triggers, verwerkingsmethodes en communicatiemiddelen nog worden toegevoegd binnen dit project. Het is de bedoeling dat een aantal usecases ook daadwerkelijk gerealiseerd zijn en kunnen werken voor een klant.

3.5 Technische eisen

Het framework en de module moet geïntegreerd worden in het huidige CMS van Content Power. Dit betekent dat er gewerkt kan worden met de volgende talen/frameworks:

- PHP (objectgeoriënteerd) en het framework van Content Power
- Javascript en jQuery
- HTML en CSS

Verder moet de code geschreven worden volgens de richtlijnen van Content Power. Zowel in stijl als documentatie, zodat de code na het project door Content Power overgenomen kan worden.

B Oriëntatieverslag

Oriëntatieverslag Bachelorproject IN3405
Workflow-management in een Content Managementsysteem

Rob Ruigrok
R.J.Ruigrok@student.tudelft.nl
1371746

Jan Pieter Mars
J.P.Mars@student.tudelft.nl
1381075

7 december 2011



ContentPower
informatiebeheersystemen

Voorwoord

Dit verslag beschrijft de oriëntatiefase, de eerste fase van het bachelorproject. In deze fase doen wij onderzoek naar het bedrijf Content Power. Wij onderzoeken welke diensten zij leveren, en welke technieken en methodes Content Power gebruikt bij de ontwikkeling van software. Onze opdracht houdt in om willekeurige bedrijfsprocessen van de klanten van Content Power eenvoudig in een Content Management Systeem (CMS) te beheren. Wij onderzoeken eerst hoe het CMS van Content Power in grote lijnen werkt. Deze kennis is van belang voor ons om een idee te krijgen hoe de uitbreidingen voor het systeem gemodelleerd en ontworpen kunnen worden. Ook bevat dit verslag een uitgebreide analyse over bedrijfsprocessen en workflow-processen. Deze kennis is onmisbaar voor het vervolg van het project, omdat basisconcepten en ideeën met betrekking tot workflow al eerder zijn onderzocht. Bovendien kan op basis van deze kennis een afbakening worden gemaakt voor de planning die in het Plan van Aanpak (volgende fase) zal worden opgesteld.

Samenvatting

Content Power is een bedrijf dat is gespecialiseerd in het ontwikkelen van websites. De basis voor elke website is het CMS. Het CMS vormt een framework om de wensen van de klant te implementeren, en is na jaren van ontwikkeling zeer uitgebreid. Veel klanten van Content Power gebruiken de website als onderdeel van het bedrijfsproces. Voorraadbeheer, webwinkels, online inschrijvingen, ledenadministratie, nieuwsbrieven, intra- en extranetten zijn al regelmatig geïmplementeerd. Content Power vermoedt dat processen die momenteel al geautomatiseerd zijn bij klanten, nog beter geautomatiseerd kunnen worden door het CMS een actieve rol te laten innemen. Want tegenwoordig is de klant nog steeds de initiatiefnemer om in te loggen in het CMS om bedrijfsprocessen te controleren (bijv. is er een nieuw lid? Is er een product verkocht? Etc.). Content Power wil graag deze rol omdraaien: het CMS moet zelf kunnen reageren op acties, en nieuwe acties daarop uitvoeren. Met als doel om bedrijfsprocessen van klanten nog meer te automatiseren, en taken van het personeel uit handen te nemen. In de praktijk bestaat hiervoor al een mooie naam: workflow-processen. Een workflow is een onderdeel van het totale bedrijfsproces. Workflows faciliteren automatisering van een gedeelte of geheel van bedrijfsprocessen door middel van een computer. Het doel van de opdracht is om in het CMS van Content Power een framework te bouwen die workflows ondersteunt.

Inhoudsopgave

1	Content Power	36
2	Huidige werkwijze	36
2.1	Servers	36
2.2	Talen, scripts, en frameworks	36
2.3	Ontwikkelomgeving	36
2.4	Subversion	37
2.4.1	Versiebeheer	37
2.4.2	OTAP	37
3	CMS	38
3.1	Kenmerken ontwikkeling CMS	38
3.2	Code-structuur	39
3.3	Database-structuur	40
3.3.1	Vanuit MySQL perspectief	40
3.3.2	Vanuit CMS perspectief	41
3.3.3	Objectrelationele mapping	41
4	Workflow	42
4.1	Specificatie van resources, rollen en kwalificaties	42
4.2	Specificatie van data	42
4.3	Beschrijving van workflow-condities	42
4.4	Vaststellen van uitvoerders, rechten	42
4.5	Workflow-management	43
4.6	Geschiktheid van bedrijfsprocessen voor workflow-management	43
4.7	Workflow-managementsysteem	44
4.7.1	Mogelijkheden van workflow-managementsysteem	44
4.7.2	Beperkingen van een workflow-managementsysteem	45
4.7.3	Voorbeeldtoepassingen voor workflow	45
4.7.4	Referentiemodel	45
4.7.5	Workflow applicatie	46
4.7.6	Softwarepakketten voor workflow	46
4.7.7	Workflow in een Content Managementsysteem	46
	Referenties	47

1 Content Power

Content Power is opgericht in 2001. Op dit moment werkt Content Power met zes personen aan het realiseren van websites voor klanten. De geleverde websites maken gebruik van het door Content Power zelf ontwikkelde Content Managementsysteem (CMS). Dit CMS is de kern van alle geleverde websites. [1]

2 Huidige werkwijze

Deze sectie beschrijft de huidige werkwijze en werking van processen bij Content Power. Deze analyse is gemaakt, omdat het framework en de module geïntegreerd dienen te worden in het huidige CMS van Content Power, en wij met deze talen en frameworks vertrouwd moeten worden.

2.1 Servers

Het bedrijf heeft een aantal servers, die voor verschillende doeleinden worden ingezet. Zo is er een databaseserver, mailservers, ontwikkelservers, en zijn er diverse productieservers waarop websites zijn geplaatst. De servers draaien op het CentOS 5 besturingssysteem (Linux). Op de servers is de webserver Apache (v2) als service geïnstalleerd voor het verwerken van de website-aanroepen. Verder is er in Apache ondersteuning voor de scripttaal PHP (v5.3.8) en database MySQL (v5) door middel van plug-ins.

2.2 Talen, scripts, en frameworks

- Objectgeoriënteerde PHP. Het bestaande CMS van Content Power is een framework dat geschreven is in de scripttaal PHP. PHP is bedoeld om op webservers dynamische webpagina's te creëren. Wanneer een browser een PHP-document oproept, wordt op de server eerst de in het document opgenomen PHP-code uitgevoerd. Dit gebeurt door de PHP-parser (de PHP-engine). Het resultaat (meestal HTML) wordt door de webserver naar de browser verstuurd. PHP kan echter ook andere documenttypen versturen, zoals allerlei soorten bestanden. [2].
- Javascript en jQuery. JavaScript is een client-side scripttaal, en wordt door browsers geïnterpreteerd. Het kan worden gebruikt om webpagina's interactief te maken. [3]. JavaScript biedt prototype-gebaseerde overerving (dus niet klasse-gebaseerde overerving zoals in Java). jQuery is een vrij framework, en is gebouwd rondom Javascript. Het biedt de mogelijkheid om met een veel eenvoudigere syntaxis dan JavaScript zelf heeft, complexe Javascript-acties uit te voeren op websites. [4].
- HTML is een opmaaktaal voor de specificatie van informatie op het World Wide Web. De belangrijkste eigenschap van HTML is de koppelingen naar andere pagina's waardoor een web van gelinkte pagina's ontstaat. Daarnaast is HTML een opmaaktaal zoals vele andere, met notaties voor het aangeven van nadruk in tekst, van kopjes, van indeling in paragrafen, van tabellen, en van afbeeldingen en multimedia. HTML bestaat uit platte tekst waarin met een tag is aangegeven hoe de tekst moet worden geïnterpreteerd, bijvoorbeeld als lijst of als opschrift. [5]
- CSS staat voor Cascading Style Sheets, oftewel stijlbladen. Het geeft websites de mogelijkheid om de vormgeving van webpagina's los te koppelen van hun feitelijke inhoud, en deze vormgeving centraal vast te leggen in een bestand. Met CSS kan de vormgeving van elk HTML element in een webpagina worden bepaald. [6]

2.3 Ontwikkelomgeving

Voor het implementeren van de processen wordt PDT-Eclipse gebruikt. Dit is een release van de populaire Eclipse IDE, maar met plug-ins voor PHP- en webontwikkeling ingebakken. Bovendien

heeft deze Eclipse versie de mogelijkheid om de syntaxis van PHP en HTML te valideren. Verder is de Subclipse plug-in geïnstalleerd in Eclipse, om gebruik te kunnen maken van Subversion repositories. In principe maakt het niet uit welke applicatie wordt gebruikt voor de ontwikkeling, want zelfs de meest eenvoudige teksteditor voldoet al. Maar vanwege gebruiksgemak geniet Eclipse de voorkeur.

2.4 Subversion

Alle code wordt via een Subversion client gecommited, en verstuurd naar de SVN repository op een van de servers van Content Power. Op deze server wordt na elke commit automatisch een post-commit script gestart. Dit script controleert wat er gewijzigd is tijdens de laatste commit. Vervolgens maakt het script achtereenvolgens SSH-connecties met alle andere servers, en worden op elke server de gewijzigde bestanden bijgewerkt vanaf de Subversion-server. Zo beschikken alle servers altijd over de laatste versies van de bestanden.

2.4.1 Versiebeheer

Voor elke website en elke CMS-versie worden nauwkeurig de versies bijgehouden. Elke versie heeft in Subversion een eigen branch, en elk domein van een website kan aan een eigen branch van de website worden gekoppeld door middel van een symbolische link. Elke website-branch is zelf ook weer gekoppeld aan een eigen CMS-branch. Verder houden databases ook een eigen CMS-versie bij. Soms komt het voor dat er wijzigingen in de structuur van de database moeten plaatsvinden. Omdat de code van het CMS op dat moment mogelijk niet meer backwards compatible is met de database, wordt van deze wijzigingen eenmalig een zogenaamde recording gemaakt. Deze recording is een XML-bestand met daarin de wijzigingen, en kan worden toegevoegd in een automatisch upgrade-systeem voor databases. Dit upgrade-systeem wordt gebruikt om andere databases bij te werken naar nieuwe CMS-versies, zodat na verloop van tijd de wijziging in elk CMS is doorgevoerd.

2.4.2 OTAP

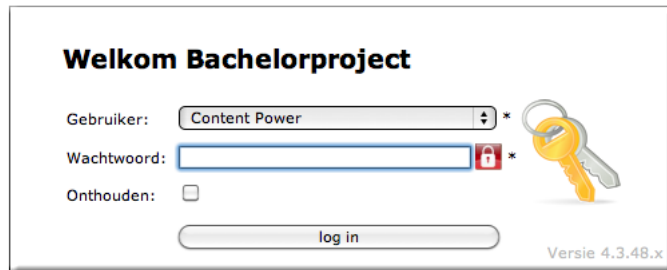
Om een betere scheiding te maken tussen de verschillende ontwikkel- en productiewebsites, wordt gebruik gemaakt van de OTAP-structuur. Dit staat voor Ontwikkeling, Test, Acceptatie, Productie. Verschillende versies (branches) van websites en het CMS kunnen worden gekoppeld aan een van deze omgevingen. Elke omgeving uit OTAP beschikt ook over een eigen database.

- Ontwikkeling: dit domein is altijd gekoppeld aan de trunk van de website. Dit domein wordt gebruikt door de programmeurs tijdens de ontwikkeling van functionaliteit.
- Test: zodra er een stabiele versie in de ontwikkelomgeving klaarstaat, wordt een branch gemaakt voor het testen door een projectleider bij Content Power.
- Acceptatie: als het testen binnen het bedrijf is geslaagd, dan wordt de branch doorgezet naar de acceptatie. De klant kan via deze omgeving de website uitproberen en testen.
- Productie: als de klant tevreden is, dan wordt de branch naar productie doorgezet. Op dat moment is de website wereldwijd voor iedereen beschikbaar.

Het voordeel van OTAP is dat alle vier verschillende omgevingen onafhankelijk en afgezonderd van elkaar gebruikt kunnen worden. Ontwikkeling van nieuwe functionaliteit is daardoor mogelijk zonder de werking van de productiewebsite te onderbreken. Zodra een functie voldoende getest en stabiel is, kan een nieuwe versie van een site worden uitgerold naar de andere omgevingen van OTAP.

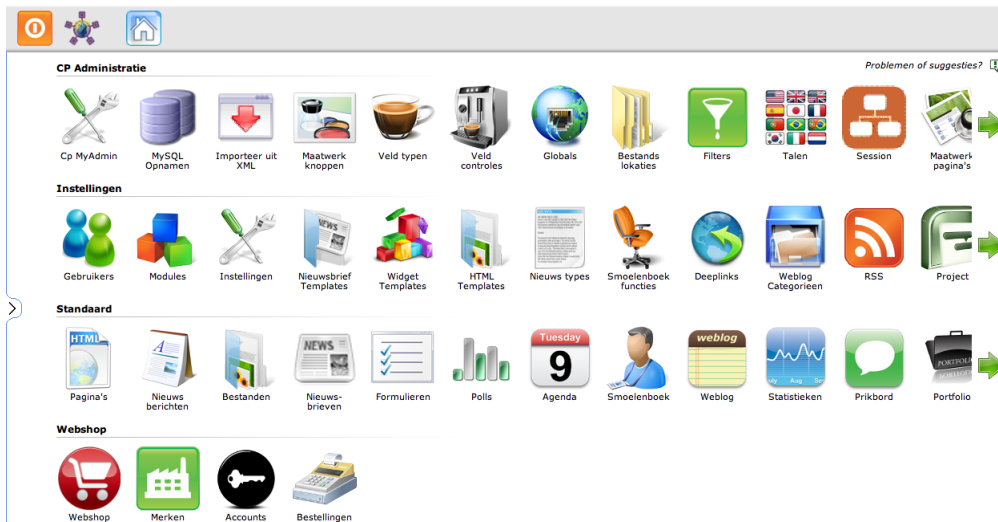
3 CMS

Het CMS is het beheersysteem dat klanten gebruiken om alle informatie die op de website staat bij te houden. In dit systeem worden alleen modules getoond die voor een ingelogde klant van



Figuur 1: Het inlogscherm van het CMS.

toepassing zijn, zoals bijvoorbeeld het bewerken van pagina's, het plaatsen van producten in de webwinkel, statistieken en dergelijke. Wanneer Content Power inlogt, krijgt zij alle modules te zien. Ongeveer de helft van de modules is alleen voor Content Power bedoeld, om de website op maat in te stellen voor de klant. Het CMS werkt op basis van AJAX (Asynchronous Javascript And XML). Acties die worden uitgevoerd in het CMS worden asynchroon verstuurd naar de server, zodat de pagina niet herlaadt.



Figuur 2: Het hoofdscherm van het CMS na het inloggen van de Content Power gebruiker.

3.1 Kenmerken ontwikkeling CMS

Bij de ontwikkeling van het CMS zijn drie kenmerken belangrijk [1] :

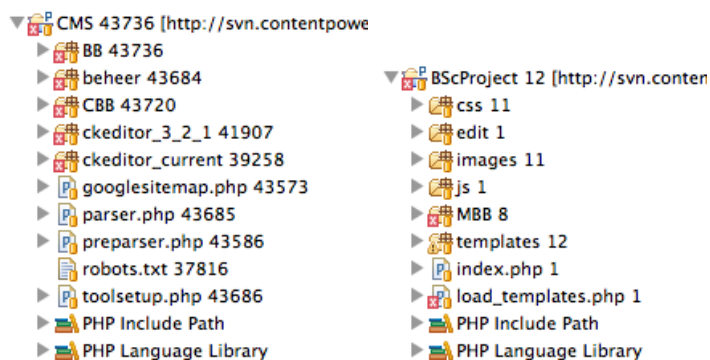
- **Onderhoudbaarheid:** het CMS moet exact aansluiten bij de wensen van de klant. Nieuwe ontwikkelingen en aangepaste functionaliteit mogen nooit een belemmering vormen. Daarom is het CMS modulair en objectgeoriënteerd opgebouwd. Bovendien kan via maatwerk functionaliteit uit het CMS worden overerft zodat het aan de eisen van de klant voldoet.
- **Gebruiksvriendelijkheid:** het CMS moet gebruiksvriendelijk zijn; de klant moet een website logisch en intuïtief kunnen beheren. Daarom krijgt een klant alleen de modules en func-

functionaliteit te zien die hij echt nodig heeft. Ongebruikte functionaliteit wordt automatisch verborgen gehouden.

- Up-to-date: ook na oplevering aan de klant wordt het CMS nog periodiek bijgewerkt. Dit gebeurt via een geautomatiseerd systeem waarbij versies worden bijgehouden. Nieuwere versies leveren meer functionaliteit, beveiliging en snelheid.

3.2 Code-structuur

Websites en CMS worden objectgeoriënteerd opgebouwd in de scripttaal PHP.



Figuur 3: De bestandsstructuur van het CMS met een gekoppelde test-website BScProject in Eclipse.

Er wordt gebruik gemaakt van drie belangrijke packages:

- Buildingblocks (BB). Deze package bevat klassen van veelgebruikte functionaliteiten. De buildingblocks kunnen los van het CMS gebruikt worden op websites. Door een instantie van de klasse aan te maken, een aantal methodes in te stellen, en vervolgens via een methode de HTML op te vragen, wordt hiervan gebruik gemaakt. Deze package bevat bijvoorbeeld klassen voor het versturen van mailtjes, opbouwen van formulieren op websites, maken van menu's, maken van stappenwizards, popups, et cetera.
- CMS Buildingblocks (CBB). Deze package bevat alle klassen die nodig zijn voor een correcte werking van het CMS, zoals bijvoorbeeld het opvragen van informatie uit de database, weergave voor modules in het CMS. Verder bevat CBB de klassen die de volledige CMS functionaliteit mogelijk maken. De CBB klassen maken soms gebruik van de algemene klassen uit BB.
- Maatwerk Buildingblocks (MBB). Deze package is voor elke website verschillend opgebouwd. Aan deze package kunnen klassen worden toegevoegd die de werking van klassen in BB en CBB automatisch overerven. Op deze manier is het mogelijk om elke website volledig naar de wensen van de klant in te richten, door methodes uit de objecten van BB en CBB te overerven.

Wanneer een aanroep voor een pagina wordt gedaan vanuit een browser, worden eerst de HTML-template, CSS- en Javascript-bestanden ingelezen door PHP. Vervolgens wordt de template opgevuld met blokken HTML-code die wordt geleverd door de diverse buildingblocks. Het resultaat is een volledig dynamisch opgebouwde pagina die wordt teruggestuurd naar de browser.

3.3 Database-structuur

3.3.1 Vanuit MySQL perspectief

Op database-niveau zijn alle tabellen uit de database van een website geordend met een prefix letter voor de tabelnaam.

- c-tabellen: dit zijn tabellen die in elk CMS (dus altijd) bestaan, en altijd beschikbaar zijn in CBB. De afbeeldingen hieronder tonen een voorbeeld.

cAccount						0	MyISAM	utf8_general_ci	1.0 KiB	-
cAdress						0	MyISAM	utf8_general_ci	1.0 KiB	-
cAgenda						0	MyISAM	utf8_general_ci	1.0 KiB	-

- w-tabellen: deze tabellen hebben dezelfde eigenschappen als c-tabellen, maar worden op websites specifiek gebruikt als widgets. Een voorbeeld hiervan is wTwitterProfile.

wTwitterProfile						0	MyISAM	utf8_general_ci	2.0 KiB	-
wWeblogArchive						0	MyISAM	utf8_general_ci	1.0 KiB	-
wWeblogCalendar						0	MyISAM	utf8_general_ci	1.0 KiB	-
wWeblogCategories						0	MyISAM	utf8_general_ci	1.0 KiB	-

- m-tabellen: deze tabellen zijn website-specifiek aangemaakt als maatwerktabel. Deze tabellen mogen dus alleen gebruikt worden in MBB.
- s-tabellen: dit zijn systeemtabellen voor het CMS, en bevatten metadata die voor een correcte werking van het CMS zorgen. Tot de metadata behoort onder andere de structuur van alle c, m en w tabellen, de bijbehorende velden, en de items die in deze tabellen aanwezig zijn. Via deze metadata tabellen is het mogelijk om rechten op items in te stellen, en diverse instellingen te doen aan de CMS structuur.

sMetaField						656	MyISAM	utf8_general_ci	81.0 KiB	-
sMetaItem						1,004	MyISAM	utf8_general_ci	165.5 KiB	-
sMetaItemOrigineel						552	MyISAM	utf8_general_ci	30.1 KiB	-
sMetaTable						131	MyISAM	utf8_general_ci	23.8 KiB	-
sRightsItem						93	MyISAM	utf8_general_ci	15.4 KiB	-

- h-tabellen: bij tabellen kan worden opgegeven of er geschiedenis van de bewerkingen moet worden bijgehouden. Als dat het geval is, dan wordt dat bijgehouden in tabellen met een prefix h.

3.3.2 Vanuit CMS perspectief

Wijzigingen in de structuur van de database worden nooit rechtstreeks in MySQL uitgevoerd, maar altijd via de module CpMyAdmin in het CMS. CpMyAdmin dient gebruikt te worden vanwege het feit dat er bij het toevoegen van tabellen of velden aan de database, ook metadata in de s-tabellen moet worden bijgehouden.

3.3.3 Objectrelationele mapping

In de module CpMyAdmin is het mogelijk om elke tabel te koppelen aan een zogenaamd DataObject. Dit is een PHP-klasse die de klasse CmsDataObject in CBB overerft. In deze klasse kunnen nuttige methodes worden opgenomen die de data uit de database verwerken. Dataobjecten in het



Figuur 4: De module CpMyAdmin om de databasestructuur binnen het CMS aan te passen.

CMS maken een objectrelationele mapping tussen PHP en MySQL mogelijk. Bij ontwikkeling van websites wordt nooit gebruik gemaakt van directe MySQL-query's. Alles verloopt via de databaseklasse CmsDB, een klasse die methodes bevat om lijsten van objecten te genereren uit het resultaat van een databasequery. De methode zal dan achter de schermen query's naar de database uitvoeren, rechten controleren, en diverse andere controles uitvoeren, en de dataobjecten hiermee instellen. Wanneer een lijst via de databaseklasse wordt opgevraagd, dan levert deze een array op met daarin de dataobjecten die aan de eisen (WHERE-clause) voldoen. Via methodes binnen de klassen van dataobjecten is het mogelijk om waarden van velden te lezen, te veranderen, en op te slaan in de database. Maar ook is het mogelijk om in deze klassen andere nuttige methodes te implementeren, bijvoorbeeld prijzen berekenen in dataobjecten van producten in een webwinkel, of de HTML om bij een nieuwsbericht automatisch een formulier te tonen waardoor je op het nieuwsbericht kan reageren.

4 Workflow

Een workflow is een opeenvolging van onderling verbonden activiteiten. Het Nederlandse woord voor workflow is "werkstroom", maar in dit document houden we de Engelse vertaling aan omdat deze in de praktijk vaker wordt gebruikt en bekender is. Workflows zijn gebaseerd op bedrijfsprocessen. Een instantie van een workflow is de uitvoer van het bedrijfsproces. Workflow kan worden gezien als een verzameling toestanden en transities. Er zijn wachttoestanden, waarbij een proces actie van een externe actor nodig heeft. Een transitie is een actie die tot gevolg heeft dat wordt overgegaan van de ene toestand naar de andere toestand. In de definitie van de workflow is vastgelegd welke transities er tussen toestanden mogelijk zijn, welke deelnemers deze transities triggeren, welke in- en uitvoerdata toestanden opleveren en de eisen die aan de uitvoering van taken worden gesteld. De volgorde van de activiteiten kan beïnvloed worden door bepaalde data, zoals bijvoorbeeld een vastgestelde hoeveelheid voorraad producten. Deze data wordt ook wel workflow relevante data genoemd. [7] [8]

4.1 Specificatie van resources, rollen en kwalificaties

Resources zijn medewerkers van de afdelingen in het bedrijf, software-resources of machines. Medewerkers hebben allemaal verschillende kwalificaties, en krijgen dus allemaal een eigen rol in het proces. In het workflow-model krijgen de individuele resources een eigen gerelateerde rol aangewezen in een organisatiemodel. Een workflow-manager bepaalt tijdens de uitvoering van het proces wie welke taken uitvoert. Deze keuze wordt gemaakt op basis van de kwalificaties van de resource. [9]

4.2 Specificatie van data

De vereiste data moet gespecificeerd worden voor elke functie. Daarnaast moet de data die de procesflow beïnvloedt (workflow relevante data) worden geïdentificeerd. Deze relevante data moet worden gedeeld met de verschillende activiteiten binnen de workflow die deze data nodig hebben. Dat kan bijvoorbeeld een verwijzing zijn naar een factuur die ergens staat opgeslagen. Maar in complexe gevallen kan het voorkomen dat het workflow-model de integriteit van bruikbare data moet beveiligen. In dat geval moet alle data die nodig is voor het proces worden gedefinieerd. Dat vereist een complexere structuur, omdat gerelateerde datatypes en entiteitstypes moeten worden geïmplementeerd. [9]

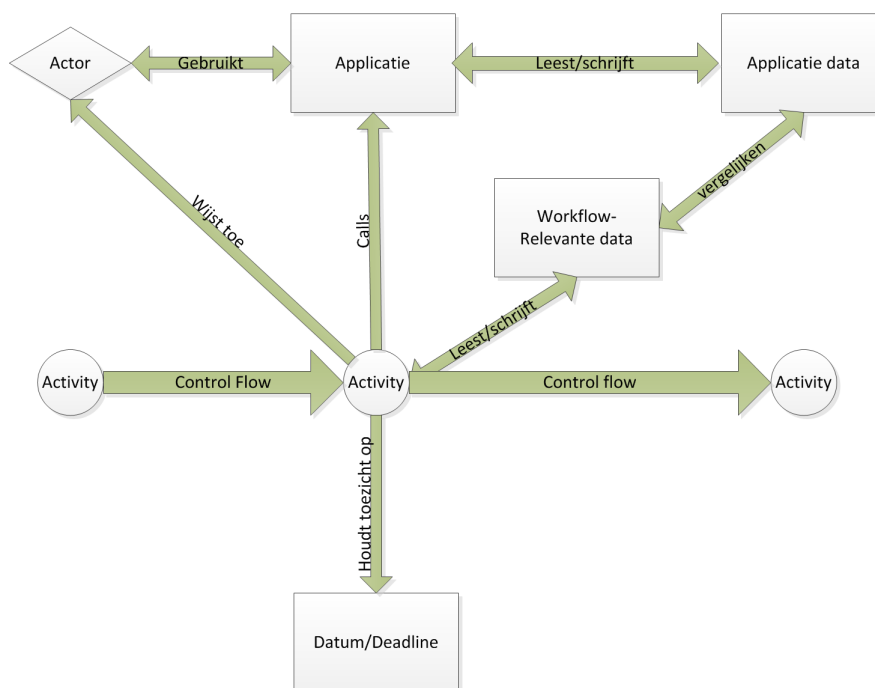
4.3 Beschrijving van workflow-condities

Soms kunnen in een workflow model keuzes gemaakt worden die afhangen van het antwoord op een bepaalde vraag. Dit zijn condities voor branching (branching is splitsen). In een bedrijfsproces-model is een conditie zoals 'controleren van tegoed door directeur' voldoende, maar deze conditie moet in het workflow-model worden vertaald in een gedetailleerd criterium zoals 'tegoed is meer dan ... dollar'. De condities voor branching in een proces model moeten beschreven worden op een manier zodat zij automatisch geëvalueerd kunnen worden. [9]

4.4 Vaststellen van uitvoerders, rechten

Voor elke activiteit moet worden vastgesteld wie de rechten heeft om de activiteit uit te voeren. Dat kan worden bereikt m.b.v. rollen, een concept dat personen scheidt van de taken die ze uitvoeren in het workflow model. Rollen zijn gedefinieerd voor elke activiteit. De rol moet worden toegewezen aan de uitvoerende persoon (bijv. spreekt Duits, mag beslissingen nemen over meer dan 30000 dollar, lid van verkoopafdeling). Tijdens de uitvoering wordt deze informatie vergeleken met de aanwezige uitvoerders, en de mensen die voldoen aan de rol zullen worden geïnformeerd over de taken die in de wachtrij staan. Wanneer een workflow-model wordt uitgevoerd, dan worden werknemers met een specifieke rol allen geïnformeerd over de status van lopende taken

via zogenaamde worklists. Als meer dan één werknemer geautoriseerd is om dit werk te doen, verdwijnt de taak uit de worklist zodra een werknemer één van deze jobs selecteert. Het workflow-managementsysteem logt de start- en eindtijd van taken. Deze log wordt de audit-trail genoemd. [9]



Figuur 5: Workflow activiteitendiagram, waarbij het verband te zien is tussen het verloop van de verschillende activiteiten en de acties die bij een activiteit ondernomen worden. [9]

4.5 Workflow-management

Workflow-management is het beheersen van workflows. Het zorgt ervoor dat uitvoer van workflow-taken volgens de regels van de workflow-definitie van de ene bij de andere afdeling van een bedrijf terecht komt. In een moderne bedrijfsinfrastructuur is het een kernonderdeel omdat het de mogelijkheid biedt om de uitvoer van kritieke bedrijfsprocessen te automatiseren. Maar in deze context hoeft automatisering niet per definitie digitaal te zijn. [8]

4.6 Geschiktheid van bedrijfsprocessen voor workflow-management

Niet elk bedrijfsproces is geschikt voor workflow-management. Om potentiële processen voor het workflow model te selecteren, moeten de processen met de hoogste succeskans voor implementatie en beste economische belofte worden gekozen. Een veelgebruikte methode in de bedrijfskunde is om een meerdimensionale criteriatalogus op te stellen. Processen worden op criteria geëvalueerd of ze geschikt zijn voor gebruik in de workflow. De criteriatalogus heeft drie evaluatie categorieën:

- Gestructureerde mogelijkheden van het proces (bijv. samenwerkende functies in proces)
- Organisationele condities binnen het framework (strategische belang, beschikbaarheid van tijd en kennis, innovatie, verplichting voor co-determinatie, en bestaande documentatie)

- Mogelijke voordelen van workflow-automatisering voor het proces (bijdrage aan bedrijfsdoel, noodzaak voor het ondernemen van actie, minder runtime, meer transparantie, routinetaken reduceren, modularisatie)

Elk elementair criterium (bijvoorbeeld het aantal deelnemende gebruikersafdelingen) heeft een weegfactor. Het elementaire criterium is opgesomd in een groep die op zijn beurt weer gewogen is (bijv. documentatie van het proces). Ten slotte zijn de groepen verzameld in een individuele evaluatiecategorie. Deze worden ook los gewogen. Het resultaat is een criteriacatalogus. [9].

4.7 Workflow-managementsysteem

Een workflow-managementsysteem (WfMS) is een applicatie die het automatiseren van een gedeelte of geheel van bedrijfsprocessen door middel van een computer mogelijk maakt. Het doel van een workflow is het geautomatiseerd uitvoeren van een taak waarbij de stappen tussen de verschillende activiteiten gecontroleerd worden door een WfMS. Een WfMS kan processen inrichten in een workflow, zodat managers de processen beter kunnen volgen en sturen, en zo de doorlooptijd binnen een organisatie kunnen verbeteren. Een WfMS biedt de mogelijkheid om bedrijfsprocessen en de problemen in processen inzichtelijk te maken, maar het lost deze problemen niet op. Een belangrijke eis aan een WfMS is dat wanneer een verandering in bedrijfsprocessen plaatsvindt, het WfMS effectieve ondersteuning levert voor naadloze verwerking van deze veranderingen. [7] [10]

4.7.1 Mogelijkheden van workflow-managementsysteem

Het voornaamste doel van workflow-managementsystemen is om coördinerende taken over te nemen in de procesuitvoering. Verder biedt een workflow-managementsysteem organisatie de volgende mogelijkheden:

- Omslag van functioneel naar procesgericht werken: door de workflows binnen de organisatie van begin tot eind in detail in kaart te brengen, en in een WfMS onder te brengen, kan een omslag worden bereikt van functioneel naar proces- en klantgericht werken.
- Routeren van werk: in het workflow-managementsysteem wordt de procesvolgorde vastgelegd, waardoor het WfMS vooraf al weet wie de volgende schakel in het proces is en tevens bijhoudt welk deel van het werk is voltooid.
- Reduceren van kosten: door tijdwinst en meer structurering in het werk, en het verbeterde bedrijfsproces, bespaart een bedrijf kosten.
- Data-redundantie: door het vaste proces wat wordt doorlopen, kan verwerkte data beter worden geïdentificeerd en geanalyseerd.
- Het verminderen van doorlooptijden: doordat de organisatie de processen organiseert, in plaats van de afdelingen, komt het proces centraal te staan waardoor scheidingen rondom diensten en afdelingen vervagen. Daardoor kan de doorlooptijd van aanvragen sterk afnemen en kunnen klanten doorgaans veel sneller geholpen worden.
- Betere toewijzing van resources: het workflow-managementsysteem kent de capaciteiten van de medewerkers in kwantitatieve zin (aantal werkuren) en kwalitatieve zin (kennis, vaardigheden, autorisatie). Hierdoor kunnen de capaciteiten beter worden benut en kan de werkdruk beter worden gespreid. Werkzaamheden kunnen parallel worden uitgevoerd, wat de doorlooptijd en het risico op bottlenecks vermindert.
- Gebalanceerde werkverdeling: het automatisch toewijzen van activiteiten aan werknemers met een bepaalde rol, leidt tot een gebalanceerde werkverdeling over alle werknemers met dezelfde rol.
- Eliminatie van routine-taken: door routine-taken te elimineren, bijv. het afhandelen van e-mail, kunnen werknemers zich concentreren op hun kernactiviteiten.

- Automatisch opschalen: door gebruik te maken van specifieke regels kan het workflow-managementsysteem een aanvraag automatisch opschalen naar een hoger of meer specialistisch niveau in de organisatie. Bijvoorbeeld wanneer het een vergunningaanvraag betreft, die door de reguliere afdeling niet kan worden afgehandeld.
- Tonen van de voortgang van aanvragen: ieder moment kun je zien waar een aanvraag zich in het proces bevindt, en voor hoelang. Zodoende kan men de voortgang in de gaten houden. Eventueel kan er op basis van historische gegevens aan de klant een indicatie gegeven worden wanneer het product geleverd zal worden.
- Operationele informatie over het proces: per aanvraag of voor een verzameling aanvragen kan de (gemiddelde) doorlooptijd berekend worden, of de werkvoorraad / werkdruk per afdeling of medewerker getoond worden. [8]

4.7.2 Beperkingen van een workflow-managementsysteem

Workflows kunnen zeer gecompliceerd zijn. Er zijn een aantal factoren die de complexiteit beïnvloeden: de angst om stakeholders uit te sluiten van het project, de angst om de controle te verliezen, en te grote beloftes als het bedrijfsproces niet goed in elkaar blijkt te zitten, en het workflow-proces daar de schuld van krijgt. Het is belangrijk om bedrijfsprocessen voorafgaand aan een workflow-ontwerp eerst op elkaar af te stemmen, te optimaliseren en vereenvoudigen, om de complexiteit van de workflow minder te maken. Ook moet een workflow ontwerp niet te specifiek gericht zijn op gevallen die zelden voorkomen. Het is beter om deze gevallen buiten beschouwing van workflow te houden, en handmatig te verwerken.

Als een uitzonderlijke situatie zich voordoet in de uitvoering van een workflow (bijv. geen enkele werknemer kiest een taak binnen een bepaalde tijd, taak geretourneerd of een ongeldige returncode), dan is er een alarmerende procedure nodig. In dat geval wordt de verantwoordelijke persoon geïnformeerd. Deze persoon kan dan zelf actie ondernemen. [7]

[11]

4.7.3 Voorbeeldtoepassingen voor workflow

Processen die goed te automatiseren zijn via WfMS zijn bijvoorbeeld:

- proces-standaardisatie. Bijvoorbeeld het proces van product- en procesontwikkeling, inkoop en verkoop, financiën en personeelszaken-standaardisatie.
- Procesverbetering. Bijvoorbeeld identificatie, ontwerp en ontwikkeling van waarde-gedreven processen. Ook het identificeren van niet rendabele activiteiten, en eliminatie daarvan behoort hiertoe.
- Web-toegankelijke processen: online, selfservice, klantenrelatie management, inventarismanagement.
- Productielijn herstructurering: bijvoorbeeld vereenvoudiging van productielijn, onderhandelen binnen de tijd, ontwikkelen van samenwerkende informatiesystemen.

4.7.4 Referentiemodel

Door het WfMC (Workflow-Management Coalition) is een standaard voor workflow-managementsystemen in het leven geroepen die het Referentiemodel wordt genoemd. Dit Referentiemodel bevat onder andere de volgende onderdelen die essentieel zijn:

- Engine. Deze beheert de in het workflow-proces gebruikte controldata en de proces-specifieke data.
- Procesdefinitie-tool. Dit definieert de mogelijke paden door het model van het bedrijfsproces.

- Clients die onderdelen van de workflow uitvoeren.
- Applicaties die aan te roepen zijn vanuit de engine.
- Administratie en monitoring tools om het workflow-proces te controleren.

[11] [10]

4.7.5 Workflow applicatie

Een workflow applicatie is een combinatie van een WfMS met een ingebouwd applicatiesysteem. De applicatie wordt gebruikt om het workflow proces te ondersteunen. Het biedt een softwaremogelijkheid om processen te automatiseren. Sommige processen vereisen menselijke activiteit, zoals goedkeuren, of het schrijven van een stuk tekst.

Een applicatie in een bestaande softwareoplossing die een rol speelt bij een bedrijfsproces kan worden aangeroepen tijdens het workflow-proces. In sommige gevallen levert een aangeroepen applicatie gebruikers de mogelijkheid om taken uit te voeren. Dit kan bijvoorbeeld een tekstverwerker zijn, maar ook een mainframe applicatie of een transactie in het systeem. Er wordt eventueel data uitgewisseld tussen het WfMS en de aangeroepen applicatie. [9]

4.7.6 Softwarepakketten voor workflow

Er zijn op de markt een aantal open- en closed source softwarepakketten beschikbaar die gespecialiseerd zijn in Workflow. Deze pakketten kunnen integreren met andere applicaties zoals een CMS, om daarmee workflow functionaliteit aan te bieden. De meesten daarvan hebben een XML gebaseerde syntax waarin de workflow is opgesteld, en een API die events kan ontvangen, en externe processen kan triggeren. Door het WfMC is een standaard gepubliceerd voor workflow, XPDL. Dit is een XML schema die door diverse workflow-pakketten wordt geaccepteerd. Of in de praktijk daadwerkelijk deze schema's op de verschillende workflow-pakketten hetzelfde uitpakken is de vraag, maar in theorie zou het overschakelen naar een ander WfMS niet meer moeten inhouden dan het XPDL-bestand verplaatsen. In tegenstelling tot veel workflow-applicaties, zijn er in vergelijking zeer weinig CMS'en die XPDL accepteren. De oorzaak hiervan zal waarschijnlijk liggen in het feit dat workflow bij kleinere bedrijven niet de interesse wekt of totaal onbekend is bij ondernemers. Grotere bedrijven hebben over het algemeen een losstaande applicatie voor workflow die niet gebonden is aan het CMS. [7]

4.7.7 Workflow in een Content Managementsysteem

De definitie van workflow binnen een CMS is niet eenduidig. Tot op zekere hoogte bevat elk CMS een aantal functionaliteiten die tot een workflow behoren. Zoals het sturen van een mailtje nadat een reactie onder een nieuwsbericht is geplaatst. Maar dat houdt natuurlijk geen volledige ondersteuning van de workflow standaard in. Deze ondersteuning is vrijwel nooit aanwezig. Het CMS delegeert in het algemeen geen taken, en het CMS volgt geen formeel proces bij een uit te voeren taak. [7]

Referenties

- [1] Content-Power, “Informatie over het bedrijf.” <http://www.contentpower.nl>.
- [2] W3Schools, “Php, definitie.” http://www.w3schools.com/php/php_intro.asp.
- [3] W3Schools, “Javascript, definitie.” http://www.w3schools.com/js/js_intro.asp.
- [4] W3Schools, “jquery, definitie.” http://www.w3schools.com/jquery/jquery_intro.asp.
- [5] W3Schools, “Html, definitie.” http://www.w3schools.com/html/html_intro.asp.
- [6] W3Schools, “Cascading style sheets, definitie.” http://www.w3schools.com/css/css_intro.asp.
- [7] ContentHere, “Workflow in cms and beyond.” <http://www.contenthere.net/2004/12/workflow-in-cms-and-beyond.html>.
- [8] Passionned-Group, “Workflow management (wfm).” <http://www.passionned.nl/workflow-management/>.
- [9] J. Becker, M. Kugeler, and M. Rosemann, eds., *Process management: a guide for the design of business processes*. 2003.
- [10] W. B. Rouse and A. P. Sage, *Work, Workflow and Information Systems*. Amsterdam, The Netherlands, The Netherlands: IOS Press, 2007.
- [11] J. Caverlee and et al., “Workflow management for enterprise transformation,” 2007.

C Plan van Aanpak

Plan van Aanpak - Bachelorproject IN3405
Workflow-management in een Content Managementsysteem

Rob Ruigrok
R.J.Ruigrok@student.tudelft.nl
1371746

Jan Pieter Mars
J.P.Mars@student.tudelft.nl
1381075

7 december 2011



ContentPower
informatiebeheersystemen

Voorwoord

Dit document dient als Plan van Aanpak voor het bachelorproject van Rob Ruigrok en Jan Pieter Mars betreffende de studie Technische Informatica aan de Technische Universiteit Delft. Het beschrijft in grote lijnen de opdracht en de tijdsplanning. Het bachelorproject zal 10 weken beslaan en het eindresultaat is de implementatie van een framework voor workflow-management in een bestaand Content Managementsysteem. Het bachelorproject wordt uitgevoerd voor het bedrijf Content Power in Delft. De bedrijfsbegeleider vanuit Content Power is ir. W.F. van Valkenburg, en vanuit TU Delft is de begeleider drs. P.R. van Nieuwenhuizen.

Samenvatting

Content Power heeft een bestaand Content Managementsysteem (CMS). Dit CMS is een framework waarin modules zijn geïmplementeerd. Het is de bedoeling om in dit bestaande framework een nieuwe module te bouwen voor workflow-management. Binnen deze module kunnen bedrijfsprocessen worden gemodelleerd volgens de ideeën zoals deze te vinden zijn in het oriëntatieverslag. De nadruk van de opdracht zit in het bouwen van het framework, met de onderdelen die beschreven staan in het Referentiemodel. Indien er later nog voldoende tijd beschikbaar is, kan er voor gekozen worden om interfaces voor een extra aantal communicatiemogelijkheden (bijvoorbeeld het sturen van e-mail, sms), triggers (activatie van bedrijfsproces vanwege een inkomende mail, actie van gebruiker op website, etc.), of import- en exportfunctionaliteit in de vorm van XPDL mogelijk te maken. Maar deze extra's horen uitdrukkelijk niet tot de aanvankelijke opdracht: dat is alleen het bouwen van een module / framework voor workflow-management.

Inhoudsopgave

1	Introductie	53
2	Projectopdracht	53
2.1	Projectomgeving	53
2.2	Doelstelling project	53
2.3	Opdrachtformulering	53
2.4	Op te leveren diensten (deliverables)	53
2.5	Eisen en beperkingen	54
2.6	Voorwaarden	54
3	Interviews met klanten	54
3.1	Vragen voor het interview	54
3.2	Verstuurde uitnodiging voor het interview naar de klanten	55
4	Aanpak en tijdsplanning	57
4.1	Definitiestudie/analyse	57
4.2	Opzetten van een framework voor workflow-management	57
4.3	Aanpak	58
4.3.1	Oriëntatie	58
4.3.2	Plan van Aanpak	58
4.3.3	Requirements	58
4.3.4	Ontwerp	58
4.3.5	Implementatie	58
4.3.6	Testen	58
4.3.7	Afronding	59
4.3.8	Onderhoud	59
4.4	Tijdsplanning	59
5	Projectinrichting en voorwaarden	61
5.1	Organisatie	61
5.2	Rapportering	61
5.3	Administratie	61
5.4	Techniek, software, huisvesting	61
6	Kwaliteitsborging	61
6.1	Proceskwaliteit	61
6.2	Productkwaliteit	61

1 Introductie

Het bedrijf Content Power levert websites. Voor de klanten worden systemen geleverd waarmee zij hun website kunnen bijhouden. Verder kunnen er op maat gemaakte systemen worden gebouwd zoals een intranet, inschrijfformulieren, koppelingen met andere bedrijven, etc. De vraag van Content Power is of klanten mogelijk meer integratie van ICT in hun bedrijf wensen, door gebruik te maken van een systeem dat de communicatie tussen verschillende afdelingen en personen kan managen. Een van de meest voor de hand liggende oplossingen hiervoor is een zogenaamde workflow-managementsysteem (WfMS) te bouwen. Dit is een systeem waarmee bedrijfsprocessen kunnen worden gemodelleerd en uitgevoerd. Dergelijke systemen hebben veel mogelijkheden en voordelen, die staan genoemd in het oriëntatieverslag. Een WfMS maakt het mogelijk om meer bedrijfsprocessen bij klanten te automatiseren. Ook voor Content Power biedt dit voordelen. Het kan namelijk de implementatie van veel routine- en maatwerk besparen als sommige taken door de workflow-module worden uitgevoerd.

Voor het verdere onderzoek naar workflow-oplossingen zijn er een aantal interviews gepland met grotere klanten van Content Power. Wij gaan op locatie bij deze bedrijven langs, om de verantwoordelijke personen vragen te stellen over de mogelijke toepassingen en verbeteringen in hun bedrijfsprocessen. Aan de hand van deze interviews werken wij extra use cases uit, en gaan wij na welke processen in aanmerking komen voor automatisering d.m.v. workflow-management.

Dit alles heeft geleid tot het formuleren van de projectopdracht, en het effectueren ervan. Hiermee is tevens dit Plan van Aanpak tot stand is gekomen.

2 Projectopdracht

Hieronder volgt een korte beschrijving van de projectopdracht. Een uitgebreidere beschrijving is te vinden in de officiële opdrachtbeschrijving die is aangeleverd door Content Power.

2.1 Projectomgeving

Het bestaande Content Managementsysteem wordt uitgebreid met een module die workflow-management mogelijk maakt. Het is de bedoeling dat deze module als een integraal onderdeel van het CMS gaat werken, niet als een onafhankelijk systeem dat naast het CMS draait.

2.2 Doelstelling project

Het doel van het project is het toevoegen van ondersteuning voor volledig workflow-management in het CMS van Content Power. In feite breiden we het CMS uit zodat het zowel een CMS als WfMS genoemd kan worden.

2.3 Opdrachtformulering

Er dient een module voor workflow-management geïmplementeerd te worden die het mogelijk maakt om workflow-processen te modelleren in het CMS, en deze processen uit te voeren. De workflow-module moet acties kunnen uitvoeren, interfaces hebben om te communiceren, en interfaces om input te ontvangen. Als er dan nog tijd over is, zullen een aantal communicatiemogelijkheden en triggers worden geïmplementeerd.

2.4 Op te leveren diensten (deliverables)

Bij afronding van de projectopdracht worden onder andere de volgende deliverables opgeleverd:

- Oriëntatiedocument (informatie over Content Power en workflows)
- Plan van Aanpak (projectplan en planning)

- Requirements Analysis Document (systeemeisen, interviews en use cases)
- Technical Design Document (gedetailleerde systeemontwerp)
- Testplan en testresultaten
- Implementatie (programmeren van framework met uitbreidingen)
- Codebeoordeling SIG
- Documentatie van de broncode
- Documentatie voor Content Power
- Documentatie voor de gebruiker
- Eindverslag van het project (reflectie)

2.5 Eisen en beperkingen

Het resultaat van de opdracht is een WfMS ingebouwd in het bestaande CMS. Workflow-processen moeten gemodelleerd kunnen worden, en er moeten interfaces geïmplementeerd zijn voor de triggers en communicatie. De prioriteiten tijdens het project liggen vooral bij het bouwen van een goede basis voor het workflow-managementsysteem, zodat later eenvoudig de functionaliteit uitgebreid kan worden.

2.6 Voorwaarden

De bedrijfsbegeleider houdt toezicht op het project, en geeft regelmatig feedback. Verder wordt er rekening gehouden met alle kennis die is opgedaan in de voorgaande fases. Het resultaat moet voldoen aan de door Content Power gestelde eisen. Dit zijn eisen die gerelateerd zijn aan de codestijl en op te leveren documenten. De eisen staan uitgebreid beschreven in de opdrachtbeschrijving. Het eindproduct is een stabiel werkende workflow-module, geïntegreerd in het CMS, inclusief documentatie voor praktisch gebruik door Content Power.

3 Interviews met klanten

Bij aanvang van het project moeten de wensen van de (potentiële) klanten van Content Power in kaart worden gebracht. Dit gebeurt door middel van interviews te houden bij een aantal grote klanten van Content Power waarvoor recent projecten zijn uitgevoerd. De opzet van het interview is per klant verschillend, vanwege verschillen in de aard en complexiteit van de bedrijfsprocessen die een rol spelen. Het doel van het interview is om te weten te komen welke processen een rol spelen binnen hun bedrijf, en in hoeverre deze processen in een WfMS opgenomen kunnen worden. Ook wordt nagegaan welke eisen een klant aan een WfMS stelt, en of er extra wensen zijn. Van alle geïnterviewde klanten zal er een worden geselecteerd. Het voorgestelde workflow-proces van deze klant zal uitgewerkt in usecases, en geïmplementeerd worden. Aan het einde van het project zal een gebruikerstest plaatsvinden door de geselecteerde klant.

3.1 Vragen voor het interview

Hieronder staan een aantal standaardvragen die we aan de klant kunnen stellen:

1. Welke afdelingen zijn er momenteel actief binnen het bedrijf?
2. Welke communicatiemiddelen worden er gebruikt?
3. Welke processen voeren de verschillende afdelingen uit, en hoe zien deze processen er uit?

- (a) Waardoor wordt het proces gestart?
 - (b) In welke stappen verloopt het proces?
 - (c) Wie zijn er betrokken bij het proces?
 - (d) Wat is het einddoel van het proces?
 - (e) Wie heeft de verantwoordelijkheid voor de uitvoer van specifieke processen?
 - (f) Wat gebeurt er als een proces niet uitgevoerd kan worden?
4. Wordt het als een voordeel gezien als een computersysteem deze processen kan begeleiden?

3.2 Verstuurde uitnodiging voor het interview naar de klanten

Op de volgende pagina is de tekst van de uitnodiging opgenomen die is uitgedeeld aan een aantal klanten van Content Power. Deze uitnodigingen zijn overhandigd tijdens het Symposium van Content Power op 16 september 2011 toen het bedrijf zijn 10-jarige jubileum vierde. Er zijn uitnodigingen overhandigd aan de volgende klanten:

- Kees van Rodenburg (Vocar, Vakopleiding Carrosseriebedrijf, in Sassenheim)
- Marcel Bon (NEVI, Nederlandse Vereniging voor Inkopers in Zoetermeer)
- David Lansen (Evenementenbureau Delft, Ondernemersbelangen Delft, Vermeercentrum Delft)
- Robbert Lasoe (Stylentio, koop en verkoop van fashionproducten, in Maastricht)

Er wordt later nog contact met deze klanten opgenomen om een definitieve afspraak te plannen.

Stageopdracht: automatiseren van bedrijfsprocessen d.m.v. workflow



Betreft: Interview bachelorproject

Voor ons bachelorproject aan de TU Delft voeren wij een stageopdracht bij Content Power uit. Onze taak is het om een zogenaamd workflowsysteem te ontwerpen en te bouwen, en als onderdeel in CMS 4 te integreren. Hiermee kunnen bedrijfsprocessen worden geautomatiseerd. Een deel van bedrijfsprocessen kan worden beschreven in de vorm van een zogenaamde "Workflow". Een workflow is het proces waarbij verschillende afdelingen in een bedrijf, medewerkers, externe factoren, etc. samenwerken om tot een eindresultaat te komen.

Verschillende processen van product- en procesontwikkeling, inkoop en verkoop, productielijn, voorraadmanagement, monitoring, webtoegankelijke processen, klantenrelatiemanagement, zelfservice, financiën en personeelszaken-standaardisatie kunnen allemaal in een workflow worden opgenomen.

Een voorbeeld van een workflow is bijvoorbeeld als een medewerker van een bedrijf maandelijks een onkostendeclaratie invult. Deze declaraties zijn bepaalde formulieren die via de website worden ingevuld. Na het invullen worden de formulieren opgeslagen. Zodra de onkosten in een declaratie een bepaald bedrag overschrijden, zal er automatisch een e-mailbericht naar een manager worden verzonden die de declaratie vervolgens kan controleren. Zodra de manager in die mail bevestigt dat de declaratie in orde is, komt de declaratie automatisch bij de administratie terecht die de overboeking uitvoert. In het systeem wordt door de administratie deze declaratie gemarkeerd als afgehandeld. Voor het ontwerp van het workflowsysteem proberen wij meer inzicht te krijgen in verschillende bedrijfsprocessen die bij klanten van Content Power een rol spelen. Daarom zouden wij u graag willen interviewen.

Alvast hartelijk bedankt!

Met vriendelijke groeten,

Jan Pieter Mars en Rob Ruigrok
(e-mail: R.J.Ruigrok@student.tudelft.nl / J.P.Mars@student.tudelft.nl)

4 Aanpak en tijdsplanning

4.1 Definitiestudie/analyse

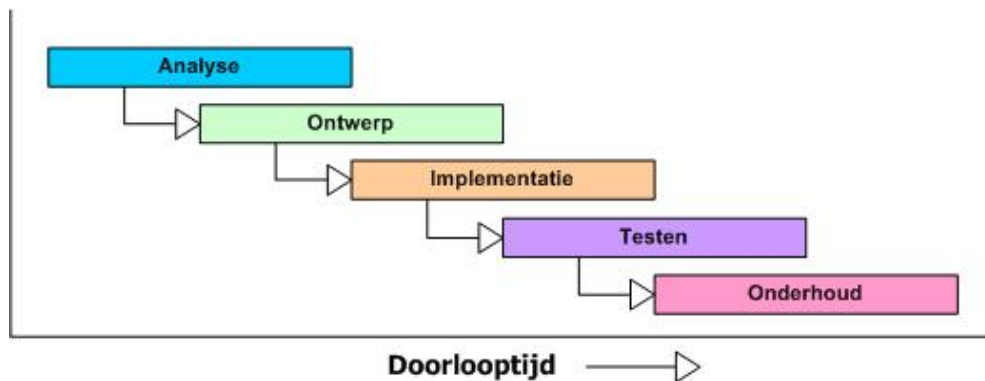
Voor ons bachelorproject aan de TU Delft voeren wij een stageopdracht bij Content Power uit. Onze taak is het om een zogenaamd workflow-systeem te ontwerpen en te bouwen, en als onderdeel in CMS 4 te integreren. Hiermee kunnen bedrijfsprocessen worden geautomatiseerd. Een deel van bedrijfsprocessen kan worden beschreven in de vorm van een zogenaamde "workflow". Een workflow is het proces waarbij verschillende afdelingen in een bedrijf, medewerkers, externe factoren, etc. samenwerken om tot een eindresultaat te komen.

4.2 Opzetten van een framework voor workflow-management

Binnen het bestaande framework van Content Power moet een framework en datamodel worden gemaakt waarin processen uit een workflow kunnen worden gemodelleerd. Bij het bouwen van het framework moet er vooral worden gericht op de eenvoudigere processen die dichtbij het beheren van een website of webwinkel liggen. Binnen het CMS van Content Power moet een module komen waarin de processen kunnen worden samengesteld. De module zal voorlopig alleen door Content Power zelf worden gebruikt. Het samenstellen van processen moet zonder fouten kunnen, maar mag enige technische kennis vereisen. Het gebruik van de workflow-module door Content Power moet in dat geval worden beschreven in de documentatie voor Content Power.

4.3 Aanpak

Voor dit project wordt gebruik gemaakt van het "sashimi-model" voor softwareontwikkeling. Dit is een doorontwikkeling op het watervalmodel, waarbij de verschillende fasen elkaar gedeeltelijk overlappen. Het voordeel van dit model boven het watervalmodel is dat tijdens de implementatiefase de ontwerpfase niet volledig afgerond hoeft te zijn, en later nog wijzigingen in het ontwerp kunnen worden aangebracht.



Figuur 1: Het sashimi-model voor softwareontwikkeling. (bron: http://nl.wikipedia.org/wiki/Watervalmethode#Het_.22sashimi.22-model)

4.3.1 Oriëntatie

In de oriëntatiefase wordt geanalyseerd wat workflow exact inhoudt, en hoe het zich in de praktijk voordoet. De mogelijkheden en gebreken van workflow-management worden onderzocht. Verder wordt er in deze fase bekeken hoe de werkwijze van Content Power in elkaar steekt, en wat het CMS precies inhoudt.

4.3.2 Plan van Aanpak

In het PvA wordt gespecificeerd wat er gebouwd gaat worden. Het bevat tevens een gedetailleerde planning voor het gehele project.

4.3.3 Requirements

Alvorens er met implementeren wordt begonnen, worden er requirements opgesteld. Als uitbreiding op de requirements zullen ook de wensen van een aantal klanten in kaart worden gebracht. Op basis van deze informatie worden vervolgens een aantal use cases opgesteld.

4.3.4 Ontwerp

Na het opstellen van requirements zal een ontwerp voor het workflow-framework worden gemaakt. Deze bevat klassendiagrammen en specificaties die gebruikt zullen worden bij de implementatie.

4.3.5 Implementatie

Implementatie is het omzetten van het ontwerp naar werkende code in PHP.

4.3.6 Testen

Tijdens de implementatiefase, en de periode die daarna volgt, zal er voortdurend worden getest. Er wordt hiervoor gebruik gemaakt van automatische tests.

4.3.7 Afronding

Tot de afronding behoort het verwerken van alle feedback, het opstellen van het einddocument, en het voorbereiden van de presentatie.

4.3.8 Onderhoud

Onderhoud zal door Content Power zelf gepleegd moeten worden, en behoort niet tot de project-opdracht. De documentatie die wordt geschreven in de afrondingsfase kan wel van pas komen bij het plegen van onderhoud, omdat het de werking van de implementatie technisch toelicht.

4.4 Tijdsplanning

Voor de uitvoering van de opdracht hebben wij een planning opgesteld die tien weken beslaat. In week 42 is Rob een aantal dagen op vakantie. Week 44 en 45 staan niet ingepland vanwege de tentamenperiode. Bovendien is de bedrijfsbegeleider alleen via e-mail of skype bereikbaar tussen week 44 en 46.

Week Nr.	Fase	Deliverables	Bezigheden
36 (5-9 sep)	Oriëntatiefase	Oriëntatieverslag concept	Kick-off bachelorproject. Verdiepen in de theorieën van workflows, en in de werkwijze van Content Power.
37 (12-16 sep)	Oriëntatiefase	Oriëntatieverslag + PvA concept	Plan van Aanpak opstellen, feedback op Oriëntatieverslag verwerken. Vragen voor interviews met klanten opstellen, en tijdens symposium en via e-mail de klanten benaderen.
38 (19-23 sep)	Analysefase	PvA + Requirements Analysis Document concept	Feedback op Plan van Aanpak verwerken. Bedrijven benaderen voor interviews, use cases en requirements opstellen
39 (26-30 sep)	Ontwerpfase	Requirements Analysis Document + Testplan concept	Testplan opstellen, Technical Design Document opstellen waarin het ontwerp voor de implementatie staat, interviews houden
40 (3-7 okt)	Ontwerpfase	Testplan + Technical Design Document concept	Technical Design Document afronden. Aan de hand van dit ontwerp beginnen met implementatie
41 (10-14 okt)	Implementatiefase	Technical Design Document	Implementatie.
42 (17-21 okt)	Implementatiefase	Prototype #1	Implementatie. Aan het eind van de week het tussentijdse resultaat als prototype klaarzetten voor Content Power. Software Improvement Group op de hoogte stellen voor controle.
43 (24-28 okt)	Implementatiefase	Prototype #2 + Code SIG	Implementatie. Aan het eind van de week het tussentijdse resultaat als prototype klaarzetten voor Content Power. Verder wordt de code verstuurd naar Software Improvement Group
44-45 (31 okt - 11 nov)	Tentamenperiode		Tijd voor studeren van tentamens
46 (14-18 nov)	Testfase	Testresultaten	Afronding implementatie. Eindproduct klaarzetten voor Content Power, en tests uit testplan toepassen en resultaten verwerken
47 (21-25 nov)	Afrondingsfase	Eindverslag concept + documentatie + code	Verwerking van de verworven inzichten, opstellen van het eindverslag van het project. Commentaar van de Software Improvement Group verwerken in het verslag en de code. Tevens een opzet voor de presentatie maken.
48 (28 nov - 2 dec)	Afrondingsfase	Eindverslag en presentatie	Feedback op concept eindverslag verwerken. Presentatie voorbereiden, uitwerken, en presenteren.

5 Projectinrichting en voorwaarden

5.1 Organisatie

De bedrijfsbegeleider van dit project beoordeelt het uiteindelijke resultaat. Vanuit het bedrijf Content Power zal Willem van Valkenburg onze dagelijkse begeleider zijn. Begeleiding tijdens het implementeren komt van Jan Niemantsverdriet. Jan is de expert op het gebied van het bestaande CMS.

5.2 Rapportering

Elke maandagochtend staat een afspraak bij Content Power gepland, om de voortgang te bespreken. Daarnaast zal er eens in de twee weken een gesprek volgen met de begeleider vanuit de TU Delft, of wordt de tweewekelijkse vragenlijst ingevuld. Verder zullen alle op te leveren diensten (2.4) aan de begeleiders in PDF-formaat worden aangeleverd. Deze documenten zijn geschreven in L^AT_EX.

5.3 Administratie

Alle op te leveren diensten staan in een SVN-repository opgeslagen voor versiebeheer. De documenten staan opgeslagen in een repository op een server van Rob. De code maakt gebruik van een repository van Content Power.

5.4 Techniek, software, huisvesting

Gedurende de onderzoeks-, analyse- en ontwerpfase zal er vooral in een van de projectruimtes op de Drebbelweg worden gewerkt. Tijdens de implementatiefase wordt gewerkt op het kantoor van Content Power aan de Burgwal in het centrum van Delft. Voor het project wordt gebruik gemaakt van de software/tools: Subversion, Eclipse PDT, Astah Community, LaTeX. De code is geschreven in de scripttaal PHP.

6 Kwaliteitsborging

6.1 Proceskwaliteit

De bedrijfsbegeleider is continu betrokken bij het project om zo de kwaliteit van de uitvoering te waarborgen.

6.2 Productkwaliteit

Om de code bruikbaar en leesbaar te houden zal de codestijl van Content Power aangehouden worden, zodat de code na het project direct door Content Power toegepast kan worden. Daarnaast zal de Software Improvement Group feedback geven over de kwaliteit van de code. Deze feedback zal waar mogelijk worden verwerkt in de code, en er zal een reflectie plaatsvinden in het eindverslag. Het product zal grondig worden getest. Voor het testen van de code worden geautomatiseerde tests geschreven. In een gebruikerstest zal het systeem worden getest door de klant. Bovendien dient de documentatie correct te zijn. Alle genoemde punten waarborgen de productkwaliteit.

D Requirements and Analysis Document

Requirements Analysis Document - Bachelorproject IN3405 Workflow-management in een Content Managementsysteem

Rob Ruigrok

R.J.Ruigrok@student.tudelft.nl

1371746

Jan Pieter Mars

J.P.Mars@student.tudelft.nl

1381075

7 december 2011



ContentPower
informatiebeheersystemen

Voorwoord

Dit document beschrijft de eisen die gesteld worden aan het te bouwen systeem. Om tot deze eisen te komen is een uitgebreid praktisch onderzoek uitgevoerd naar workflows. Bijna op elke denkbare plek vinden workflow-processen plaats. Thuis, bij bedrijven, in verenigingen, en ga zo maar door. Soms zijn deze processen bewust, en soms onbewust. Als onderdeel van het verzamelen van eisen, worden interviews gehouden met managers uit een aantal bedrijven. Verder worden een aantal bekende fictieve voorbeelden van workflow-processen geanalyseerd.

Samenvatting

In dit document wordt een analyse gemaakt van een aantal verschillende workflow-processen die in de praktijk veel voorkomen, uit welke taken ze bestaan, en hoe deze taken onderling van elkaar afhankelijk zijn. Hieruit worden vervolgens usecases afgeleid. Er is een lijst met eisen in MoSCoW-vorm (Must, Should, Could, Won't) opgesteld. Deze lijst wordt tijdens de implementatiefase gehanteerd om prioriteiten aan de verschillende te bouwen delen van het systeem toe te kennen, zodat de belangrijkste onderdelen geïmplementeerd worden binnen de beschikbare tijd die er is voor deze opdracht. Eerst worden de must-haves geïmplementeerd, daarna de should-haves, etc.



Figuur 1: MoSCoW [?]

Inhoudsopgave

1	Huidige situatie	67
1.1	Bedrijfsprocessen	67
1.2	Procesmanagers	67
1.3	Communicatie	67
2	Het te bouwen systeem	67
2.1	Overzicht	67
3	Interviews	68
3.1	Interview met Marcel Bon, Manager Marketing & Communicatie bij Nevi	68
3.2	Interview met David Lansen, Projectleider Evenementenbureau Delft	72
3.3	Interview met Kees Rodenburg, Senior Medewerker B&BD, VOC Sassenheim	75
4	Functionele eisen	77
4.1	MoSCoW	77
4.1.1	Must have	77
4.1.2	Should have	78
4.1.3	Could have	78
4.1.4	Won't have	78
4.1.5	Triggers, Verwerking en Communicatie	79
5	Niet-functionele eisen	80
5.1	Gebruikersinterface	80
5.2	Documentatie	80
5.3	Software-eisen	80
5.4	Prestaties	80
5.5	Foutafhandeling en extreme omstandigheden	80
5.6	Interactie tussen systemen	80
5.7	Kwaliteitseisen	80
5.8	Systeemaanpassingen	81
5.9	Beveiligingseisen	81
5.10	Beheer	81
6	Constraints	81
7	Systeemmodellen	81
7.1	Usecase modellen	81
7.1.1	Usecase diagram	82
7.1.2	Usecases	83
7.2	Klassendiagram	85

1 Huidige situatie

Hieronder wordt een vereenvoudigd beeld geschetst van een bedrijf waar geen workflow-management wordt toegepast.

1.1 Bedrijfsprocessen

In een bedrijf vinden bedrijfsprocessen plaats. Een bedrijfsproces is een keten van activiteiten die een resultaat opleveren. Dat kan een primair proces zijn dat rechtstreeks een product of dienst voor de klant oplevert, maar ook een intern proces dat andere bedrijfsprocessen direct of indirect ondersteunt.

1.2 Procesmanagers

Bedrijfsprocessen die plaatsvinden bestaan uit losse activiteiten. Deze activiteiten kunnen afhankelijk van elkaar zijn. Om de afhankelijke activiteiten met elkaar te koppelen, zijn er procesmanagers die beslissingen nemen om het proces te sturen zodat de koppeling vloeiend verloopt.

1.3 Communicatie

Activiteiten van bedrijfsprocessen worden soms door verschillende afdelingen binnen een bedrijf verwerkt, en soms kunnen meerdere afdelingen hetzelfde werk leveren. Communicatie is zeer belangrijk om de afdelingen en procesmanagers op die afdelingen op elkaar af te stemmen zodat het bedrijfsproces vlot verloopt. Deze communicatie verloopt via e-mail, (in-)formele gesprekken, telefoon, etc.

2 Het te bouwen systeem

2.1 Overzicht

Er is gevraagd om een systeem te ontwikkelen om de hierboven beschreven situatie waar mogelijk verder te automatiseren. Dit houdt in dat er een workflow-management systeem moet worden geïmplementeerd. In dit WfMS kunnen bedrijfsprocessen als workflow-processen worden gemodelleerd. Het WfMS zorgt voor een standaard aanpak voor de workflow-processen. De processen in het model zijn herbruikbaar voor verschillende gevallen (instanties) van het bedrijfsproces. Er kunnen verschillende instanties simultaan plaatsvinden. Verder kan het WfMS de status van alle instanties tonen (openstaand, in uitvoering, afgerond, etc.). In deze status is ook te zien in welke stap het workflow-proces zich bevindt. Op basis van automatische regels kunnen prioriteiten worden gesteld aan verschillende instanties van processen. Er kunnen deadlines aan instanties worden gesteld. Het WfMS is in staat om automatisch taken toe te wijzen, of mededelingen te doen aan personen of rollen. Verder beschikt het systeem over interfaces voor koppelingen aan externe informatiesystemen. Deze koppelingen kunnen tot gevolg hebben dat nieuwe of bestaande instanties van workflow-processen worden geactiveerd. Workflow-processen kunnen eenvoudige todo-processen inhouden, repeterende processen (bijv. het uitvoeren van betalingen), maar ook processen met taken voor de lange termijn (zoals doelstellingen).

Het systeem automatiseert weliswaar een aantal bedrijfsprocessen, maar het systeem kan niet alle communicatie en triggers afhandelen. Het systeem kan hoogstens een instructie geven dat een taak (bijvoorbeeld: bel persoon x en zeg y) moet worden uitgevoerd, maar de verantwoordelijke personen binnen een bedrijf moeten zelf dit telefoongesprek voeren, en achteraf in het systeem aangeven dat de taak is uitgevoerd. Bovendien moeten werknemers in het systeem meer gespecificeerd worden op hun capaciteiten dan in de oude situatie, zodat taken efficiënter kunnen worden

verdeeld. Door gebruik te maken van een workflow-management systeem kunnen een aantal bedrijfsprocessen veel efficiënter verlopen omdat routine-taken uit handen genomen kunnen worden, en mede daardoor in minder tijd meer werk kan worden verzet.

3 Interviews

3.1 Interview met Marcel Bon, Manager Marketing & Communicatie bij Nevi

Interview heeft plaatsgevonden op 26 september 2011 van 16:00 uur tot 17:45 uur, op Schiphol International Airport.

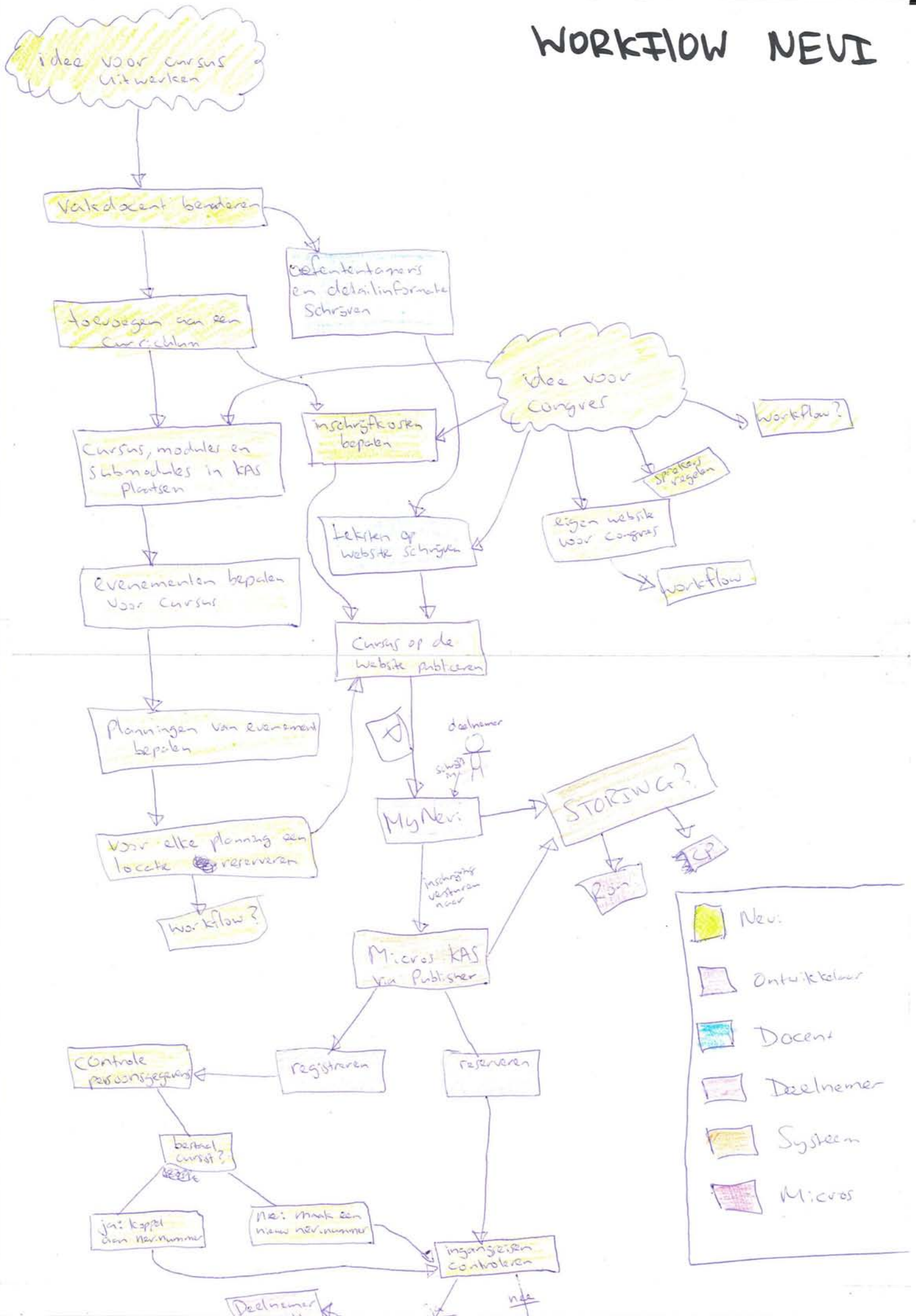
Nevi is een vereniging voor inkoop- en voorraadmanagers. Zij verzorgt branche-gerelateerde opleidingen, cursussen, congressen en trainingen. Nevi maakt gebruik van het cursusadministratiesysteem Micros KAS, en van het relatiebeheersysteem Micros RAS. De website en het CMS is geleverd door Content Power. Er is een XML-koppeling (genaamd Micros Publisher voor cursusinformatie en Micros Connector voor inschrijvingen) tussen het administratiesysteem KAS en het CMS van Content Power. Hiermee is het mogelijk om cursisten online voor cursussen in te laten schrijven, en deze inschrijvingen direct in het administratiesysteem van Nevi te verwerken.

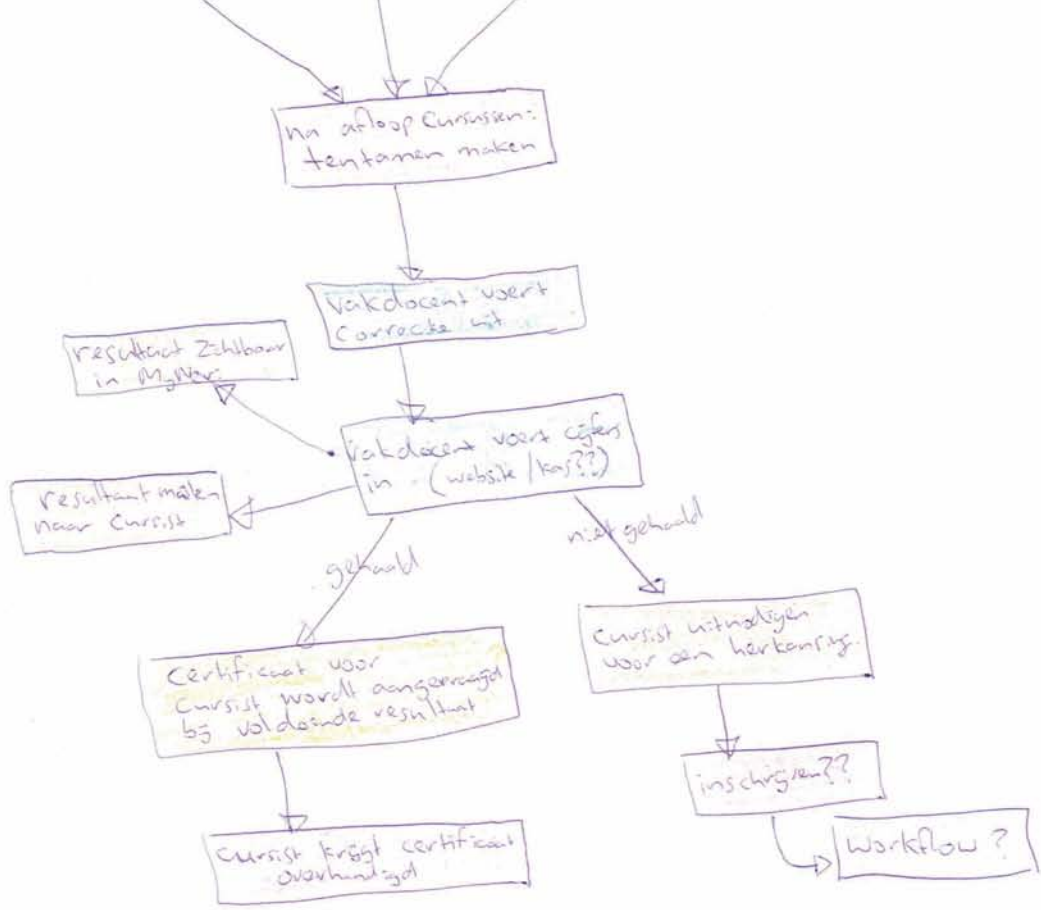
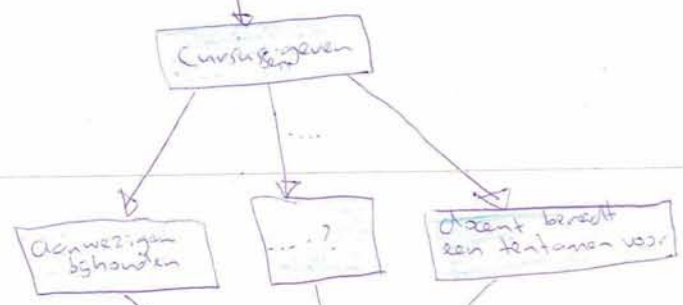
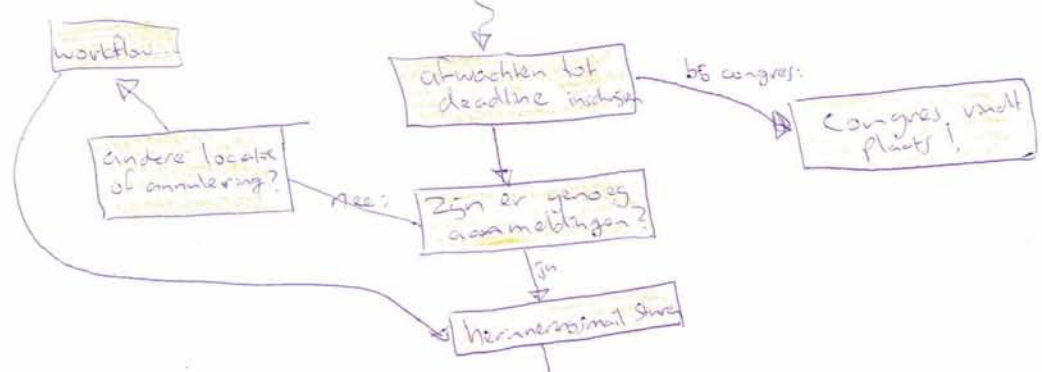
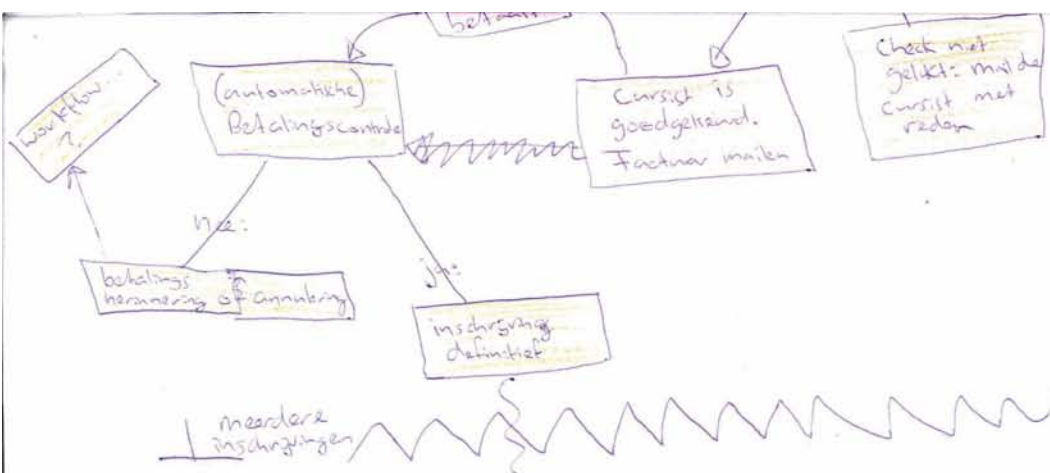
Vanwege informatie over bedrijfsvoering is een deel van het interview uit dit document gelaten.

Vanwege informatie over bedrijfsvoering is dit interview uit dit document gelaten.

Voorafgaand aan het interview hebben wij ons verdiept in workflow-processen die bij Nevi een rol zouden kunnen spelen. Op de volgende pagina staat een door ons opgesteld model van het workflow-proces van een cursus. Het interview is gehouden rondom dit model.

WORKFLOW NEVI





3.2 Interview met David Lansen, Projectleider Evenementenbureau Delft

Interview heeft plaatsgevonden op 6 oktober 2011 van 15:00 uur tot 16:30 uur, in het Vermeercentrum in Delft. Ook projectleider Susanne Dix was aanwezig en heeft deelgenomen. Het interview is begonnen met een beknopte uitleg over workflows, en een aantal praktijkvoorbeelden. David merkt op dat er bij het Evenementenbureau Delft een aantal workflows een rol spelen. Een belangrijke workflow is die van het organiseren van een project. Bij het bespreken van workflows komen een aantal wensen naar voren.

- In de praktijk moet je vaak lang wachten op een reactie waardoor proces lang stil komen te liggen. Een workflow-proces kan uitkomst bieden, omdat processen dan automatisch gestuurd kunnen worden.
- Inplannen van deadlines kost veel tijd, het zou handig zijn als dit voor zover dat mogelijk is automatisch voor je wordt gedaan door het model van het workflow-systeem. Veel deadlines zijn van elkaar afhankelijk of staan vooraf vast.
- Een systeem moet in dagelijks gebruik geen beperking vormen tijdens processen, als dingen anders gebeuren moet dat mogelijk blijven. Deze vrijheid kan als voordeel gezien worden, maar teveel vrijheid in een proces brengt nieuwe risico's met zich mee, bijvoorbeeld taken die opzij gezet worden, en later in vergetelheid raken. Ons workflow-systeem moet voor de gulden middenweg kiezen.
- Het hele traject van een workflow moet goed overzien kunnen worden. Van elk project moet zichtbaar zijn wat de status is, en waar nog werk open staat.
 - Sommige taken zijn urgenter dan andere taken. Er moet een mogelijkheid zijn om urgentie aan een activiteit te geven, zodat ze met prioriteit worden behandeld.
 - Bij elk onderdeel van een proces is een invoer en uitvoer aanwezig. Alleen dan gebeurt er iets nuttigs.
 - Een koppeling tussen het workflow-systeem en de agenda zou handig zijn. Taken komen dan automatisch in je lijst met afspraken te staan.
- Sommige mensen zijn niet enthousiast om in een nieuw systeem te werken, dat komt vaak omdat ze te gehecht zijn aan het oude systeem. David ziet het liefst alles in 1 systeem, en in 1 interface, bij voorkeur ziet hij alles in Microsoft Outlook. Voor het workflow-systeem zou dat betekenen dat het communicatiemiddel e-mail een belangrijke rol speelt. Verder moet e-mail de mogelijkheid hebben om weer invoer naar het workflow-proces te leveren. Hetzij door een link te klikken in de mail, hetzij door het mailtje te beantwoorden.
- Voor elk onderdeel uit een proces moet de status op te vragen zijn (bijv. aanvaard, bevestigd, rapport ingediend, enz.)
- Projecten bestaan uit een hoop onderdelen. De onderdelen van een project moeten afgevinkt kunnen worden.
- Als je in een overzicht ziet dat een brief af is, wil je hierop kunnen klikken waarna je de brief weer ziet. In het te bouwen workflow-systeem hangt het er van af waar de brief is opgeslagen. Als dat in het CMS is, kan de brief worden getoond. In andere gevallen kan alleen tekstueel een indicatie worden gegeven: de brief is te vinden op schijf x in map y.
- David ziet e-mail als een goede reminder, want die lees je altijd.
- Per project wil je documenten opslaan. Momenteel worden documenten op een centrale schijf 'in the cloud' opgeslagen. Via Windows Verkenner worden de documenten benaderd. De werknemers bij EBD zijn blij met deze huidige werking, en de bestandenmodule in het CMS gaat de centrale schijf niet vervangen. Verschillende mensen werken namelijk zo nu en dan

in de documenten, en altijd moet de laatste versie beschikbaar blijven. Voor het geval van onze testcase zal het workflow-proces puur activiteiten gaan begeleiden. Bestanden worden niet fysiek via het CMS aangeboden, maar in de communicatie zelf zal gerefereerd moeten worden waar de bestanden te vinden zijn. Met Windows Verkenner moet grip gehouden kunnen worden op de documenten. Ook in verband met de dagelijkse back-up die worden gemaakt.

- Er moet een duidelijk overzicht zijn van wat er gaande is, zodat je weet waar aan wordt gewerkt. Het workflow-systeem moet daarom rollen of gebruikers aan de taken kunnen toekennen.
- Meldingen en herinneringen via de mail moet te doseren zijn. (1x per dag, 1x per uur enz.) Alleen op die manier kan een herinneringsmail effectief blijven. Want het is een risico als een te grote lading mails ervoor zorgt dat e-mails ongelezen blijven.
- Bij een herinnering of het opvragen van de status zijn een aantal feiten over projecten belangrijk om te tonen:
 - De voortgang van het proces. Wat is wel en niet gedaan?
 - Wat levert het project op (in euro)?
 - Hoeveel tijd is er nog tot het verstrijken van de deadline?
- Indien de deadline is verstreken blijft een proces in het systeem bestaan totdat het proces daadwerkelijk is afgerond. Het blijft wel escalatie e-mails sturen.
- Projecten worden niet zo zeer op prioriteit ingeschaald. Het belangrijkste is hoeveel het project oplevert.

Volledige workflow-stappenplan voor het organiseren van een project

Een fonds subsidieert verschillende projecten en elke fonds heeft de volgende kenmerken:

- Contactpersoon
- Adres
- E-mailadres

Elk project wordt maar door een of meer fondsen gesubsidieerd en bevat de volgende stappen:

1. Project aanmaken naam, nummer/id
2. Plan + begroting uitwerken (deadline + verantwoording(projectleider))
3. Plan + begroting intern opsturen (deadline + verantwoording(projectleider)) en wachten op goedkeuring
4. Krijgt goedkeuring binnen (of plan + begroting verbeteren en opnieuw opsturen)
5. Plan + begroting opsturen naar de verschillende fondsen die dit project financieren.
6. Definitieve begroting + dekkingsplan maken (deadline + verantwoording(projectleider))
7. Definitieve begroting + dekkingsplan intern opsturen en wachten op goedkeuring
8. Krijgt goedkeuring binnen (of begroting + dekkingsplan verbeteren en opnieuw opsturen)
9. Definitieve begroting + dekkingsplan opsturen naar de verschillende fondsen
10. Contract/doorgangsbevestiging maken en opsturen (deadline + verantwoording (projectleider) + bedrag)

11. Contract/doorgangsbevestiging ondertekend
12. Geld ontvangen/afrekening
13. Project uitvoeren (datum)
14. Bestedingsverklaring + verslag maken (deadline + verantwoording(projectleider))
15. Bestedingsverklaring + verslag opsturen (deadline + verantwoording(projectleider))
16. Project afsluiten en archiveren

Evenementenbureau Delft heeft een zeer duidelijke en concrete workflow aangeleverd. Ook hebben zij aangegeven dat een systeem dat helpt bij het organiseren zeer gewenst is. Daarom hebben wij deze case geselecteerd om verder uit te werken, en te gebruiken in de komende fases en presentaties. Gedurende de implementatieperiode van het project zal het proces van EBD gemodelleerd worden, ook zal er een nieuwe afspraak met David worden gemaakt om een gebruikerstest uit te voeren. EBD krijgt bij afronding van het project de module aangeboden as-is.

3.3 Interview met Kees Rodenburg, Senior Medewerker B&BD, VOC Sassenheim

Interview heeft plaatsgevonden op maandag 17 oktober 2011 van 15:00 uur tot 16:45 uur, op het hoofdkantoor van VOC in Sassenheim. Bij het interview was ook leidinggevende Manager Binnen- en Buitendienst Ed Rademaker aanwezig.

Vanwege informatie over bedrijfsvoering is een deel van het interview uit dit document gelaten.

Vanwege informatie over bedrijfsvoering is dit interview uit dit document gelaten.

4 Functionele eisen

4.1 MoSCoW

Voor deze opdracht wordt gebruik gemaakt van de MoSCoW-methode. Deze methode stelt de prioriteiten vast van de verschillende onderdelen waaruit de opdracht bestaat. Het woord MoSCoW staat voor:

- **Must have:** dit zijn eisen die onmisbaar zijn. Zonder deze eisen is het eindresultaat niet bruikbaar, en is het project mislukt.
- **Should have:** dit zijn eisen die zeer gewenst zijn. Zonder deze eisen werkt het eindproduct nog wel, maar heeft het tekortkomingen in gebruik.
- **Could have:** dit zijn wensen die pas prioriteit krijgen als er nog genoeg tijd over is, en de must haves en should haves zijn geïmplementeerd.
- **Won't have:** dit zijn ideeën voor vervolgprojecten om in het achterhoofd te houden. Deze punten zullen niet in het eindresultaat aan bod komen.

De letters o hebben geen betekenis, maar maakt MSCW uitspreekbaar. In de MoSCoW hieronder zijn alle functionele eisen opgesomd die volgen uit ons eerdere onderzoek in de oriëntatie en analyse-fase. Verder hebben wij ons verdiept in een bestaand softwarepakket van Microsoft genaamd Dynamics CRM 4.0. Dit softwarepakket biedt ondersteuning voor workflows, en gaf ons extra ideeën voor de mogelijkheden van de workflow-module. [?]

4.1.1 Must have

- In de workflow-module moeten verschillende processen gemodelleerd kunnen worden.
- Processen moeten toegevoegd, gewijzigd en verwijderd kunnen worden.
- Workflow-rollen moeten toegevoegd, gewijzigd, verwijderd kunnen worden.
- Gebruikers moeten toegevoegd, gewijzigd, verwijderd kunnen worden.
- Gebruikers moeten een of meerdere workflow-rollen toegekend kunnen krijgen.
- Activiteiten moeten toegevoegd, gewijzigd en verwijderd kunnen worden.
- Er moet een *interface* voor triggermethodes aanwezig zijn. Een trigger reageert op een signaal van buiten het workflow-systeem.
- Er moet een *interface* voor communicatiemethodes aanwezig zijn.
- Sequentiële routing in het workflow-proces: activiteiten kunnen na elkaar worden uitgevoerd.
- Per activiteit kan één trigger, verwerkingsmethode, en communicatiemethode aangeroepen worden.
- De trigger-, verwerking- en communicatiemethodes uit de Must-categorie bij sectie 4.1.5 worden geïmplementeerd.

4.1.2 Should have

- Activiteiten kunnen los aan/uit gezet worden (in workflow-termen: publiceren / niet publiceren).
- Bij elke activiteit uit een workflow-proces wordt statusinformatie bijgehouden. Dit omvat de status (actief, inactief, succes, gepauzeerd, geannuleerd, etc.) en status-reden (wacht op details, niet meer geïnteresseerd, etc.)
- Voortgang van processen en activiteiten monitoren. Deze toont statusberichten, gebruikers en rollen bij alle processen en activiteiten.
- Een activiteit kan meerdere triggers, verwerkingsmethodes, en communicatiemethodes aanroepen.
- De trigger-, verwerking- en communicatiemethodes uit de Should-categorie bij sectie 4.1.5 worden geïmplementeerd.

4.1.3 Could have

- Activiteiten of processen herstarten / resetten.
- Er moet een *interface* voor de verschillende condities waarmee routingmechanismes kunnen vergelijken aanwezig zijn.
- Mogelijkheid om processen en activiteiten te controleren d.m.v. logboek (audit trail). Deze toont alle statusberichten van een activiteit met tijd en datum.
- Wachtconditie: een time-out toevoegen, uitvoer van workflow-proces wordt pas hervat nadat de tijd is verlopen.
- Parallele routing in het workflow-proces: meerdere activiteiten kunnen tegelijkertijd worden uitgevoerd.
- Iteratieve routing in het workflow-proces: activiteiten of een verzameling van activiteiten kunnen meerdere malen worden uitgevoerd.
- De trigger-, verwerking- en communicatiemethodes uit de Would-categorie bij sectie 4.1.5 worden geïmplementeerd.

4.1.4 Won't have

- Een werklist waarmee gebruikers van een rol zelf taken kunnen uitkiezen.
- Taken verdelen over verschillende gebruikers van dezelfde rol.
- Eenvoudige berekeningen moeten in een e-mail opgenomen kunnen worden.
- De scope van een workflow-activiteit kan worden opgegeven. Bij verwerking worden alleen items die aan de eisen voldoen verwerkt. (bijv. items die toegekend zijn aan een bepaalde gebruiker of rol).
- In lijstweergave van het CMS een extra knop plaatsen waarbij een workflow kan worden uitgevoerd op een selectie van items.
- Ondersteuning van de XPDL-standaard voor het importeren en exporteren van workflow-definities.
- Het modelleren van workflow-processen nog intuïtiever maken zodat de module ook door klanten gebruikt kan worden i.p.v. alleen Content Power

- Oneindige-lus detectie. Deze detectie is mogelijk op basis van de diepte, en het instellen van een tijdslimiet voor de uitvoer van een proces.
- Conditievergelijking: waarden van een bepaald item, tellers of uitvoertijden vergelijken.
- Een *interface* aanwezig om workflow-patterns en routingmechanismes voor afhankelijkheden toe te voegen (OR, XOR, branching, etc.).
- Conditionele routing in het workflow-proces: een activiteit wordt alleen uitgevoerd als aan een bepaalde conditie is voldaan. De condities worden gecontroleerd, en afhankelijk van die condities worden acties ondernomen.
- Dynamische waarden van items moeten in een e-mail opgenomen kunnen worden.
- De trigger-, verwerking- en communicatiemethodes uit de Won't-categorie bij sectie 4.1.5 worden geïmplementeerd.

4.1.5 Triggers, Verwerking en Communicatie

Methode:	Naam:	Must	Should	Could	Won't
Trigger	een proces of activiteit wordt handmatig gestart	X			
Verwerking	trigger een andere workflow-activiteit uit het huidige workflow-proces	X			
Communicatie	stuur een e-mail naar x	X			
Trigger	er is een nieuw item aangemaakt van een bepaalde entiteit		X		
Trigger	de status van een bepaald item is gewijzigd		X		
Verwerking	wijzig de status van een item		X		
Trigger	elke minuut wordt via een cronjob een getimed trigger uitgevoerd			X	
Trigger	een specifieke pagina is aangeroepen			X	
Trigger	een datum en/of tijd (deadline) is verstreken			X	
Trigger	de rol / gebruiker die is toegekend aan een item is gewijzigd naar een andere rol of gebruiker			X	
Trigger	een of meerdere waarden van een item zijn gewijzigd			X	
Trigger	een item is verwijderd			X	
Verwerking	maak een nieuw item aan			X	
Verwerking	wijzig een of meerdere waarden van een item			X	
Verwerking	wijs een item toe aan een andere gebruiker of rol			X	
Verwerking	verwijder een item			X	
Verwerking	termineer de uitvoer van het workflow-proces			X	
Trigger	e-mail ontvangen op een e-mailadres x, met invoer				X
Trigger	een teller bereikt een bepaalde waarde				X
Verwerking	plaats een nieuwe afspraak in een agenda				X
Verwerking	voer een extern proces uit, bijv. een shell-applicatie				X
Verwerking	start asynchroon een ander workflow-proces (huidige workflow-proces gaat door)				X
Verwerking	start synchroon een child-workflow-proces (huidige workflow-proces gaat pas door nadat child termineert)				X
Communicatie	plaats een bericht op Twitter / andere sociale media				X
Communicatie	plaats een of meerdere afspraken in een agenda				X
Communicatie	stuur een SMS				X

5 Niet-functionele eisen

5.1 Gebruikersinterface

De gebruikersinterface zal primair door Content Power worden gebruikt. In de opdrachtbeschrijving is aangegeven dat het gebruik van de module enig technisch inzicht mag vereisen. Daarom zijn er geen zware eisen gesteld aan de gebruikerservaring. De gebruikersinterface hoeft alleen te werken in de recente versies van webbrowser Firefox. Voor zover mogelijk wordt functionaliteit in de gebruikersinterface zo veel mogelijk gepresenteerd op de manier zoals bestaande functionaliteit in het CMS wordt gepresenteerd.

5.2 Documentatie

Het systeem wordt geleverd met een duidelijke documentatie. Zowel een hardcopy gebruikshandleiding voor Content Power, als helpfuncties in de module zelf. Deze helpfunctie geeft uitleg over de functie die op dat moment actief is. De code dient van degelijke documentatie voorzien te zijn. Deze documentatie moet voldoen aan de PhpDoc standaard.

5.3 Software-eisen

De module moet werken op de bestaande serverinfrastructuur van Content Power. Code is geschreven in PHP en maakt gebruik van MySQL databases. Aangezien het bestaande CMS closed-source is, en het CMS strikt en alleen op eigen servers van Content Power wordt geïnstalleerd (besturingssysteem CentOS 5 Linux), is het geen noodzakelijke eis om de code platformonafhankelijk te schrijven. De workflow-module moet binnen het bestaande CMS-framework gebouwd worden. De workflow-module moet voor triviale taken gebruik maken van bestaande buildingblocks.

5.4 Prestaties

De module moet vlot werken, dat wil zeggen dat de module onder normale omstandigheden binnen 0 en 5 seconden respons geeft. Een workflow model moet snel en optimaal gemodelleerd kunnen worden in de module. Het geheugen- en processorgebruik van de workflow-module en alle ingebouwde triggers, verwerking- en communicatiemethodes moet acceptabel zijn, dat wil zeggen vergelijkbaar met andere bestaande modules in het CMS.

5.5 Foutafhandeling en extreme omstandigheden

Codefouten worden via bestaande methodes voor foutmeldingen en debuggen afgehandeld. De integriteit van de code moet hoog zijn, en mag geen bugs bevatten. Omdat dit in de praktijk niet haalbaar is, omdat er altijd bugs in een systeem overblijven, is de eis dat de gebruiker in geen enkel geval de foutmeldingen te zien krijgt. In dat geval worden de fouten wel in een logboek geplaatst zodat Content Power het probleem kan oplossen. Het maken en terugzetten van een back-up via de workflow-module is niet nodig, want in het CMS bestaan al functies om ervoor te zorgen dat workflows geïmporteerd en geëxporteerd kunnen worden. Ook wordt er al automatisch elke nacht een backup gemaakt. Daarom hoeft er nooit data verloren te gaan.

5.6 Interactie tussen systemen

Overbodig om te melden, maar de systemen waarop de workflow-module draait, dienen aangesloten te zijn op het wereldwijde web: Internet.

5.7 Kwaliteitseisen

De module moet robuust werken en stabiel zijn. Overal waar mogelijk iets fout kan gaan moet deze fout worden afgevangen. In gevallen van onzekere data, bijvoorbeeld bij koppelingen met

externe systemen, mag het systeem niet crashen als er geen verbinding gemaakt kan worden. De module dient gebruik te maken van het reeds in het CMS aanwezige systeem om systeemfouten te melden. Fouten worden gemeld met behulp van de methode `vError()`.

5.8 Systeemaanpassingen

De module moet zo geschreven zijn dat het in de toekomst makkelijk uitgebreid kan worden, met bijvoorbeeld nieuwe triggers, communicatiemethodes, en kernfunctionaliteiten. Ook moet de module goed blijven schalen. Als er bij praktisch gebruik heel veel activiteiten, taken en instanties worden aangemaakt, moet de module snel en stabiel blijven werken.

5.9 Beveiligingseisen

Bij alle functionaliteiten die de workflow-module heeft, moet rekening gehouden worden met rechten die volgen uit het rechtenmodel dat al in het CMS is geïmplementeerd. De module mag geen datalekken of beveiligingsproblemen veroorzaken in de huidige werking van het CMS.

5.10 Beheer

Content Power moet het systeem centraal kunnen beheren, en de module bij verschillende klanten aan- en uit kunnen zetten. Wanneer de module is gedeactiveerd, wordt geen enkel workflow-proces uitgevoerd.

6 Constraints

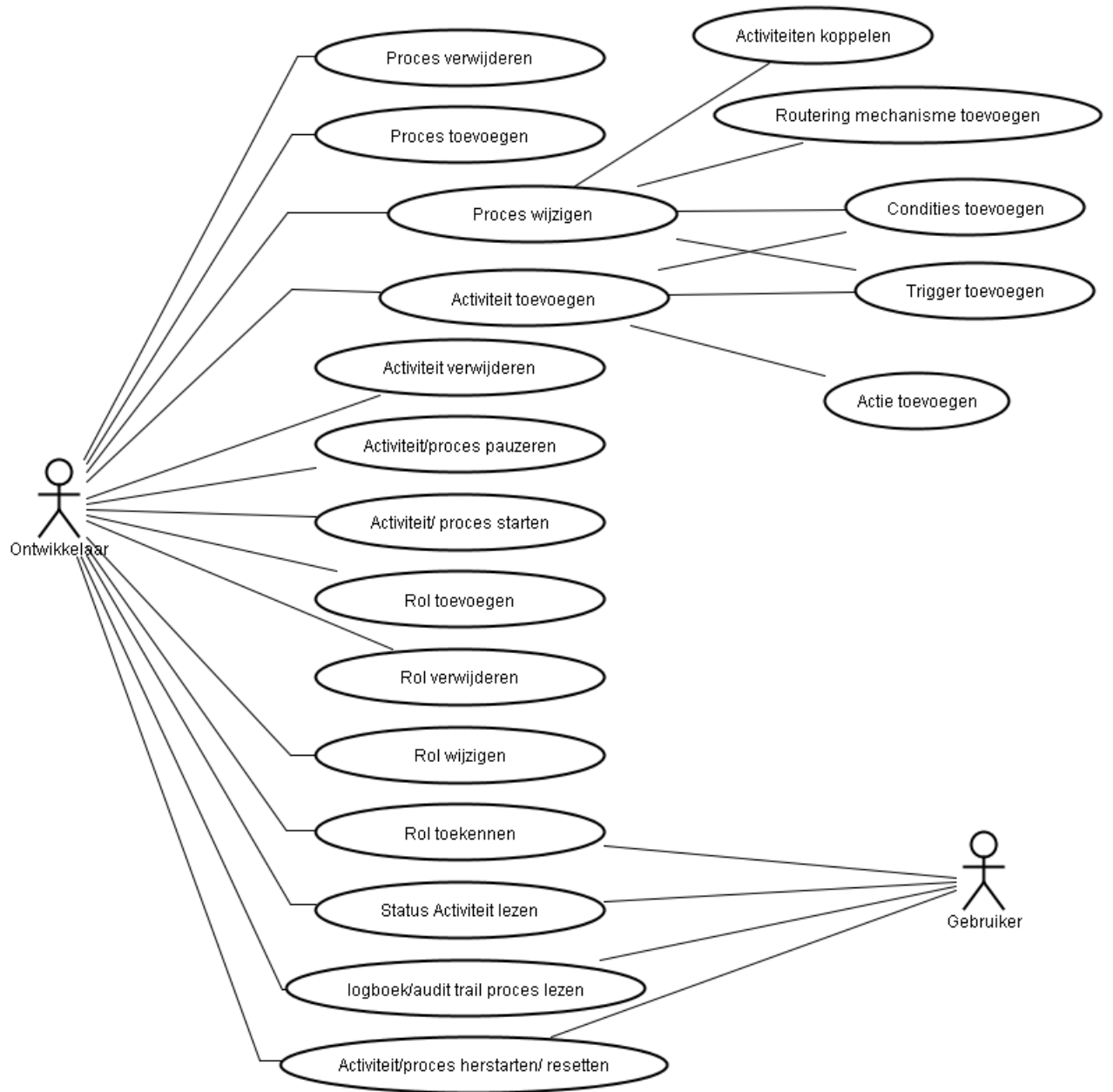
De module moet in objectgeoriënteerde PHP worden geschreven. De gegevens worden in dezelfde database opgeslagen als waar het CMS ook zijn data plaatst. Communicatie met de database dient plaats te vinden via de bestaande `CmsDB` klasse uit de `CBB`-package. Er zijn in totaal 10 weken beschikbaar om alle fases van het project te doorlopen. Ook het testen valt binnen deze periode. Het testen zal in de periode daarna door Content Power worden voortgezet.

7 Systeemmodellen

7.1 Usecase modellen

Hieronder zijn een aantal simpele usecases opgesteld waarmee eenvoudige workflow-processen te modelleren zijn. Daarnaast is een usecase diagram gemaakt met alle functionaliteiten die volgen uit de functionele eisen, en wie de rechten heeft om de functionaliteiten te gebruiken.

7.1.1 Usecase diagram



Figuur 2: Basisfuncties die de gebruiker en de ontwikkelaar moeten kunnen gebruiken

7.1.2 Usecases

1. Webwinkel

Actoren: Klant, winkelier.

Doel: Bestelling afhandelen.

Samenvatting: Door een bestelling te plaatsen wordt een heel proces in werking gesteld waarbij de winkelier de bestelling moet controleren

Trigger: een klant plaatst een bestelling.

1. Webwinkel neemt de bestelling in behandeling
 - (a) Klant krijgt een bevestiging dat de bestelling in behandeling is.
 - (b) De bestelling wordt toegevoegd aan een lijst met te controleren bestellingen
2. Winkelier controleert en accepteert de bestelling
3. Bestelling wordt verstuurd
 - (a) Klant krijgt bericht dat de bestelling is verstuurd
4. Na een week wordt een evaluatieformulier naar de klant verstuurd

2. Herinnering

Actoren: Gebruiker.

Doel: Gebruiker mag een afspraak niet vergeten.

Samenvatting: De gebruiker moet automatisch herinnerd worden aan een afspraak door een dag vooraf een herinneringsmail te krijgen

Trigger: Gebruiker maakt een herinnering aan.

1. Herinnering wordt aan de database toegevoegd.
2. De afspraak is verplaatst dus wijzigt de gebruiker de datum.
3. Herinnering wordt een dag voor de afspraak verstuurd
4. Herinnering wordt een uur voor de afspraak nog een keer verstuurd

3. Declaratie

Actoren: Werknemer en manager.

Doel: Werknemer kan declaraties invullen waarna deze automatisch worden verwerkt

Samenvatting: De werknemer dient een declaratie in van 3000 euro. Deze zal automatisch worden verwerkt. De declaratie wordt eerst gecontroleerd door de manager.

Trigger: Werknemer dient een declaratie in.

1. Declaratie wordt opgeslagen
 - (a) Omdat de declaratie hoger is dan 2500 euro wordt er een mail naar de manager gestuurd
2. Manager keurt declaratie goed
3. Declaratie wordt doorgestuurd naar een medewerker van de afdeling Finance
4. De medewerker voert de betaling uit, en de declaratie wordt gemarkeerd als afgehandeld.

4. NEVI: nieuw cijfer in de cijferadministratie

Actoren: Cursist en examencommissie.

Doel: Cursist komt zo snel mogelijk te weten als er nieuwe resultaten zijn

Samenvatting: De examencommissie heeft een examen tweevoudig nagekeken. De resultaten worden verwerkt in KAS. Zodra de resultaten zijn ingevoerd en bevestigd, zijn de resultaten zichtbaar voor de cursist.

Trigger: De resultaten zijn ingevoerd, en worden definitief bevestigd.

1. Het resultaat wordt bewaard in de cijferadministratie.
2. De cursist krijgt een e-mail met daarin informatie over het behaalde resultaat.
3. Als de cursist inlogt in zijn eigen omgeving ziet hij daar de resultaten in het cijferoverzicht staan.

5. Evenementenbureau Delft: nieuw project

Actoren: Projectleider, Fondsen

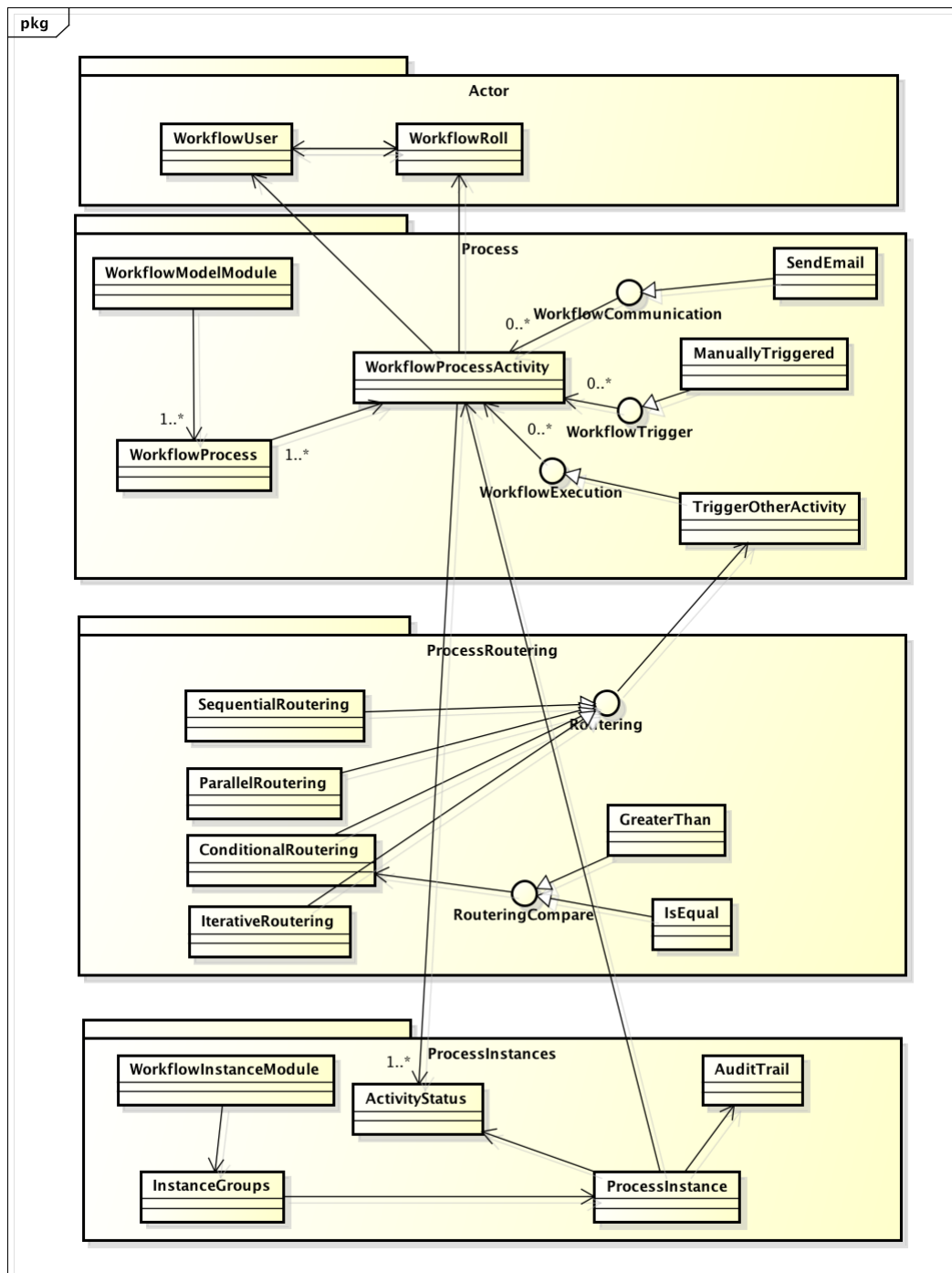
Doel: Via fondsenwerving komt geld beschikbaar om evenementen mogelijk te maken. Elk evenement is een project.

Samenvatting: Elk project wordt maar door 1 (of meer) fondsen gesubsidieerd en bevat de volgende stappen. Een fonds subsidieert verschillende projecten en elke fonds heeft een aantal kenmerken: contactpersoon, adres, maximaal budget voor alle projecten.

Trigger: Project aanmaken in module in het CMS onder de module Projecten. naam, nummer/id.

1. Project aanmaken naam, nummer/id
2. Plan + begroting uitwerken (deadline + verantwoording(projectleider))
3. Plan + begroting intern opsturen (deadline + verantwoording(projectleider)) en wachten op goedkeuring
4. Krijgt goedkeuring binnen (of plan + begroting verbeteren en opnieuw opsturen)
5. Plan + begroting opsturen naar de verschillende fondsen die dit project financieren.
6. Definitieve begroting + dekkingsplan maken (deadline + verantwoording(projectleider))
7. Definitieve begroting + dekkingsplan intern opsturen en wachten op goedkeuring
8. Krijgt goedkeuring binnen (of begroting + dekkingsplan verbeteren en opnieuw opsturen)
9. Definitieve begroting + dekkingsplan opsturen naar de verschillende fondsen
10. Contract/doorgangsbevestiging maken en opsturen (deadline + verantwoording (projectleider) + bedrag)
11. Contract/doorgangsbevestiging ondertekend
12. Geld ontvangen/afrekening
13. Project uitvoeren (datum)
14. Bestedingsverklaring + verslag maken (deadline + verantwoording(projectleider))
15. Bestedingsverklaring + verslag opsturen (deadline + verantwoording(projectleider))
16. Project afsluiten en archiveren

7.2 Klassendiagram



powered by astah

Klassendiagram vanuit het perspectief van de requirements

E Technical Design and Specification Document

Technical Design and Specification Document -
Bachelorproject IN3405
Workflow-management in een Content Managementsysteem

Rob Ruigrok
R.J.Ruigrok@student.tudelft.nl
1371746

Jan Pieter Mars
J.P.Mars@student.tudelft.nl
1381075

7 december 2011



ContentPower
informatiebeheersystemen

Voorwoord

Dit document geeft een technische beschrijving van de functionaliteiten uit het Requirements and Analysis Document. Met behulp van UML, diagrammen en tabellen worden de specificaties van het ontwerp uitgewerkt. In het ontwerp zijn alle Must, Should en Could punten uit het MoSCoW document opgenomen. Tijdens de implementatiefase wordt aan de hand van deze structuur de code opgebouwd. Functionaliteiten waar geen tijd voor is, zullen lege klassen blijven.

Samenvatting

Het ontwerp bestaat uit twee delen, die van elkaar afhankelijk zijn. Deze twee delen zijn:

- Database-ontwerp: in dit ontwerp wordt de datastructuur voor de database uitgewerkt. Het beschrijft de structuur van tabellen, velden, sleutels en relaties. Het ontwerp wordt vervolgens via de module CpMyAdmin in het CMS in de database geplaatst.
- Package- en klassendiagram: deze diagrammen beschrijven de structuur van de code. Een opzet van dit diagram is al te vinden in het RAD, maar in dit document zal een uitgebreider diagram worden opgesteld.
- Klassenspecificaties: Elke klasse in het diagram wordt uitgebreid beschreven door methodes en velden te specificeren.

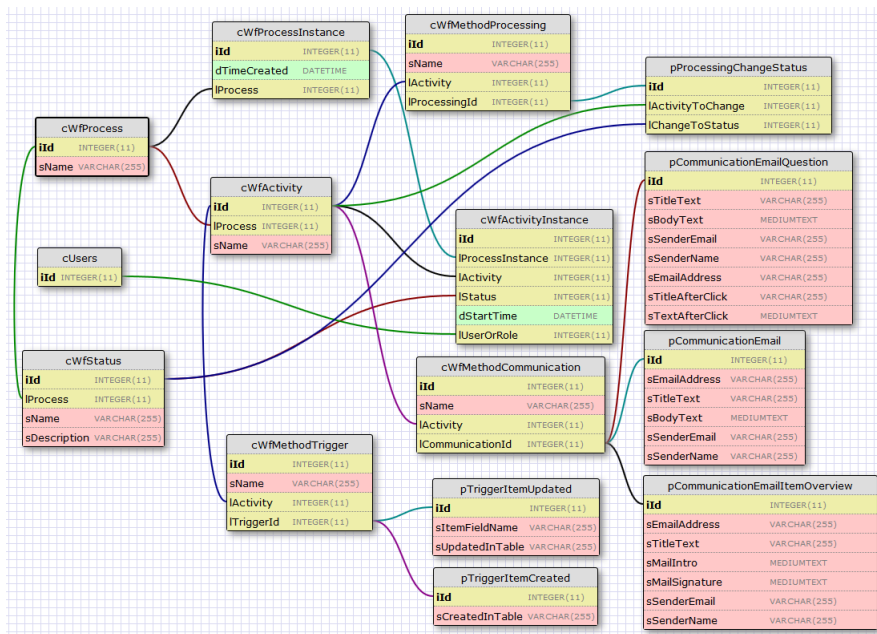
Inhoudsopgave

1	Inleiding	92
2	Database ontwerp	92
2.1	cWfProcess	92
2.2	cWfActivity	92
2.3	cWfProcessInstance	92
2.4	cWfActivityInstance	93
2.5	cWfStatus	93
2.6	cWfMethodTrigger	93
2.7	pTriggerItemUpdated	93
2.8	pTriggerItemCreated	93
2.9	cWfMethodProcessing	93
2.10	pProcessingChangeStatus	93
2.11	cWfMethodCommunication	93
2.12	pCommunicationEmailQuestion	93
2.13	pCommunicationEmailItemOverview	94
2.14	cUsers	94
2.15	sMetaItem	94
2.16	sLinkTable	94
2.17	mFonds	94
2.18	mProject	94
3	Package- en klassendiagram	95
4	Gebruikersinterface	97
4.1	Hoofdscherm	97
4.2	Processen	97
4.3	Activiteiten	98
4.4	Triggers	98
4.5	Verwerking	99
4.6	Communicatie	99
4.7	Status	100
4.8	Projecten	100
4.8.1	Project bewerken tab 1	101
4.8.2	Project bewerken tab 2	101
4.8.3	Project bewerken tab 3	102
5	Klassenspecificaties	102

1 Inleiding

In de volgende secties zullen technische implementatiedetails worden beschreven. Een belangrijk gedeelte is het datamodel in de database. Dit is een zeer belangrijk deel waarop de code is gebaseerd. Deze stap zal vooral veel werk in het CMS vereisen, via de module CpMyAdmin wordt het datamodel verwerkt.

2 Database ontwerp



Figuur 1: Datamodel voor de structuur van de tabellen in de database

2.1 cWfProcess

Dit is een CMS systeemtabel. Deze bevat een lijst van gemodelleerde processen. Elk proces heeft een leesbare naam (sName).

2.2 cWfActivity

Dit is een CMS systeemtabel. Deze tabel hangt onder cWfProcess, en bevat voor elk gemodelleerde proces alle onderliggende activiteiten. Onder een activiteit hangen trigger-, verwerking- en communicatiemethodes. Deze drie methodes zijn technisch identiek aan elkaar qua interface. Toch is gekozen voor een opsplitsing, omdat dat het overzicht in het CMS ten goede komt. Bij activiteiten worden dan drie afzonderlijke methodes als onderliggende items getoond, in plaats van 1 onderliggend item waarin alle methodes door elkaar staan verwerkt.

2.3 cWfProcessInstance

Dit is een CMS systeemtabel. Het bevat een verwijzing naar een procesmodel uit cWfProcess. Van een enkel procesmodel kunnen meerdere instanties worden aangemaakt. Van elke instantie wordt statistiek-informatie bijgehouden, die wordt gebruikt om het proces te monitoren.

2.4 cWfActivityInstance

Dit is een CMS systeemtabel. Deze tabel hangt onder cWfProcessInstance, en bevat verwijzingen naar procesactiviteiten uit het activiteitenmodel cWfProcessActivity. Deze tabel bevat basisinformatie over de gelinkte activiteit.

2.5 cWfStatus

Dit is een CMS systeemtabel. Het bevat alle mogelijke statussen voor elk proces en activiteit. In het model kan gekozen worden bij welke activiteiten welke statussen mogelijk zijn. In de tabel cWfActivityInstance wordt voor alle activiteitinstanties de huidige status bewaard.

2.6 cWfMethodTrigger

Dit is een CMS systeemtabel. Deze tabel hangt onder cWfProcessActivity, en bevat voor elke activiteit de bijbehorende triggermethodes. In deze tabel wordt verwezen naar een item uit een van de p-tabellen. Dit kunnen verschillende types triggermethodes zijn die in een van de p-tabellen zijn opgeslagen.

2.7 pTriggerItemUpdated

Dit is een implementatietabel van een triggermethode die reageert wanneer de in de database een veld in een bepaalde tabel wordt gewijzigd.

2.8 pTriggerItemCreated

Dit is een implementatietabel van een triggermethode die reageert als er een object is aangemaakt in een bepaalde tabel.

2.9 cWfMethodProcessing

Dit is een CMS systeemtabel. Deze tabel hangt onder cWfActivity, en bevat voor elke activiteit de bijbehorende verwerkingsmethodes. In deze tabel wordt verwezen naar een item uit een van de p-tabellen. Dit kunnen verschillende types verwerkingsmethodes zijn die in een van de p-tabellen zijn opgeslagen.

2.10 pProcessingChangeStatus

Dit is een implementatietabel van een verwerkingsmethode die in staat is om de status van een activiteitinstantie te veranderen.

2.11 cWfMethodCommunication

Dit is een CMS systeemtabel. Deze tabel hangt onder cWfActivity, en bevat voor elke activiteit de bijbehorende communicatiemethodes. In deze tabel wordt verwezen naar een item uit een van de p-tabellen. Dit kunnen verschillende types communicatiemethodes zijn die in een van de p-tabellen zijn opgeslagen.

2.12 pCommunicationEmailQuestion

Dit is een implementatietabel van een communicatiemethode die een e-mail kan versturen met daarin tekst, met tevens de mogelijkheid om links toe te voegen voor ja/nee vragen voor diverse velden. Deze link wordt in de mail opgenomen, en wanneer er op geklikt wordt, wordt een waarde in de database bijgewerkt. De interface om velden toe te voegen als klikbare links is momenteel nog technisch. Door in de bodytekst een regel op te nemen met ~link|veldnaam in tabel|tekst voor

klikbare link~wordt een link in de website op die plaats opgenomen. Wanneer er een pTriggerItemUpdated bestaat die de wijziging detecteert, kan dit voortgang in het workflow-proces tot gevolg hebben. Soms komt het voor dat er een tekst van een veld in de mail moet worden opgenomen. Dat gebeurt via de regel ~text|veldnaam in tabel~.

2.13 pCommunicationEmailItemOverview

Dit is een implementatietabel van een communicatiemethode die een e-mail kan versturen met daarin tekst, en tevens de mogelijkheid om een overzicht van de waarden van het gekoppelde item mee te sturen in de mail.

2.14 cUsers

Dit is een bestaande tabel in het CMS, het bevat gebruikers van het systeem. In het workflow-systeem wordt van de bestaande gebruikers/rollen-structuur gebruik gemaakt.

2.15 sMetaItem

Dit is een CMS systeemtabel. De code van het CMS gebruikt deze tabel om de tabellen op te zoeken waar een specifiek id te vinden is. Alle cWfMethod*-tabellen maken gebruik van deze tabel om te weten te komen in welke p-tabel zij een id kunnen vinden.

2.16 sLinkTable

Dit is een CMS systeemtabel die multilinks mogelijk maakt. In het CMS wordt deze tabel gebruikt voor alle velden die gedefinieerd zijn met het type "popup met opties".

2.17 mFonds

Dit is een maatwerktabel voor de case van Evenementenbureau Delft (EBD). Het bevat fondsen die geld beschikbaar stellen voor het organiseren van projecten.

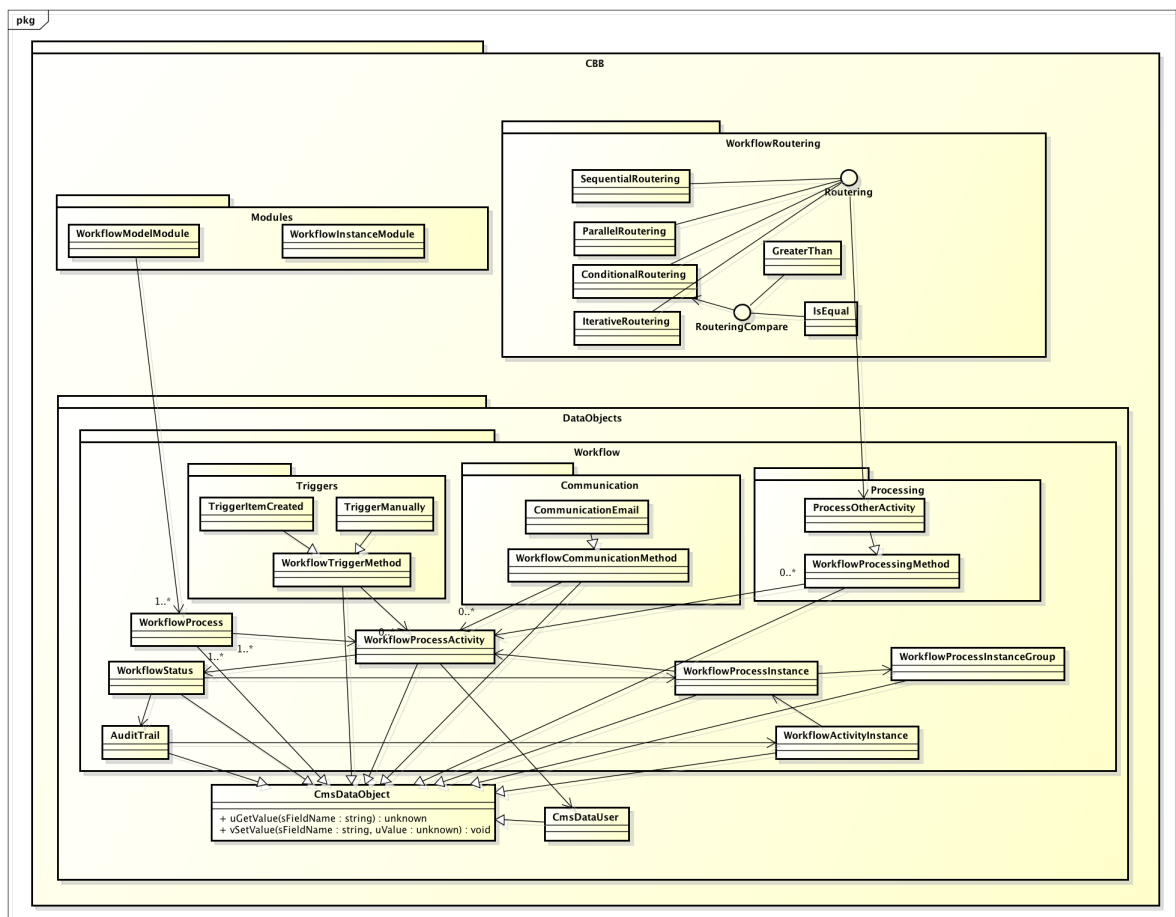
2.18 mProject

Dit is een maatwerktabel voor de case van Evenementenbureau Delft (EBD). Het bevat informatie over projecten die EBD organiseert, zoals deadlines, en welke taken zijn uitgevoerd. Het organiseren van projecten is voor EBD een grote workflow. De projecten-tabel is gekoppeld aan het workflow-managementsysteem. Procesinstanties worden aangemaakt zodra er een nieuw project in de module Projecten wordt toegevoegd.

3 Package- en klassendiagram

In de scripttaal PHP was oorspronkelijk geen ondersteuning voor packages aanwezig. Dit had tot gevolg dat klasse-namen in een codebase globaal uniek moesten zijn. In recente versies van PHP is ondersteuning voor zogenaamde namespaces opgenomen, maar omdat de bestaande code van het CMS hier niet voor gebouwd is, is bewust gekozen om geen gebruik te maken van namespaces. Klasse-namen worden zo geformuleerd dat zij uniek zijn binnen de codebase, en aan de naam te herkennen is tot welke type zij behoren. Hieronder volgt een voorbeeld om dit te verduidelijken: de klasse `CmsDBContentProviderObjectListWorkflowInstance` is de databaseklasse (`CmsDB`) die als controller dient (`ContentProvider`), om objecten uit de database op te halen (`ObjectList`), bestaande uit workflow-instanties (`WorkflowInstance`).

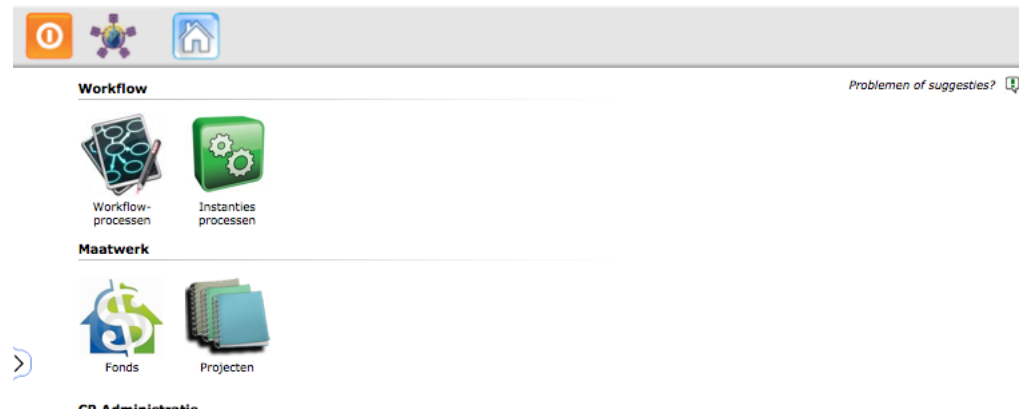
Onderstaande klassendiagrammen tonen wel packages, maar dat is in werkelijkheid de map-structuur waarin de code is gestructureerd.



Figuur 2: Klassendiagram met klassen geordend in folders

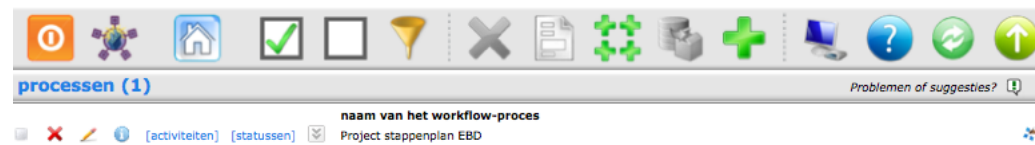
4 Gebruikersinterface

4.1 Hoofdscherm



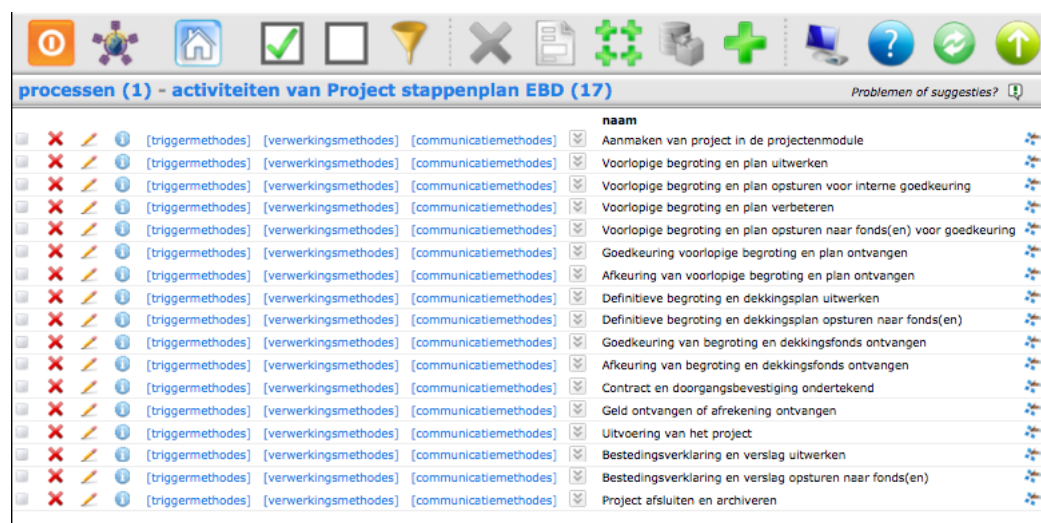
Dit is het hoofdscherm van het CMS. In dit hoofdscherm zijn twee modules te zien die het volledige workflow management mogelijk maken. De eerste is workflow-processen: hierin kunnen modellen van processen worden opgesteld. In de tweede worden de instanties van workflow-modellen bijgehouden. Deze instanties zijn gekoppeld aan items uit andere modules, zoals de paginamodule, of in ons testgeval: de modules fondsen en projecten. Voor een uitgebreide beschrijving over de case, zie het RAD.

4.2 Processen



In de module Workflow-processen kunnen workflow-processen worden toegevoegd, gewijzigd, verwijderd. Ook is het mogelijk om onder de workflow-processen mogelijke statussen en workflow-activiteiten te hangen.

4.3 Activiteiten



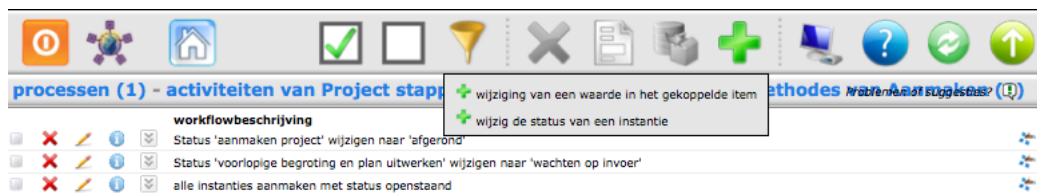
Onder workflow-processen hangen activiteiten. Een activiteit is een losse taak of combinatie van taken die zonder communicatie of extra gebruikersinvoer uitgevoerd kan worden, het vormt tezamen met alle andere activiteiten een volledig workflow-proces. Een activiteit wordt getriggerd, voert daarna de verwerkingsmethodes uit, en doet verslag via de communicatiemethodes.

4.4 Triggers



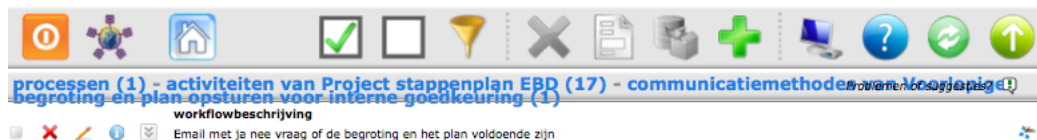
Onder een activiteit kunnen triggers worden toegevoegd. Als een van deze triggers 'afgaat', dan zullen de verwerkings- en communicatiemethodes van deze activiteit worden uitgevoerd.

4.5 Verwerking



Verwerkingsmethodes houden onder andere wijzigingen in in de statussen van andere activiteitinstanties, of het wijzigen van waarden in de database. Het is mogelijk dat deze taken weer nieuwe activiteiten triggeren, hierdoor is parallele uitvoering van processen al mogelijk. Deze functionaliteit dient wel nog grondig getest te worden.

4.6 Communicatie



Communicatiemethodes zorgen ervoor dat er uitvoer naar de gebruiker wordt gegeven. Dat kan een eenvoudige e-mail met tekst zijn, maar ook een e-mail met een vraag erin, of een e-mail met een overzicht van een item.

4.7 Status

processen (1) - statussen van Project stappenplan EBD (10) Problemen of suggesties? ⓘ										
status										
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	wachten op invoer
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	openstaand
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	afgerond
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	goedgekeurd
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	wachten op goedkeuring plan en voorlopige begroting (intern)
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	wachten op goedkeuring definitieve begroting en dekkingsplan (intern)
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	wachten op goedkeuring plan en voorlopige begroting (extern)
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	wachten op goedkeuring definitieve begroting en dekkingsplan (extern)
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	wachten op invoer (plan en/of voorlopige begroting afgekeurd)
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	wachten op invoer (dekkingsplan en/of definitieve begroting afgekeurd)

Elke activiteit in een workflow-proces kan een status bezitten. Zoals het wachten op invoer van een gebruiker, of dat een activiteit is afgerond. De verschillende trigger-, verwerkings- en communicatiemethodes maken gebruik van deze status, kunnen deze wijzigen, of reageren op wijzigingen. Voor de eindgebruiker biedt een statusoverzicht een duidelijk beeld waar het workflow-systeem zich in bevindt.

4.8 Projecten

projecten (7) Problemen of suggesties? ⓘ				
	naam van het project	inkomsten in euro's	projectleider vanuit EBD	fondsen
<input type="checkbox"/>	test	€ 0,00	Content Power	Jan Pieter
<input type="checkbox"/>	test2	€ 0,00	Cms gebruikers	-- geen --
<input type="checkbox"/>	test3	€ 0,00	Anderen	-- geen --
<input type="checkbox"/>	test4	€ 0,00	Intranet gebruikers	-- geen --
<input type="checkbox"/>	test5	€ 0,00	Cms gebruikers	-- geen --
<input type="checkbox"/>	test	€ 4,00	Cms gebruikers	-- geen --
<input type="checkbox"/>	testt	€ 0,00	Content Power test account	-- geen --

In het database-model zijn tabellen mProject en mFonds opgenomen. Dit zijn maatwerktabellen die voor de case van Evenementenbureau Delft een rol spelen. De modules Projecten en Fondsen representeren deze tabellen. Aan deze tabellen is momenteel een workflow-proces gekoppeld dat de activiteiten die ondernomen moeten worden voor het organiseren van een project chronologisch afwerken.

4.8.1 Project bewerken tab 1

The screenshot shows a web interface for editing a project. At the top, there is a toolbar with various icons. Below it is a table titled 'projecten (7)' with columns: 'naam van het project', 'inkomsten in euro's', 'projectleider vanuit EBD', and 'fondsen'. The table lists four projects: 'test', 'test2', 'test3', and 'test4'. The 'test' project is selected, and a modal window titled 'Bewerk "test"' is open. This modal contains the following fields:

- naam van het project:
- startdatum van activiteit: :
- inkomsten in euro's:
- projectleider vanuit EBD:
- fondsen: [Jan Pieter](#)

At the bottom of the modal are two buttons: 'opslaan' and 'annuleer'.

De maatwerktafel met projecten heeft een aantal standaard instellingen die voor deze case specifiek zijn. Zo is er een projectleider (dat is een gebruiker van het CMS: uit tabel cUsers), en een naam van het project voor de interne administratie van EBD.

4.8.2 Project bewerken tab 2

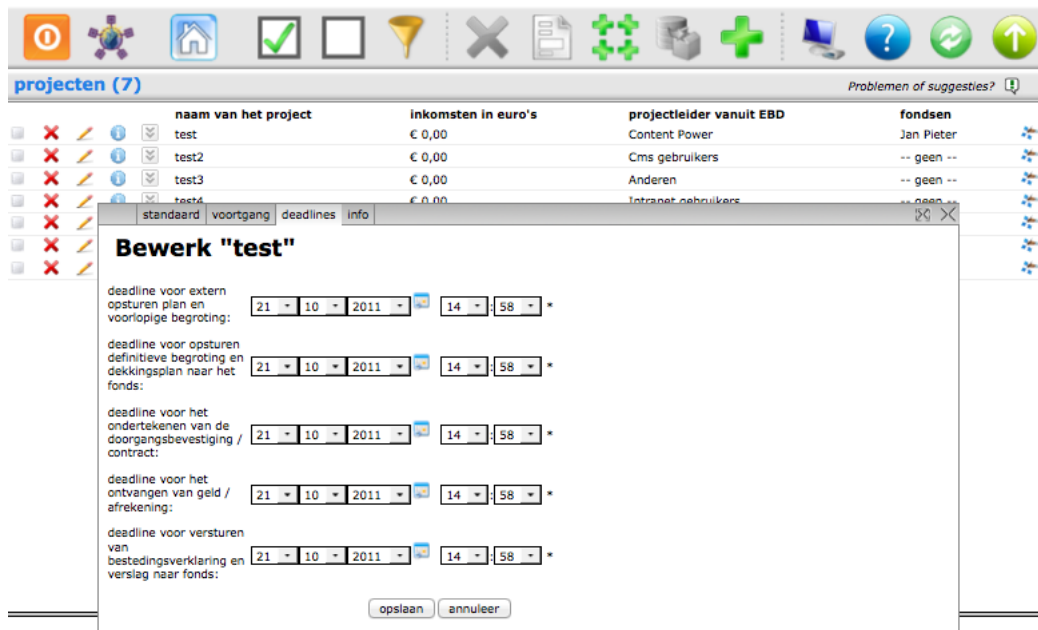
This screenshot shows the same 'Bewerk "test"' modal window, but with a different set of options. At the top, there are two buttons: 'opslaan' and 'annuleer'. Below them is a list of checkboxes for various project milestones:

- plan en voorlopige begroting uitgewerkt:
- plan en voorlopige begroting intern goedgekeurd:
- Plan en voorlopige begroting intern afgekeurd:
- plan en voorlopige begroting opgestuurd naar fonds(en):
- plan en voorlopige begroting goedgekeurd door fonds(en):
- begroting en dekingsplan uitgewerkt:

At the bottom of the modal, the text 'beoordelen en' is partially visible.

Veel activiteiten bij EBD zijn afhankelijk van taken die uitgevoerd moeten worden. In het systeem kunnen de taken worden gerepresenteerd als het afvinken van selectievakjes. Het afvinken van selectievakjes kan door de workflow-manager als trigger dienen om opvolgende taken uit te voeren.

4.8.3 Project bewerken tab 3



Het traject dat een project bij EBD doorloopt ligt redelijk vast. Deadlines zijn al bij het aanmaken van het project bekend. Er zijn een aantal deadlines, en vlak voor, op, of na deze deadlines kunnen automatisch acties worden getriggerd in het workflow-proces. De triggers zijn gebaseerd op een datum, en een aantal dagen positief of negatief ten opzichte van de datum. De server waarop het workflow-systeem draait, zorgt ervoor dat er periodiek een aanroep wordt gedaan om deze automatische trigger mogelijk te maken.

5 Klassenspecificaties

Een klein voorbeeld van de code is de klasse `CmsDataWorkflowActivity`. Deze klasse is een data-object, en correspondeert met een bijbehorende activiteit in de database. Onderstaande code is geschreven in de stijl die Content Power hanteert. Ook alle andere code is vergelijkbaar opgesteld, maar is niet relevant om in het verslag op te nemen.

```
1
2 <?php
3 /**
4  * @author      Content Power, Rob Ruigrok en Jan Pieter Mars
5  * @package     CBB/DataObjects/Workflow
6  */
7
8 /**
9  * Super klasse
10 */
11 require_once($sCBBDir . 'DataObjects/CmsDataObject.php');
12
13 /**
14  * @version     1.0.1 – 16 oktober 2011
15  * @package     CBB/DataObjects/Workflow
16  *
17  * @copyright   1.0.1, 16 oktober 2011, Rob Ruigrok, Activiteit-instantie kan
18  *              worden opgevraagd
19  * @copyright   1.0.0, 12 oktober 2011, Rob Ruigrok, Gemaakt
20  */
21 class CmsDataWorkflowActivity extends CmsDataObject {
```

```

22
23 /**
24 * Methode om van een gegeven tabelnaam alle naar deze activiteit gelinkte items
    uit te voeren
25 *
26 * @param string $a_sProcesTableName de naam van de tabel met verwerkings of
    communicatiemethodes
27 * @return boolean true als alle processen zonder problemen zijn
    uitgevoerd. Anders false
28 * @author Rob Ruigrok
29 * @since 1.0.0 – 12 oktober 2011
30 */
31 protected function bExecuteMethodsFromTable($a_sProcesTableName,
    $a_sProcesFieldId, $a_aParameters) {
32     global $oDB;
33
34     // haal de processen op uit de database
35     $iActivity = $this->iGetId();
36     $aWfProcessingMethods = $oDB->aGetIdList($a_sProcesTableName, array('lActivity'
    , '=' , $iActivity));
37
38     // voer elk proces achtereenvolgens uit
39     $bSuccess = true;
40     foreach ($aWfProcessingMethods as $oProcessingMethod) {
41         $iProcessingTabel = $oProcessingMethod->uGetValue($a_sProcesFieldId);
42         $oProcessingTabel = $oDB->oGetItem($iProcessingTabel);
43         if (!is_object($oProcessingTabel)) {
44             $this->vError(3,3, sprintf('Object met id %s kan niet worden opgehaald',
    $iProcessingTabel));
45             continue;
46         }
47         $bSuccess |= $oProcessingTabel->bExecuteMethod($a_aParameters);
48     }
49
50     return $bSuccess;
51 }
52
53
54 /**
55 * Geeft het proces terug dat behoort bij deze activiteit
56 * @return CmsDataWorkflowProcess het bijbehorende proces
57 * @author Jan Pieter Mars
58 * @since 1.0.0 – 12 oktober 2011
59 */
60 public function oGetProcess() {
61     global $oDB;
62     $oProcess = $oDB->oGetItem($this->uGetValue('lProcess'));
63     return $oProcess;
64 }
65
66
67
68
69 /**
70 * Methode om van een gegeven tabelnaam alle naar deze activiteit gelinkte items
    uit te voeren
71 *
72 * @param string $a_sProcesTableName de naam van de tabel met verwerkings of
    communicatiemethodes
73 * @return boolean true als alle processen zonder problemen zijn
    uitgevoerd. Anders false
74 * @author Rob Ruigrok
75 * @since 1.0.0 – 12 oktober 2011
76 */
77 public function bExecuteProcessingAndCommunication($a_aParameters) {
78     $bSuccess = true;

```

```

79     $bSuccess |= $this->bExecuteMethodsFromTable('cWfMethodProcessing', '
      lProcessingId',$a_aParameters);
80     $bSuccess |= $this->bExecuteMethodsFromTable('cWfMethodCommunication', '
      lCommunicationId',$a_aParameters);
81     return $bSuccess;
82 }
83
84
85
86
87 /**
88 * Geeft een instantie van deze activiteit terug. Indien deze niet bestaat wordt
      een nieuwe instantie aangemaakt
89 *
90 * @param string $a_iIdentification    het id van het item dat gekoppeld is aan
      dit workflow-proces
91 * @return CmsDataWorkflowActivityInstance  Het object van de instantie
92 * @author Rob Ruigrok
93 * @since 1.0.0 - 12 oktober 2011
94 */
95 public function oGetActivityInstance($a_iIdentification) {
96     global $oDB;
97     // eerst het proces ophalen
98     $oProcess = $this->oGetProcess();
99     if (!is_object($oProcess)) {
100         $this->vError(3,3,'Proces kan niet worden opgevraagd');
101         return null;
102     }
103     // daarna de bijbehorende procesinstantie ophalen
104     $oProcessInstance = $oProcess->oGetProcessInstance($a_iIdentification);
105     if (!is_object($oProcessInstance)) {
106         $this->vError(3,3,'Procesinstantie kan niet worden opgevraagd');
107         return null;
108     }
109
110     // vervolgens kijken of er een instantie van de activiteit bestaat
111     $iProcessInstance = $oProcessInstance->iGetId();
112     $iActivity = $this->iGetId();
113     $aWhereClause = array( array('lActivity','=',$iActivity),
114         array('lProcessInstance','=',$iProcessInstance));
115     $oActivityInstance = $oDB->oGetFirstFromList('cWfActivityInstance',
116         $aWhereClause);
117     if ($oActivityInstance) {
118         return $oActivityInstance;
119     } else {
120         // en zo niet, dan wordt een activiteit instantie aangemaakt
121         $oNewInstance = $oDB->oGetPrototype('cWfActivityInstance');
122         $oNewInstance->vSetValue('lActivity', $this->iGetId());
123         $oNewInstance->vSetValue('lProcessInstance', $oProcessInstance->iGetId());
124         $iInstanceId = $oNewInstance->iStore();
125         return $oNewInstance;
126     }
127 }

```


F Testplan

Testplan - Bachelorproject IN3405
Workflow-management in een Content Managementsysteem

Rob Ruigrok
R.J.Ruigrok@student.tudelft.nl
1371746

Jan Pieter Mars
J.P.Mars@student.tudelft.nl
1381075

7 december 2011



ContentPower
informatiebeheersystemen

Voorwoord

In dit document wordt geanalyseerd hoe tests op de workflow-module worden uitgevoerd, en welke onderdelen getest dienen te worden. De International Standards Organisation (ISO) heeft een standaard opgesteld voor de verschillende criteria waaraan software moet voldoen. Deze criteria zijn genoemd in ISO-norm 9126, en beschrijven zes verschillende kwaliteitskenmerken van software. Dit document is opgebouwd aan de hand van deze kwaliteitskenmerken.

Samenvatting

Voor de testfase zijn er twee type tests, de implementatietest en de gebruikerstest.

- Bij de implementatietest wordt er geanalyseerd of de code correct werkt, en of er aan de requirements is voldaan. Deze tests vinden plaats door middel van regressietests, als krachtiger alternatief voor unit tests.
- Bij de gebruikerstest wordt gekeken naar de gebruiksvriendelijkheid van het programma, en of alles duidelijk en functioneel is.

Daarnaast worden de verschillende criteria waaraan de code moet voldoen behandeld. Verder worden in dit document de verantwoordelijkheden en risico's bekeken.

Inhoudsopgave

1	Inleiding	110
2	Aanpak	110
2.1	Implementatietest	110
2.1.1	Regressietests	110
2.1.2	Testen op functionaliteit	112
2.2	Gebruikerstest	112
2.3	Gebruikerstest door Evenementenbureau Delft	112
2.3.1	Testen op gebruiksvriendelijkheid	113
2.3.2	Gebruikers	113
2.3.3	Ontwikkelaars	113
3	Criteria ISO-norm 9126	113
3.1	Functionaliteit	114
3.2	Betrouwbaarheid	114
3.3	Bruikbaarheid	114
3.4	Efficiëntie	114
3.5	Onderhoudbaarheid	114
3.6	Overdraagbaarheid	114
4	Verantwoordelijkheden	114
5	Risico's	115
	Referenties	116

1 Inleiding

Om het correct functioneren van de code te garanderen is dit testplan opgesteld. Dit document beschrijft hoe de code en functionaliteit getest kan worden. Dit is nodig voor de waarborging van de kwaliteit van het eindproduct, en het maakt de kans op problemen bij gebruik door eindgebruikers kleiner.

2 Aanpak

Het testplan bestaat uit twee delen.

- In het eerste deel van het testplan worden alle activiteiten vastgelegd die onderdeel uitmaken van het workflow-framework getest door de ontwikkelaars.
- In het tweede deel van het testplan wordt de gebruikerstest beschreven. Het testplan zal uitgevoerd worden door de eindgebruikers. Dat zijn medewerkers van Content Power, en medewerkers van het bedrijf dat in de testcase wordt gebruikt.

2.1 Implementatietest

De implementatietest bestaat uit een aantal verschillende tests. Deze tests zijn opgesteld aan de hand van de functionele en niet-functionele eisen uit het RAD. Het uiteindelijke doel is dat alle tests slagen, dat wil zeggen dat alle resultaten van de unit-tests en regressietests correct zijn.


2.1.1 Regressietests

Tijdens het implementatieproces worden regressietests geschreven. Dit zijn tests voor alle afgeronde werkende stukken code in het framework. Telkens als een functionerend stuk code is geschreven, wordt tegelijkertijd een regressietest hiervoor geschreven. Met regressietesten wordt gecontroleerd of de niet aangepaste onderdelen van een applicatie nog steeds juist werken, als er elders nieuwe functionaliteit is toegevoegd. Wanneer dit niet het geval is, wijst dit op fouten en moeten deze gecorrigeerd worden.

Om de regressietesten te kunnen uitvoeren wordt een testsuite gebouwd waaraan verschillende tests toegevoegd kunnen worden. Voor elke test wordt er automatisch een nieuwe/schone context gecreëerd waarbinnen de testen kunnen worden uitgevoerd. Hieronder is het startscherm te zien met daarin de tests die uitgevoerd kunnen worden.

Bachelorproject test suite

Jan Pieter Mars & Rob Ruigrok



ContentPower
informatiebeheersystemen


- Fonds aanmaken
- Geheel proces aanmaken

[Start de test suite](#)

Nadat alle testen zijn uitgevoerd volgt een scherm met resultaten. Indien er een fout is gevonden tijdens een test, wordt dit achter de test door een rood blokje weergegeven.

Bachelorproject test suite

Jan Pieter Mars & Rob Ruigrok



ContentPower
informatiebeheersystemen

- Fonds aanmaken [error](#)
- Geheel proces aanmaken gelukt

[Terug](#)

Achter de test staat een link. Wanneer op deze link wordt geklikt, opent een pop-up, met daarin alle fouten en backtraces van de fouten in de test.



2.1.2 Testen op functionaliteit

Alle klassen dienen te worden getest op de aanwezigheid van functionaliteiten.

- Het framework is pas volledig als alle must-have eisen uit het MoSCoW zijn geïmplementeerd. Deze functionaliteitseisen hebben de hoogste implementatieprioriteit, en vormen samen de basis voor een werkende applicatie.
- De functionaliteiten onder het kopje should-have zijn niet kritiek, maar wel zeer gewenst. Bij een should-have eis kan een functionaliteit die een vergelijkbare eigenschap heeft ook voldoende zijn.
- De functionaliteiten onder het kopje could-have zijn optioneel, en zullen alleen worden geïmplementeerd als er tijd over is.
- De functionaliteiten onder het kopje won't-have kunnen in de toekomst interessant zijn voor vervolgoopdrachten, maar komen nu niet aan bod.

2.2 Gebruikerstest

Bij het testen van de gebruiksvriendelijkheid is het de bedoeling dat aan alle punten uit de onderstaande checklist wordt voldaan. Bovendien moeten de gevallen die door de usecases worden beschreven uitvoerbaar zijn op het systeem.

2.3 Gebruikerstest door Evenementenbureau Delft

In het RAD is een case van Evenementenbureau Delft beschreven. De workflow die behoort bij het organiseren van projecten, zal als gebruikerstest voor het project dienen. Dat betekent dat de volgende stappen worden ondernomen:

- David Lansen zal ons een gedetailleerd stappenplan aanleveren over de workflow.
- De workflow zal door ons gemodelleerd worden in de workflow-module in het CMS
- In november zal een afspraak gemaakt worden met David om de workflow te testen. Deze test houdt onder andere in:
 - De projectmodule, en audit-trail zijn de enige zichtbare modules voor David.
 - De modules moeten gebruiksvriendelijk zijn, zoals verderop wordt gedefinieerd.
 - Een stappenplan voor het toevoegen van een project zal worden opgesteld. Dit zal in de week voorafgaand aan de gebruikerstest worden uitgevoerd, omdat de interface op dat moment pas definitief zeker is, en de gebruikerstest daardoor niet constant aan veranderingen onderhevig is.
- De resultaten en opmerkingen van de test zullen voor zover dat mogelijk is binnen de tijd nog worden verwerkt, verder zal de feedback worden opgenomen in het eindverslag.

2.3.1 Testen op gebruiksvriendelijkheid

Het framework moet gebruiksvriendelijk zijn, dat wil zeggen dat alle mogelijkheden voor de gebruiker duidelijk zichtbaar aanwezig en functioneel moeten zijn. Een checklist voor het testen van de gebruiksvriendelijkheid is hieronder opgesteld. Deze moet door de eindgebruiker worden ingevuld. Een aantal vragen aan de gebruiker zijn:

- Is het duidelijk hoe een proces aangemaakt kan worden?
- Is het duidelijk hoe een activiteit aan een proces toegevoegd kan worden?
- Is het duidelijk waar alle instellingen en opties voor dienen?
- Zijn er onderdelen die u mist?
- Zijn er onderdelen die u omslachtig vindt werken, of eenvoudiger kunnen?
- Hebben alle knoppen en functies een duidelijke naam en logische plaats in de applicatie?

2.3.2 Gebruikers

Door een praktijkvoorbeeld als testcase te schrijven, laten we een gebruiker, klant van Content Power, een aantal handelingen verrichten. Deze handelingen dienen bij voorkeur gebruik te maken van alle aanwezige functionaliteit van het framework. Wanneer de gebruiker ergens moeite mee heeft, iets niet kan vinden, of op andere problemen stuit, is dat een teken dat de betreffende functionaliteit mogelijk voor verbetering vatbaar is. Alle feedback die volgt uit deze testsessie moet worden vastgelegd en onderzocht worden wat daadwerkelijk verbeterd of verduidelijkt moet worden.

2.3.3 Ontwikkelaars

Ontwikkelaars, medewerkers van Content Power zijn ook onderdeel van de gebruikerstest. Zij dienen te controleren of het systeem correct functioneert bij allerlei manieren van interactie en randgevallen, met als hoofddoel het vinden van gevallen waarin het framework incorrecte resultaten of fouten oplevert.

3 Criteria ISO-norm 9126

De ISO-norm 9126 beschrijft de kwaliteitskenmerken van software waarbij de kenmerken in een zestal hoofdcategorieën worden onderscheiden. [1]

3.1 Functionaliteit

De software moet functioneel zijn en de belangrijkste functies bevatten. Dit betekent dat in ieder geval de must-haves en eigenlijk ook de should-haves uit de requirements zijn geïmplementeerd.

3.2 Betrouwbaarheid

Betrouwbaarheid houdt in dat de software stabiel draait, en om weet te gaan met fouten die zich tijdens de uitvoering van processen voor kunnen doen. Dit houdt in dat de software fouten zelf kan herstellen en niet crasht.

3.3 Bruikbaarheid

Bij bruikbaarheid is het van belang dat de workflow-module duidelijk te begrijpen en bruikbaar is voor de gebruikers, en dat gebruikers in staat zijn om met alle functionaliteit die de workflow-module biedt om te gaan. In geval van het workflow-framework wordt de module alleen gebruikt door Content Power. De module moet bruikbaar zijn, maar mag technische kennis vereisen. Dit moet dan wel goed beschreven worden in de technische documentatie die is bedoeld voor Content Power.

3.4 Efficiëntie

Het framework en alle onderliggende onderdelen moeten efficiënt omgaan met geheugen, proces-tijd en het netwerk. Van functionaliteit waarvan bekend is dat het niet efficiënt is, wordt onderzocht of er een alternatief mogelijk is, of hoe een optimalisatie kan plaatsvinden. Een aantal tools om snelheid en geheugengebruik te testen is beschikbaar bij Content Power, en kan hiervoor worden gebruikt. Efficiëntie verhoogt de response-snelheid van het systeem en vermindert de load op de servers.

3.5 Onderhoudbaarheid

Bij onderhoudbaarheid gaat het erom hoe gestructureerd en onderhoudbaar de code is. Goed onderhoudbare code is netjes, en bespaart veel tijd bij het fouten opsporen (debuggen) en oplossen. Bovendien zijn toekomstige aanpassingen en uitbreidingen beter te implementeren in nette code.

3.6 Overdraagbaarheid

Overdraagbaarheid geeft aan in hoeverre het framework kan draaien op verschillende systemen zonder al te veel acties uit te voeren om het framework aan de praat te krijgen. Het framework hoeft alleen op interne servers van Content Power overdraagbaar te zijn. Daarnaast wordt er ook onderzocht of er onderdelen in het framework aanwezig zijn die taken vervullen die door andere software uitgevoerd kunnen worden. Als dat het geval is, kan dat soms voorkomen dat het wiel opnieuw uitgevonden moet worden. De belangrijkste eis blijft wel dat de integratie probleemloos werkt.

4 Verantwoordelijkheden

Content Power is de eindverantwoordelijke en zal alle risico's dragen omtrent het gebruik van de geschreven code. Wij, Rob en Jan Pieter zijn alleen verantwoordelijk voor de bruikbaarheid van de geschreven code gedurende de looptijd van het bachelorproject.

5 Risico's

De grootte van risico's die het systeem mogelijk met zich meebrengt, zullen afhangen van de uiteindelijke toepassing waar de klant het systeem voor gebruikt. Er kunnen zich op twee manieren problemen voordoen in een systeem:

- In het geval van storing op een server of fouten in de workflow-module, kan niet worden gegarandeerd dat een proces correct verloopt. Deze fouten kunnen zich op allerlei manieren voordoen. Niet altijd zal een foutmelding het gevolg zijn. Bij deze fouten wordt een taak mogelijk niet volledig uitgevoerd, wordt dubbel uitgevoerd, of bijbehorende data is corrupt.
- In het geval van een fout of vergissing in het workflow-model van een klant, werkt de workflow-module zelf wel correct, maar veroorzaakt de modelfout problemen in het proces. Een activiteit uit een proces ontbreekt, of de afhankelijkheden tussen activiteiten in een proces zijn niet goed vastgelegd. Deze problemen dienen door de klant zelf herkend te worden.

Referenties

- [1] ISO-9126:2001, “Standaard voor kwaliteitskenmerken van software,” 2001.

G Gebruikerstest

Gebruikerstest Evenementenbureau Delft
Bachelorproject IN3405
Workflow-management in een Content Managementsysteem

Rob Ruigrok
R.J.Ruigrok@student.tudelft.nl
1371746

Jan Pieter Mars
J.P.Mars@student.tudelft.nl
1381075

7 december 2011



ContentPower
informatiebeheersystemen

1 Gebruikerstest stappenplan EBD

1.1 Overzicht van projecten

De verschillende stappen van het bedrijfsproces van Evenementenbureau Delft hebben wij gemodelleerd in de workflow-module. Om de uitvoering van projecten eenvoudig te kunnen volgen hebben we voor jullie een mooie interface gemaakt, deze is te zien via de volgende link:
<http://bscproject.acc.contentpower.net/audittrail>

Vervolgens komt u in het volgende scherm terecht:



Figuur 1: overzicht van projecten

Hierin kunt u de verschillende projecten zien die zijn aangemaakt. Voor elk project zijn de afzonderlijke activiteiten in kolommen te zien, en elke kleur in een kolom staat voor een status.

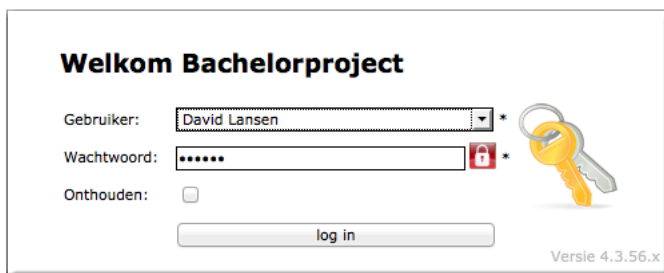
1.2 CMS

Het beheeren van projecten en fondsen gebeurt in het CMS.

<http://bscproject.acc.contentpower.net/edit/>

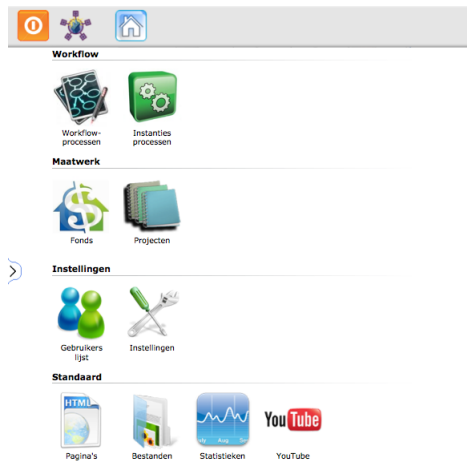
Login naam: David Lansen of Susanne Dix

wachtwoord: test



Figuur 2: inlogschermb van het CMS

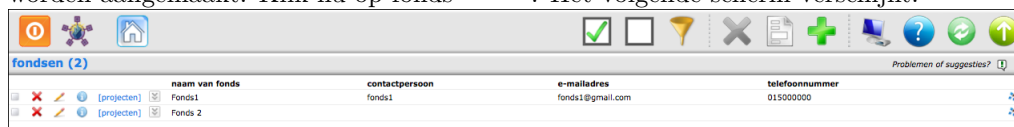
Na het inloggen verschijnt het hoofdschermb:



Figuur 3: hoofdscherm van het CMS

De categorie maatwerk in het CMS is hier belangrijk, want daar kunnen fondsen en projecten

worden aangemaakt. Klik nu op fonds . Het volgende scherm verschijnt:



Figuur 4: beheren van fondsen

Voeg hier een fonds toe door op  te klikken. De volgende pop-up verschijnt:

standaard info

Voeg fonds toe

naam van fonds:

contactpersoon:

e-mailadres:


telefoonnummer:

Figuur 5: het toevoegen van een fonds

Vul hier de gegevens in die relevant zijn voor het fonds, en klik vervolgens op opslaan. Nu is er een fonds toegevoegd met deze naam. Aan dit fonds kunnen projecten worden gekoppeld. Ga

nu terug naar het beginscherm van het CMS door op  te klikken (rechtsboven).



Nu gaan we een project toevoegen. Klik op  om naar de projectenmodule te gaan. Het volgende scherm verschijnt:

	naam van het project	inkomsten in euro's	projectleider vanuit EBD
	Project voorbeeld 1	€ 20000,00	Jan
	Project voorbeeld 2	€ 10000,00	Piet

Figuur 6: beheren van projecten

Hierin ziet u een aantal bestaande projecten als voorbeeld. We gaan nu een nieuwe project

aanmaken. Klik weer op . Het volgende scherm verschijnt:

Voeg project toe

opslaan annuleer

naam van het project:

inkomsten in euro's:

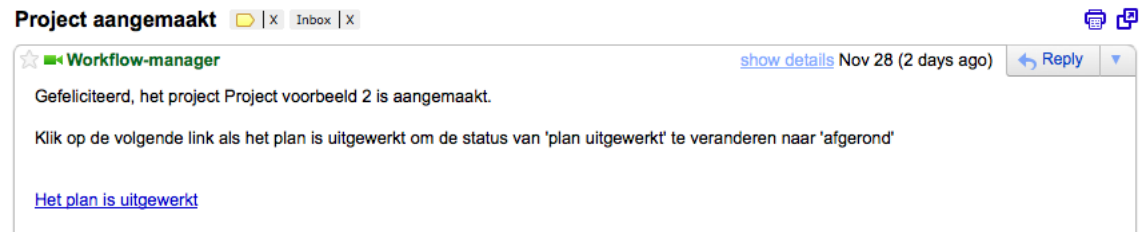
geld ontvangen:

projectleider vanuit EBD:

fondsen: [-- selecteer --](#)

Figuur 7: nieuw project aanmaken

Vul het formulier op het eerste tabblad volledig in. De naam van het project en de projectleider zijn later in het proces erg belangrijk. U kunt ook deadlines invoeren door naar het tabje 'deadlines' te gaan. Zodra alles is ingevoerd, kan het formulier worden opgeslagen. Nu is er een project aangemaakt. Kijk in het projectenoverzicht dat dit inderdaad het geval is. Verder is er ook een e-mail gestuurd naar de projectleider waarin staat vermeld dat er een nieuw project is aangemaakt.



Figuur 8: e-mail na aanmaken van een nieuw project

In deze e-mail staat ook een link waarop geklikt kan worden als de volgende activiteit (plan uitwerken) afgerond is. Nadat u op de link heeft geklikt komt u op de website terecht waarop feedback vermeld staat.



Figuur 9: feedback na klikken op de link in de e-mail.

Via de mail is het mogelijk om het volledige workflow-proces te doorlopen, en bij elke stap zal het projectenoverzicht bijgewerkt worden.

Probeer nu het volledige proces van dit project te doorlopen met behulp van de e-mails.

H Aanbevelingen van Software Improvement Group

De code van het systeem scoort bijna 5 sterren op het onderhoudbaarheidsmodel, wat betekent dat de code bovengemiddeld onderhoudbaar is. De score wordt naar beneden gehaald door de Unit size en Unit Complexity.

Voor Unit Size wordt er gekeken naar het percentage code dat bovengemiddeld lang is. Het opsplitsen van dit soort methodes in kleinere stukken zorgt er ook weer voor dat elk onderdeel makkelijker te begrijpen, te testen en daardoor eenvoudiger te onderhouden wordt. Commentaar regels zoals `'// prototype opslaan in de cWfMethod* tabel'` en `'// controle of het opslaan is gelukt'` geven meestal al aan dat er aparte blokken van functionaliteit te onderscheiden zijn binnen methodes welke makkelijk los te trekken zijn. Het is sterk aan te raden kritisch te kijken naar de langere methodes binnen dit systeem en deze waar mogelijk op te splitsen.

Voor Unit Interfacing wordt er gekeken naar het percentage code in units met een bovengemiddeld aantal parameters. Doorgaans duidt een bovengemiddeld aantal parameters op een gebrek aan abstractie. Daarnaast leidt een groot aantal parameters nogal eens tot verwarring in het aanroepen van de methode en in de meeste gevallen ook tot langere en complexere methoden. Wat in dit systeem opvalt is dat er op veel plaatsen naast parameters ook gebruik wordt gemaakt van globale variabelen, gemiddeld zijn er twee 'global' statements per source-file te vinden. Het gebruik van globale variabelen zorgt voor impliciete afhankelijkheden tussen methodes, wat later voor problemen kan zorgen. Het is daarom aan te raden te bekijken of het gebruik van globale variabelen kan worden voorkomen, of in ieder geval gecentraliseerd kan worden.

Over het algemeen scoort de code bovengemiddeld, hopelijk lukt het om dit niveau te behouden zodra het systeem verder groeit. Het uitbreiden van de alreeds bestaande test-suite (en ervoor zorgen dat alle tests te allen tijde slagen) kan hierbij helpen.