

Eindverslag

Generieke sidebar plug-in

voor IBM Lotus Notes & Domino

AedifiComm

| | |
|---------------------|--------------------------------|
| Opdrachtgever: | F.J.C. Spruit |
| Bedrijfsmentor: | N.G. Scheppink |
| Begeleider: | D.R. Stikkolorum |
| Expert: | A.G.P. Pronk |
| Afstudeerder: | Chris den Arend |
| Email afstudeerder: | Chris.den.Arend@aedificomm.com |
| Plaats: | Barendrecht |
| Datum: | 02-06-2010 |
| Versie: | 1.0 |



Voorwoord

Dit is het eindverslag van de afstudeeropdracht voor de opleiding Technische Informatica aan de Haagse Hogeschool/Academie voor ICT & Media. Tijdens de opdracht heb ik één van de softwarepakketten van het bedrijf AedifiComm gebruikersvriendelijker gemaakt. Dat is gebeurd door een plug-in die de gebruiker kan installeren.

Na vier jaar studeern is dit verslag de laatste opdracht voor de studei. In die vier jaar heb ik veel geleerd, waardoor ik nu kan afstuderen. Nu ik aan het einde ben van mijn afstuderen zijn er een aantal mensen die ik wil bedanken.

Allereerst Dave Stikkolorum, tijdens heel mijn opleiding was hij mijn studieloopbaanbegeleider, daarnaast begeleider van mij stage en afstuderen. Na een moeilijk eerste jaar op de opleiding ging de voortgang daarna voorspoedig, mede dankzij zijn adviezen. Daarnaast heeft hij als begeleider van deze afstudeeropdracht duidelijke feedback gegeven op het moment wanneer dat nodig was.

Als tweede wil ik alle docenten van de opleiding Technische Informatica aan de Haagse Hogeschool bedanken voor de kennis die ze hebben overgedragen.

Als derde wil ik Nico Scheppink bedanken, mijn begeleider vanuit AedifiComm. Hij investeerde tijd en enthousiasmeerde mij in de opdracht, waardoor mijn motivatie vergroot werd.

Als vierde een bedankje voor Annette Spruit, mijn tweede begeleider vanuit AedifiComm. Bij aanvang van de afstudeeropdracht heb ik met haar besproken welke opdracht geschikt was voor mij. Daarnaast heeft ze ervoor gezorgd dat alle middelen die nodig waren voor de start van mijn opdracht aanwezig waren.

Het vijfde en laatste bedankje wil ik besteden aan AedifiComm en de opdrachtgever, die het voor mij mogelijk hebben gemaakt om af te studeren. Tijdens het project had ik alle benodigdheden tot mijn beschikking. Daarnaast wil ik ook mijn collega's bedanken, het samenwerken verliep positief en als ik vragen had namen ze altijd alle tijd om die vragen te beantwoorden.

Barendrecht, 24 mei 2010
Chris den Arend

Samenvatting

Opdracht

De opdracht van dit afstudeerproject is het maken van een generieke plug-in voor de sidebar van IBM Lotus Notes en Domino. Deze opdracht is gegeven door het bedrijf AedifiComm en aangenomen door Chris den Arend van de opleiding Technische Informatica op de Haagse Hogeschool.

AedifiComm heeft het softwarepakket CSP ontwikkeld, dat is gebaseerd op het IBM Lotus Notes platform. Dit platform bevat een sidebar waarin de plug-in moet komen. De gebruikers van CSP doen er nu lang over om naar relevante informatie te navigeren. Om die reden wil AedifiComm een plug-in waar de gebruikers informatie in kunnen zetten die altijd zichtbaar is. Het woord generiek in de opdrachtschrijving houdt in dat de plug-in met alle modules van CSP werkt.

Aanpak

Het project heeft nog geen geschiedenis en de oplevering is een werkend eindproduct. Om die reden moeten alle ontwikkelfasen doorlopen worden. Voor het project is een incrementele ontwikkelstrategie gekozen en er worden twee incrementen doorlopen.

Verloop

Het doel van increment één was een prototype van de plug-in. Het prototype moest te installeren en te deïnstalleren zijn. Daarnaast moest het mogelijk zijn om informatie uit een vastgestelde module uit CSP zichtbaar te maken in de plug-in. Het resultaat van increment één was een prototype dat geïnstalleerd en gedeïnstalleerd kon worden. De mogelijkheid om een vastgestelde module uit CSP zichtbaar te maken was er niet. Wel werden de kolommen van elke geselecteerde gegevensbron uit CSP zichtbaar in de plug-in. Het prototype was zodanig generiek dat de kolommen van iedere geselecteerde CSP module zichtbaar werden.

Het doel van increment twee was een werkende versie maken van de plug-in. De voornaamste eisen waren dat de inhoud van elke module zichtbaar moest worden in de plug-in. Verder moest de gebruiker de informatie in de plug-in kunnen filteren, sorteren en doorzoeken. Ook moest het mogelijk zijn om kolommen te verbergen en zichtbaar te maken en de plug-in moest gerefreshed kunnen worden.

Het resultaat van increment twee klopte met de eisen, alleen de functies sorteren en doorzoeken ontbreken in de eerste versie van de plug-in. Tijdens increment twee is een vertraging opgelopen van een week.

Terugblik

Het project is geslaagd, want de opgeleverde plug-in voldoet aan de eisen die de hoogste prioriteit hadden. Op een week vertraging na tijdens increment twee is het project verlopen zoals gepland. Verder heeft AedifiComm meer kennis verworven over het maken van plug-ins voor Lotus Notes. De afstudeerder heeft leren werken met Lotus Notes, Eclipse en CSP. Daarnaast heeft hij kennis opgedaan over hoe een ontwikkelstrategie en methode bepaald moeten worden. Ook heeft hij door dit project ervaren hoe het is om een project met alle ontwikkelfasen te doorlopen.

Inhoudsopgave

| | |
|------------------|---|
| 1 Inleiding..... | 7 |
|------------------|---|

I - Beschrijving situatie bij aanvang van afstuderen

| | |
|--|----|
| 2 Organisatie..... | 8 |
| 2.1 AedifiComm..... | 8 |
| 2.2 CSP..... | 8 |
| 3 Opdracht..... | 10 |
| 3.1 Opdrachtgever en opdrachtnemer..... | 10 |
| 3.2 Probleemstelling..... | 10 |
| 3.3 Doelstelling..... | 10 |
| 3.4 Opdrachtbeschrijving..... | 11 |
| 3.5 Niveau van de opdracht..... | 11 |
| 3.6 Leerdoelen..... | 12 |
| 3.6.1 Technische doelen..... | 12 |
| 3.6.2 Organisatorische doelen..... | 12 |
| 4 Gekozen aanpak..... | 14 |
| 4.1 Ontwikkelstrategie afstudeerproject..... | 14 |
| 4.1.1 Lineaire strategie..... | 14 |
| 4.1.2 Iteratieve strategie..... | 15 |
| 4.1.3 Incrementele strategie..... | 15 |
| 4.1.4 Ongeschikte strategieën..... | 17 |
| 4.1.5 Geschikte strategie..... | 17 |
| 4.2 Werkzaamheden..... | 19 |
| 4.3 Planning..... | 20 |

II - Beschrijving verloop van het project

| | |
|--|----|
| 5 Increment 1..... | 22 |
| 5.1 Plan van Aanpak..... | 22 |
| 5.1.1 Achtergrond..... | 22 |
| 5.1.2 Projectopdracht..... | 22 |
| 5.1.3 Aanpak..... | 22 |
| 5.1.4 Projectinrichting..... | 23 |
| 5.1.5 Kosten en baten..... | 23 |
| 5.1.6 Risicofactoren..... | 23 |
| 5.2 Definitiefase..... | 23 |
| 5.2.1 Onderzoek vergelijkbare plug-ins..... | 23 |
| 5.2.2 Cursus Lotus Notes en Domino en CSP..... | 25 |
| 5.3 Analyse..... | 26 |
| 5.3.1 Eisen achterhalen..... | 26 |
| 5.3.2 Programma van Eisen (PvE)..... | 27 |

| | |
|---|----|
| 5.3.3 Incrementplan..... | 29 |
| 5.4 Ontwerp..... | 30 |
| 5.4.1 Platformen | 30 |
| 5.4.2 Packagediagram | 32 |
| 5.4.3 Klassediagram..... | 34 |
| 5.5 Implementatie..... | 35 |
| 5.5.1 Opbouw van plug-in | 36 |
| 5.5.2 Verandering van selectie | 38 |
| 5.6 Testen..... | 39 |
| 5.6.1 Testplan..... | 39 |
| 5.6.2 Testrapport..... | 39 |
| 6 Increment 2 | 40 |
| 6.1 Analyse | 40 |
| 6.1.1 Eisen achterhalen | 40 |
| 6.1.2 Programma van eisen | 40 |
| 6.1.3 Incrementplan..... | 42 |
| 6.2 Ontwerp..... | 43 |
| 6.2.1 Packagediagram | 43 |
| 6.2.2 Klassediagrammen..... | 45 |
| 6.2.3 Sequentiediagrammen..... | 46 |
| 6.3 Implementatie..... | 49 |
| 6.3.1 Plug-in vullen met gegevens uit CSP | 49 |
| 6.3.2 Acties in de plug-in | 51 |
| 6.4 Testen..... | 53 |
| 6.4.1 Testplan..... | 53 |
| 6.4.2 Testrapport..... | 53 |
| 6.5 Invoer..... | 53 |

III - Terugblik op het afstuderen

| | |
|---|----|
| 7 Conclusies..... | 55 |
| 7.1 Aanpak..... | 55 |
| 7.1.1 Increment 1 | 55 |
| 7.1.2 Increment 2 | 55 |
| 7.2 Plug-in..... | 55 |
| 7.2.1 Prototype van plug-in na increment 1..... | 55 |
| 7.2.2 Plug-in na increment 2 | 56 |
| 8 Aanbevelingen..... | 57 |
| 8.1 Toevoegingen voor de plug-in..... | 57 |
| 8.2 Aanpassing in CSP..... | 57 |
| 8.3 Nieuwe versie Eclipse..... | 58 |
| 9 Eindproduct..... | 59 |
| 9.1 Plan van Aanpak | 59 |
| 9.1.1 Eisen | 59 |

| | |
|---|-------------------------------------|
| 9.1.2 Op te leveren producten | 59 |
| 9.1.3 Planning | 59 |
| 9.2 Programma van Eisen | 59 |
| 9.3 Ontwerp | 60 |
| 9.3.1 Increment 1 | 60 |
| 9.3.2 Increment 2 | 60 |
| 9.4 Plug-in | 61 |
| 9.4.1 Prototype | 61 |
| 9.4.2 Eerste versie van de plug-in | 61 |
| 9.5 Testen | 61 |
| 9.6 Eindverslag | 61 |
| 10 Evaluatie | 63 |
| 10.1 Leerdoelen | 63 |
| 10.1.1 Technische doelen | 63 |
| 10.1.2 Organisatorische doelen | 65 |
| 10.2 Aanpak | 67 |
| 10.2.1 Aanvang afstuderen | 67 |
| 10.2.2 Increment 1 | 68 |
| 10.2.3 Increment 2 | 70 |
| 10.3 Het project | 71 |
| 11 Geraadpleegde literatuur | 73 |
| 11.1 Boeken | 73 |
| 11.2 Websites | 73 |
| 11.3 Overige | 74 |
| Bijlagen | Error! Bookmark not defined. |
| 1 Begrippenlijst | |
| 2 Afstudeerplan | |
| 3 Plan van Aanpak | |
| 4 Incrementplan increment 1 | |
| 5 Programma van eisen versie 1.0 | |
| 6 Ontwerp increment 1 | |
| 7 Testplan increment 1 | |
| 8 Testrapport increment 1 | |
| 9 Incrementplan increment 2 | |
| 10 Programma van eisen versie 2.2 | |
| 11 Ontwerp increment 2 | |
| 12 Testplan increment 2 | |
| 13 Testrapport increment 2 | |
| 14 Stappenplan updatesite (1) | |
| 15 Stappenplan updatesite (2) | |
| 16 Stappenplan updatesite (3) | |
| 17 Checklist volledigheid plan van aanpak | |
| 18 IBM install Expeditor Toolkit | |

1 Inleiding

Dit document is het schriftelijke resultaat van de afstudeeropdracht "Generieke sidebar plug-in voor IBM Lotus Notes & Domino". De opdracht is door AedifiComm gegeven om een van haar softwarepakketten, CSP, gebruikersvriendelijker te maken. Dit rapport beschrijft hoe het project is verlopen en wat er is gedaan tijdens het project. De vraag waar dit eindverslag antwoord op geeft is: Hoe kan de sidebar van het IBM Lotus Notes en Domino platform worden ingezet om door middel van een configureerbare en generieke plug-in de gewenste informatie altijd zichtbaar te maken?

De inhoud van dit document is opgedeeld in drie delen. Het eerste deel gaat over de aanpak zoals die bij aanvang van het project is bepaald. Er wordt beschreven in welke organisatie de opdracht plaatsvindt, wat de opdracht is en op welke manier deze wordt aangepakt.

Het tweede deel vertelt in chronologische volgorde hoe het project is verlopen. In de aanpak is het project opgedeeld in twee incrementen, deze incrementen en elke fase die deze incrementen bevatten worden beschreven.

Het derde deel van dit verslag bestaat uit de conclusies, de aanbevelingen, een bespreking van de opgeleverde producten en de evaluatie. Hier wordt teruggekeken op het gelopen project.

De plug-in die gemaakt is heeft de ontwikkelnaam "ChriSP". Dat is een combinatie van het softwarepakket CSP en de voornaam van de afstudeerder, Chris.

Tijdens het schrijven van dit verslag is gebruik gemaakt van [Noordtzij04]. In hoofdstuk 11 staat de geraadpleegde literatuur. Verder staat in bijlage 1 staat een begrippenlijst, welke bijlagen verder bij dit verslag zijn gevoegd staat in het vervolg van het verslag.

2 Organisatie

In dit hoofdstuk staat binnen welke organisatie de opdracht wordt uitgevoerd. Ook wordt stilgestaan bij het softwarepakket waar het resultaat van het project op aansluit.

2.1 AedifiComm

De afstudeeropdracht wordt uitgevoerd bij het bedrijf AedifiComm. Dit bedrijf is gevestigd in Barendrecht. Volgens [AedifiComm0902] is AedifiComm een onafhankelijke leverancier van diensten en producten op het gebied van kennismanagement, workflowmanagement, messaging-technologie, real time communication en collaboration oplossingen. Binnen het bedrijf wordt gebruik gemaakt van oplossingen van IBM (Lotus, Websphere) en Microsoft (Sharepoint en .NET). Tijdens dit project wordt gewerkt met producten van IBM, namelijk IBM Lotus Notes en Domino.

De focus van het bedrijf is op kennis en software-projecten. Samen met de zusterorganisatie NedPortal (netwerkinfrastructuur) is AedifiComm in staat oplossingen te ontwerpen, implementeren en ook te beheren.

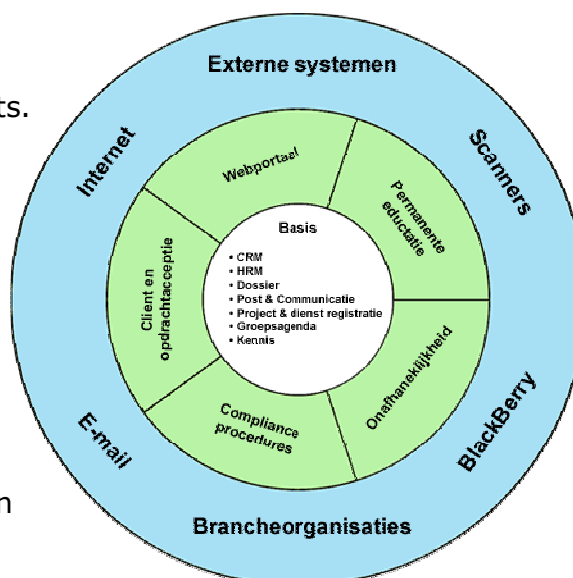
AedifiComm heeft in eigen beheer producten ontwikkeld, variërend van complete dossieroplossingen tot technische beheertools voor Lotus Domino. Deze producten worden over de hele wereld geleverd.

Het bedrijf verdeelt per project werkzaamheden over medewerkers. Er zijn verschillende afdelingen binnen AedifiComm, namelijk sales, secretariaat, administratie en softwareontwikkeling. Alle projecten die met software te maken hebben worden op de afdeling softwareontwikkeling gedaan, zo ook de uitvoering van het afstudeerproject.

AedifiComm heeft verschillende producten ontwikkeld. Eén daarvan is CSP voor accountants. Het afstudeerproject heeft betrekking op dat pakket, daarom komt alleen CSP aan bod in dit verslag, de overige pakketten die AedifiComm ontwikkelt zijn niet van toepassing. CSP biedt ondersteuning voor de werkprocessen van accountants en belastingadviseurs.

2.2 CSP

CSP is modulair opgebouwd, zoals te zien is in afbeelding 2.1. Op [AedifiComm1102] is te lezen dat de basismodule van CSP beschikt over verschillende functies, enkelen daarvan zijn CRM functionaliteit, HRM functionaliteit, elektronisch dossier, post en communicatie, persoonlijke agenda en groepsagenda.



Afbeelding 2.1: modules van CSP

AedifiComm biedt ook een pakket met uitbreidingen op de basismodule. Daarnaast is het mogelijk om maatwerk op CSP te verkrijgen. Op dit moment is CSP een standalone applicatie, een webversie van CSP wordt mogelijk in de toekomst ontwikkeld.

Verschillende ontwikkelaars zijn voortdurend bezig met het uitbreiden en verbeteren van CSP. Tot 2009 werd CSP aangeboden zonder onderhoudscontract. Gevolg daarvan was dat bij nieuwe of andere wensen van de klant altijd maatwerk gedaan werd. Daardoor ontstond de situatie dat elke klant een eigen versie van CSP had. In 2009 is AedifiComm begonnen met het leveren van een onderhoudscontract, het bedrijf heeft er toen voor gezorgd dat veel maatwerk dat bij verschillende klanten aanwezig was in het standaardpakket van CSP opgenomen werd. Alle nieuwe klanten krijgen sindsdien een standaard versie van CSP. Bestaande klanten worden gemigreerd naar de standaardversie.

CSP is ontwikkeld op het IBM Lotus Notes en Domino platform. Recentelijk is een nieuwe versie van IBM Lotus Notes en Domino op de markt gekomen. AedifiComm is met deze versie gaan werken. Deze nieuwe versie heeft veel functies die nog niet gebruikt worden door CSP. Tijdens deze opdracht wordt één van die ongebruikte functies gebruikt voor het ontwikkelen van een voor eindgebruikers nuttige toevoeging aan CSP.

3 Opdracht

In dit hoofdstuk wordt beschreven wie de opdrachtgever en opdrachtnemer zijn. Verder wordt duidelijk waarom dit project wordt uitgevoerd en wat het project inhoudt, daarbij wordt verder gegaan op het afstudeerplan dat te vinden is in bijlage 2. De projectgrenzen en randvoorwaarden van het project komen ook aan bod. Als laatste staat beschreven wat het niveau van de opdracht is en wat de leerdoelen voor de student zijn.

3.1 Opdrachtgever en opdrachtnemer

Het bedrijf dat de opdracht verstrekt is AedifiComm. De opdrachtgever is dhr. F.J.C. Spruit. De bedrijfsmentor van de afstudeerder is dhr. N.G. Scheppink. Tijdens het afstuderen treedt de bedrijfsmentor op namens de opdrachtgever. De opdracht is aangenomen en wordt uitgevoerd door Chris den Arend, student Technische Informatica (TI) van de Haagse Hogeschool, vestiging Delft.

3.2 Probleemstelling

Het softwarepakket CSP bestaat uit diverse modules en er zijn veel handelingen nodig om de meest relevante informatie altijd zichtbaar te maken voor de eindgebruiker. Elke module bevat informatie en voor elke gebruiker heeft het begrip relevante informatie een andere invulling.

Voor dit project is een structureel probleem geconstateerd. Namelijk dat relevante informatie niet op elk moment in CSP weergegeven wordt. Om informatie te vinden moet eerst genavigeerd worden naar de module waar de gezochte informatie in staat. Vervolgens moet binnen die module een weergave geopend worden waarin de juiste informatie te raadplegen is. Al deze acties kosten veel tijd en klikmomenten en dat is niet gebruikersvriendelijk.

Zoals in [IBM0904] staat is in IBM Lotus Notes en Domino versie 8 een sidebar toegevoegd waarin plug-ins gebruikt kunnen worden. Het idee is om deze altijd zichtbare sidebar te gebruiken voor het weergeven van informatie. De hoofdvraag van dit project is: hoe kan de sidebar van het IBM Lotus Notes en Domino platform worden ingezet om door een configureerbare en generieke plug-in de gewenste informatie zichtbaar te maken?

Generiek betekent hier dat de plug-in met elke willekeurige module van CSP, dat zijn er nu circa 20, moet werken. Configureerbaar betekent dat de gebruiker kan bepalen welke informatie uit de gekozen module getoond wordt.

3.3 Doelstelling

De doelstelling van het project is het uitbreiden en daarmee gebruikersvriendelijker maken van het softwarepakket CSP. Het doel is duidelijk geworden na verschillende gesprekken met de opdrachtgever. Het vriendelijker maken van de software gebeurt op twee manieren. Ten eerste kost het de gebruiker minder tijd voordat de gewenste informatie zichtbaar is. Ten tweede is de gebruiker meer en vaker op de hoogte doordat de relevante informatie altijd zichtbaar is. Dit zorgt ervoor dat de

eindgebruiker efficiënter kan werken. Eindgebruikers kunnen eerder actie ondernemen als dat nodig is, waardoor hun klant beter geholpen wordt. In oktober 2008 is versie 8.5 van IBM Lotus Notes op de markt gebracht. Deze versie heeft, net als versie 8.0, een sidebar. AedifiComm heeft nog weinig ervaring opgedaan met het ontwikkelen van een generieke plug-in voor die sidebar. Daarom is het doel van de opdrachtgever ook om meer kennis te krijgen over het ontwikkelen van een plug-in voor de sidebar van Lotus Notes.

3.4 Opdrachtbeschrijving

De opdracht bestaat uit verschillende delen. Het eerste deel is het vastleggen van de eisen voor de plug-in. De eisen staan in een apart document. Het vastleggen van de eisen gebeurt met de MoSCoW methode. Deze methode beschrijft wanneer welke eis gerealiseerd moet zijn aan de hand van zijn prioriteit. Daarnaast wordt beschreven waar de eisen vandaan komen.

Als de eisen vastgelegd zijn wordt het functionele en technische ontwerp gemaakt. Het functionele ontwerp beschrijft hoe de plug-in gaat werken. Het is goed mogelijk dat de opdrachtgever door het functionele ontwerp wat aanpassingen heeft op de eisen. In dat geval worden de eisen aangepast. Het ontwerp wordt gemaakt in UML, door een use-case-diagram een sequentiediagram en een klassediagram. Een eis aan het ontwerp is dat het uitgebreid moet kunnen worden.

Als laatste wordt het eindproduct ontwikkeld, de eerste versie van de plug-in. De plug-in moet voldoen aan het ontwerp en aan de eisen die vastgelegd zijn in het programma van eisen.

3.5 Niveau van de opdracht

Tijdens de afstudeeropdracht wordt een plug-in gemaakt voor de sidebar van IBM Lotus Notes. Op [Socialtext1002] is te lezen dat er veel plug-ins beschikbaar zijn voor de sidebar van Lotus Notes. De te maken plug-in is door verschillende aspecten uniek en niet te vergelijken met bestaande plug-ins:

1. De plug-in die tijdens de afstudeeropdracht gemaakt wordt moet werken met CSP. Voor CSP bestaan nog geen plug-ins, dat zorgt ervoor dat samenwerking met modules van CSP nieuw is.
2. De te maken plug-in moet generiek zijn. Alle modules van CSP moeten werken in de plug-in, zonder dat de plug-in van tevoren weet welke module gebruikt gaat worden. Veel bestaande plug-ins hebben niets te maken met verschillende gegevensbronnen, maar staan op zichzelf. Anderen hebben bepaalde functies met betrekking tot gegevensbronnen, daar ligt in de plug-in wel vast op welke gegevensbron de functie betrekking heeft.
3. De plug-in moet eenvoudig te installeren zijn. Voor het installeren en inrichten van bestaande plug-ins is vaak technische kennis noodzakelijk. Doelstelling van de te ontwikkelen plug-in is dat een gemiddelde eindgebruiker bij een accountantskantoor de plug-in kan installeren en configureren.

Eén van de onderwerpen van de opleiding Technische Informatica is het doorlopen van een ontwikkelproces met betrekking tot software. Deze opdracht is uiterst

geschikt om dat te doen. De opdrachtgever heeft nog geen kennis van de te maken plug-in en de systeemeisen liggen niet eenduidig vast. Het resultaat van de afstudeerperiode wordt een werkend eindproduct. Om dat te bereiken worden alle ontwikkelfasen doorlopen.

3.6 Leerdoelen

Tijdens de afstudeeropdracht zijn er verschillende gebieden waar de afstudeerder kennis opdoet. In deze paragraaf staan de leerdoelen die zijn vastgesteld aan het begin van de afstudeerperiode.

3.6.1 Technische doelen

- Leren werken met Lotus Notes en Eclipse
- Leren werken met niet-relatieve databases
- Leren ontwikkelen van een plug-in met Eclipse
- Ervaring opdoen met de programmeertaal Java

AedifiComm heeft CSP ontwikkeld, dat is gebaseerd op IBM Lotus Notes. Tijdens de opdracht wordt veel gewerkt met Lotus Notes en Eclipse, Lotus Notes is namelijk gebaseerd op Eclipse. De student leert door de opdracht werken met de twee platformen en de koppeling daartussen.

Lotus Notes werkt met niet-relatieve databases. Tijdens de opleiding heeft de student kennis en ervaring opgedaan met relationele databases, maar niet met niet-relatieve databases. Deze afstudeeropdracht geeft daardoor de mogelijkheid om te leren werken met niet-relatieve databases.

Verder biedt het platform Eclipse ontwikkelaars de mogelijkheid plug-ins te ontwikkelen voor verschillende andere platformen, tijdens de opdracht leert de student een plug-in te ontwikkelen met Eclipse.

Ten slotte moet de plug-in worden geschreven in de programmeertaal Java. Deze taal is tijdens de opleiding niet aan bod gekomen, waardoor het leren van deze taal een doel is tijdens de opdracht.

3.6.2 Organisatorische doelen

- Leren bepalen van een ontwikkelstrategie en methode
- Leren doorlopen van alle ontwikkelfasen in één project
- Leren zoeken naar eisen met opdrachtgever
- Leren een product in te voeren

De plug-in die gemaakt wordt bestaat nog niet en het resultaat van de opdracht is de eerste versie van de plug-in. Daardoor worden alle fasen van ontwikkelstrategieën doorlopen. Verschillende fasen zijn tijdens de opleiding aan bod gekomen, sommige niet. Individueel bepalen welke ontwikkelstrategie en methode gebruikt worden tijdens het project en een planning maken die daarbij past is een leerdoel van deze opdracht. De ontwikkelaar en de opdrachtgever zoeken samen naar de beste oplossing voor de probleemstelling. Wel heeft de opdrachtgever verschillende eisen op papier staan, maar nog niet gedetailleerd.

Als laatste wordt het product ingevoerd in CSP. Er moet bepaald worden hoe dat gebeurt. Tijdens de opdracht leert de student hoe dat moet.

4 Gekozen aanpak

In dit hoofdstuk wordt beschreven welke aanpak bij aanvang van de opdracht gekozen is. Het grootste deel van dit hoofdstuk staat ook in het plan van aanpak dat te vinden is in bijlage 3. Toch staat de aanpak ook in dit eindverslag, omdat het dan goed te lezen is afzonderlijk van het plan van aanpak. Als eerste worden de ontwikkelstrategieën besproken en daarna de planning.

4.1 Ontwikkelstrategie afstudeerproject

Ontwikkelstrategieën zijn opgebouwd uit verschillende fasen. De verschillende fasen zijn:

- Definitie
- Analyse
- Ontwerp
- Realisatie
- Testen
- Invoering

Er zijn een aantal ontwikkelstrategieën mogelijk:

1. Lineaire ontwikkelstrategie
2. Iteratieve ontwikkelstrategie
3. Incrementele ontwikkelstrategie

Bij het kiezen van een ontwikkelstrategie zijn er een aantal punten belangrijk bij het maken van die keuze:

- De duur van het project
- Grootte van het project
- Deskundigheid ontwikkelaar
- Ervaring van opdrachtgever met formuleren van systeemeisen
- Zekerheid van systeemeisen

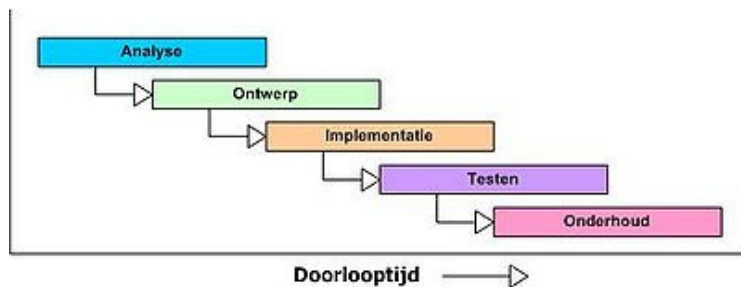
De conclusie van dit hoofdstuk is dat tijdens het project een incrementele ontwikkelstrategie doorlopen wordt. De toelichting voor deze keuze komt in de volgende paragrafen aan de orde. Bij het schrijven van dit gedeelte is gebruik gemaakt van [Bosch07].

4.1.1 Lineaire strategie

Bij de lineaire ontwikkelstrategie worden alle ontwikkelfasen achter elkaar doorlopen. Na het doorlopen van alle fasen is het product af. Deze strategie is goed toe te passen op kleine projecten waar de opdrachtgever de systeemeisen goed kan definiëren. Ook is het belangrijk dat de ontwikkelaars de systeemeisen goed begrijpen, dus ervaring van de ontwikkelaar is wenselijk. Het nadeel van de strategie is dat de opdrachtgever de eisen niet kan aanpassen na de definitie- en analysefase.

Watervalmethode

De watervalmethode valt onder de lineaire ontwikkelstrategieën. Bij deze methode kunnen fasen elkaar overlappen, zie afbeelding 4.1. Vroeger werd deze manier van werken veel gebruikt, ook voor grote projecten, tegenwoordig wordt het minder gebruikt. De watervalmethode is niet geschikt voor projecten waar systeemeisen tijdens het project veranderen.



Afbeelding 4.1 de watervalmethode

4.1.2 Iteratieve strategie

Bij de iteratieve ontwikkelstrategie worden, net als bij de lineaire strategie, alle ontwikkelfasen doorlopen, verschil is dat bij de iteratieve strategie teruggegaan kan worden naar een eerdere fase. Dat kan nodig zijn om een fase te verbeteren. Net zoals bij de lineaire strategie wordt het eindresultaat in een keer opgeleverd. Deze strategie wordt gebruikt bij grote projecten en projecten waar de probleemruimte niet duidelijk is. De nadelen zijn dat het proces moeilijker te beschrijven is en de oorspronkelijke doelstellingen kunnen vervagen.

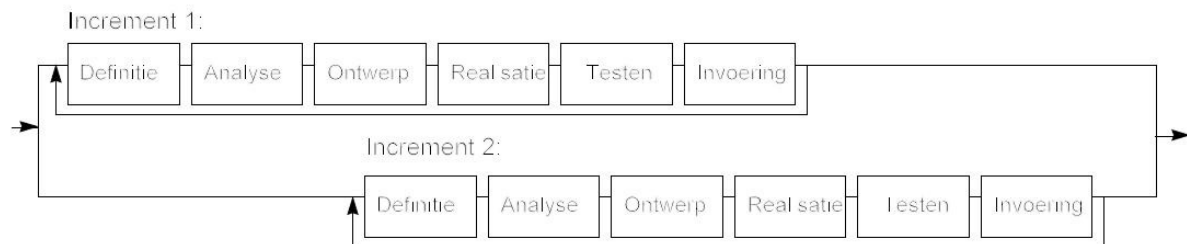
4.1.3 Incrementele strategie

Als laatste de incrementele ontwikkelstrategie. Bij deze strategie worden verschillende incrementen doorlopen, zoals te zien is in afbeelding 4.2. Binnen een increment worden alle ontwikkelfasen doorlopen. Na elk increment is er een resultaat, dat kan bijvoorbeeld een verslag, een prototype of een versie van het product zijn. Er is dus een duidelijk verschil tussen de iteratieve strategie en de incrementele strategie. Bij de iteratieve strategie worden de fasen verbeterd, terwijl aan heel het project gewerkt wordt, bij incrementeel worden telkens gedeelten van het project gemaakt, waarbij de evaluatie van het vorige deel meegenomen wordt om het product te verbeteren. Er kan gezegd worden dat bij iteratief één increment doorlopen wordt, waarbij fasen tijdens dat increment verbeterd kunnen worden. Een combinatie van deze strategieën is mogelijk door meerdere incrementen iteratief te doorlopen.

Incrementeel ontwikkelen wordt veel gebruikt bij grote projecten waar de systeemeisen onzeker zijn. Door verschillende incrementen is er al tijdens het project resultaat waardoor de opdrachtgever een gedeelte van het eindresultaat ziet. Als er een verkeerde interpretatie is geweest van de eisen komt dat vaak bij een prototype aan het licht.

Een nadeel van de incrementele strategie is dat het ontwikkelproces lastig te doorgronden is en dat de doelstellingen tijdens het project onbedoeld kunnen verschuiven.

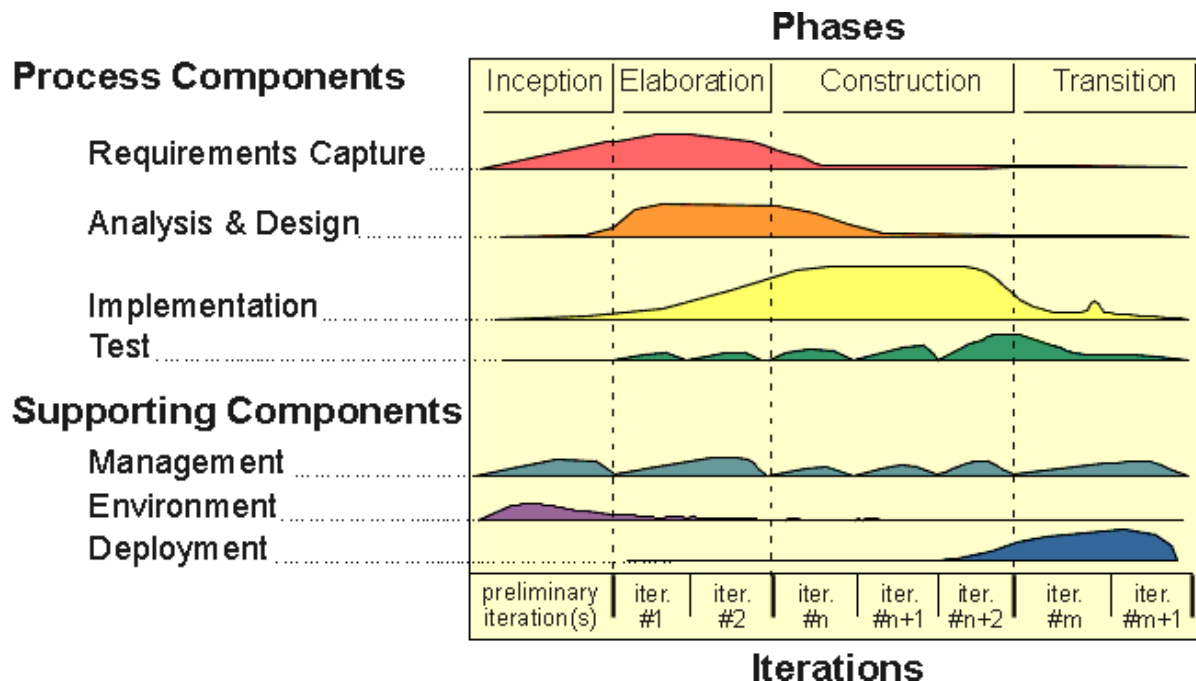
Onder de incrementele strategie vallen de agile methoden. Deze methoden zijn heel 'behendig' en 'lenig', omdat er veel verandering binnen een project mogelijk is. Goede voorbeelden van een agile methoden zijn Scrum en Rational Unified Process (RUP).



Afbeelding 4.2: incrementele ontwikkelstrategie

Rational Unified Process (RUP)

Er zijn verschillende mogelijkheden als het gaat om incrementele en iteratieve ontwikkelmethoden. Een van de bekendste is RUP. Bij RUP wordt vastgehouden aan een aantal best practices. Een daarvan is het iteratief ontwikkelen, het is mogelijk om RUP in te richten als combinatie van de iteratieve en incrementele strategie. Tijdens een iteratie is het voor de opdrachtgever en gebruiker mogelijk om de probleemstelling en de eisen aan te passen. In elke iteratie zijn de ontwikkelfasen weer anders verdeeld. In de eerste iteratie wordt meer tijd besteed aan de eisen en de analyse, in latere iteraties verschuift dat naar implementeren en testen. Dit is goed te zien in afbeelding 4.3.



Afbeelding 4.3: RUP diagram

Meer uitleg over bovenstaande strategieën en methoden die daar gebruik van maken staan in [Larman04].

4.1.4 Ongeschikte strategieën

Zoals eerder genoemd is dit een klein project waarbij de systeemeisen nog niet vastliggen en de ontwikkelaar niet ervaren is. Een lineaire ontwikkelstrategie is niet gepast, omdat de eisen niet vastliggen. De eisen moeten nog duidelijk worden en na een resultaat kunnen de eisen veranderen. Dat is niet mogelijk bij de lineaire strategie.

Een iteratieve strategie is ook niet goed toepasbaar op dit project. De reden is grotendeels dezelfde als bij de lineaire strategie. Bij die twee strategieën wordt het eindresultaat in een keer opgeleverd, waardoor de eisen moeilijk kunnen veranderen aan de hand van een resultaat.

Het is ook mogelijk om iteratief en incrementeel ontwikkelen te combineren. Dan worden er iteraties doorlopen in een increment. Deze manier van werken is niet handig voor dit project, omdat de incrementen kort zijn. Als tijdens een kort increment iteraties doorlopen worden is het lastig om het ontwikkelproces te overzien. Daardoor wordt de focus op het doel van het project verminderd.

4.1.5 Geschikte strategie

De incrementele ontwikkelstrategie sluit wel goed aan bij dit project. Tijdens het project wordt een resultaat neergezet. Aan de hand van het resultaat kan de opdrachtgever de eisen bijstellen of verduidelijken. Dat is van groot belang, omdat

de systeemeisen onzeker zijn en de ontwikkelaar is onervaren met dit soort projecten.

Bij incrementeel ontwikkelen is het mogelijk om componenten uit eerdere incrementen te verbeteren, dat zorgt voor hogere kwaliteit van het product. Ook is met verschillende incrementen de haalbaarheid van het project eerder in te schatten. Na een increment wordt geëvalueerd of de doelstellingen nog haalbaar zijn.

Daarnaast wordt tijdens het project een gedeelte van het product opgeleverd, dus als het project veel vertraging oploopt is er nog wel een bruikbaar deel van het product.

De genoemde keuze is een strategie, geen methode. Een methode is een kant en klare invulling van een strategie, zoals RUP en Scrum. Methodes hebben het kenmerk dat ze een strategie invullen. Daar hoort bij dat ze voorschrijven op welke manier een project doorlopen wordt en welke documenten opgeleverd worden. Het voordeel daarvan is dat de ontwikkelaar of de projectgroep na de keuze van de methode niet hoeft na te denken over de invulling van de strategie. Als gewerkt wordt in een project weet iedereen hoe het project doorlopen wordt, omdat de methode bekend is.

Voor dit project is geen methode gekozen, omdat dit ook nadelen met zich meebrengt. Methodes zijn vaak uitgebreid en er worden veel documenten voorgeschreven. Bij grote projecten waar veel mensen bij betrokken zijn zorgt dat voor duidelijkheid. Bij kleine projecten, zoals dit project, moet de methode zo worden aangepast dat er ook in een klein project met de methode kan worden gewerkt. Het aanpassen van een methode kost veel werk, in dit project kostte het minder tijd om een eigen invulling te geven aan de strategie. Verder wordt met een methode een gedetailleerde invulling gegeven aan het verloop van een project. Bij dit project zou dat betekenen dat er minder flexibel gewerkt kan worden. Een groot voordeel van dit project is dat de opdrachtgever en opdrachtnemer veel contact hebben en waar nodig is het project bij kunnen sturen. Dit voordeel wordt weggenomen als het verloop van het project vast is gelegd in een methode.

Aantal incrementen

Tijdens het project worden twee incrementen doorlopen. Bij het kiezen voor een incrementele ontwikkelstrategie is minder dan twee incrementen al uitgesloten, omdat één increment doorlopen geen incrementele ontwikkelstrategie is. Meer dan twee incrementen is gezien de duur van het project niet praktisch. Er worden documenten opgeleverd tijdens het project en per increment wordt het aantal opleveringen van documenten groter. De verwachting is dat met drie of meer incrementen teveel tijd verloren gaat aan het maken en verbeteren van de documenten. Daarnaast worden de documenten beoordeeld door de bedrijfsmentor en in sommige gevallen ook door de begeleider. Bij drie of meer incrementen is het mogelijk dat de periode waarin een document beoordeeld wordt langer is dan een increment. Dan is het project toe aan de volgende versie van een document, maar de feedback op de voorgaande versie is nog niet beschikbaar. Bij twee incrementen is dat risico kleiner.

Voor de werkzaamheden van dit project is het niet van belang of er twee of drie incrementen doorlopen worden. De opdracht is goed in twee incrementen op te delen, maar dat is ook het geval bij drie incrementen.

De invoerfase van het afstudeerproject is klein, omdat het om een plug-in gaat die in een bestaand systeem komt. De plug-in komt in de sidebar, de invoer daarvan is erg gemakkelijk, omdat er geen veranderingen optreden in het systeem. Verder is ervoor gekozen om geen definitiefase te doorlopen in increment twee, omdat de ontwikkelaar dan bekend is met de omgeving en al heeft onderzocht of er vergelijkbare producten bestaan. In de planning wordt meer duidelijk over de toepassing van de ontwikkelstrategie.

Risico's

Een risico van de gekozen ontwikkelstrategie is dat de eisen tijdens het project onbedoeld kunnen verschuiven. Het is mogelijk om eisen tijdens het project aan te passen of toe te voegen. Deze mogelijkheid kan verkeerd uitpakken, waardoor een eis verandert terwijl dat niet moet.

Dit risico is te voorkomen door goed te beargumenteren waarom een eis verandert. Daarnaast moeten de opdrachtgever en opdrachtnemer beide op de hoogte zijn van veranderingen van eisen. Alleen na overleg en goede argumentatie over waarom een eis moet veranderen mogen eisen aangepast worden.

4.2 Werkzaamheden

Het project bestaat uit twee incrementen en een uitloop. De uitloop wordt gebruikt als er vertraging is tijdens het project. Als er geen vertraging is wordt de beschrijving van de plug-in uitgebreid, zodat een andere ontwikkelaar moeiteloos verder kan gaan met de plug-in. In deze paragraaf wordt besproken welke werkzaamheden horen bij de twee incrementen. Documentatie over het project wordt tijdens alle fasen gemaakt.

Als naar de werkzaamheden wordt gekeken is het project goed op te delen in twee incrementen. Eerder in dit document zijn drie moeilijkheden van dit project naar voren gekomen. Dat de plug-in moet werken in CSP is van toepassing tijdens heel het project. De andere moeilijkheden, het generieke en het installeren en deïnstalleren zijn wel goed te scheiden.

Increment 1

Er is voor gekozen om in increment één de nadruk te leggen op het installeren en deïnstalleren van de plug-in. Om dat te kunnen moet eerst onderzocht worden welke platformen nodig zijn voor de ontwikkeling van de plug-in. Het prototype dat na increment één wordt opgeleverd moet de opdrachtgever een beeld geven over hoe de plug-in eruit komt te zien. Daarom wordt het prototype zo gemaakt dat het werkt met één gegevensbron.

Het resultaat van increment één is een prototype van de te maken plug-in in de sidebar van Lotus Notes. In de plug-in kan één module gesleept worden en daarvan wordt informatie weergegeven. Dat betekent dat de plug-in geïnstalleerd moet kunnen worden in CSP.

Tijdens increment één wordt samen met de opdrachtgever gezocht naar eisen voor de plug-in. Ook wordt een ontwerp gemaakt en aan de hand daarvan wordt de plug-in geïmplementeerd. Als laatste wordt de plug-in getest.

Increment 2

Tijdens increment twee wordt de plug-in generiek gemaakt daarnaast worden functies aan de plug-in toegevoegd. De gebruiker kan dan informatie filteren, zoeken en de weergave van de plug-in aanpassen.

Het resultaat van increment twee is een plug-in in de sidebar van Lotus Notes. In de plug-in kunnen alle modules van CSP geslept worden en de gebruiker kan de plug-in zo configureren dat de gewenste informatie weergegeven wordt.

De werkzaamheden van increment twee liggen minder vast dan die van het eerste increment. Dit komt, omdat de resultaten van het eerste increment invloed hebben op increment twee.

4.3 Planning

In de planning zijn, zoals eerder aangegeven, twee incrementen opgenomen. De globale planning van de afstudeerperiode staat in tabel 3.1.

| | | | |
|------|-------------|-------------|----------|
| Week | 1 t/m 8 | 9 t/m 14 | 15 en 16 |
| | Increment 1 | Increment 2 | Uitloop |

Tabel 3.1: globale planning

Naast de globale planning is er een gedetailleerdere planning, met milestones per week. Deze staat in tabel 3.2. In de planning staat het afstudeerdossier gepland voor week 15. Het afstudeerdossier bestaat uit:

- Afstudeerplan
- Plan van aanpak
- Programma van eisen
- Ontwerp (increment één en increment twee)
- Gedocumenteerde programmacode
- Testplan en testrapport (increment één en increment twee)
- Documentatie over het implementeren van de plug-in
- Documentatie over de invoer van de plug-in.

| Week project (weeknummer) | fase | Milestone |
|------------------------------|-------------------------|---------------------------|
| 1 (6) | Opstart / Definitie | Plan van Aanpak, Planning |
| 2 (7) | Definitie / Analyse | |
| (8) | Afstudeerder is afwezig | |
| 3 (9) | Analyse | Programma van eisen |
| 4 (10) | Ontwerp | |
| 5 (11) | Ontwerp | Functioneel ontwerp |
| 6 (12) | Realisatie | Voortgangsverslag |

| | | |
|----------------|----------------------|--------------------------------------|
| 7 (13) | Realisatie | Oplevering prototype 1 |
| 8 (14) | Testen | |
| 9 (15) | Analyse | Aangepast programma van eisen |
| 10 (16) | Ontwerp | |
| 11 (17) | Ontwerp / Realisatie | Aangepast functioneel ontwerp |
| 12 (18) | Realisatie | |
| 13 (19) | Realisatie | Oplevering eerste versie van plug-in |
| 14 (20) | Testen / Invoer | Testrapport |
| 15 (21) | uitloop | Afstudeerdossier |
| 16 (22) | uitloop | |

Tabel 3.2: planning op weekbasis

Naast de planning die is opgenomen komt er voor elk increment een apart plan. Dat is nodig, omdat de planning die gemaakt is niet zeker is voor increment twee. In die plannen staat ook wat er na increment één en na increment twee opgeleverd wordt.

5 Increment 1

Het project bestaat uit twee incrementen en een uitloop. De uitloop wordt gebruikt als er vertraging is tijdens het project. Als er geen vertraging is wordt de beschrijving van de plug-in uitgebreid, zodat een andere ontwikkelaar moeiteloos verder kan gaan met de plug-in. De fasen die tijdens elk increment aan bod komen worden per stuk beschreven, ook wordt er verteld wat de werkzaamheden en de belangrijke beslismomenten zijn. Het plan van aanpak wordt als eerste en afzonderlijk van de fasen besproken, omdat die niet tijdens een bepaalde fase gemaakt is.

5.1 Plan van Aanpak

Bij een project is het belangrijk dat er overeenstemming is over de uit te voeren opdracht. De opdrachtgever en opdrachtnemer moeten het eens zijn met de opdracht. Voor deze opdracht is het ook van belang dat de begeleider en expert vanuit school op de hoogte zijn van de aanpak van de opdracht, zodat zij in kunnen grijpen als dat nodig is. Er wordt ingegrepen als uit de aanpak blijkt dat de opdracht niet HBO-waardig aangepakt wordt, of als de opdracht te weinig of geen aspecten heeft van de gevolgte opleiding.

De onderwerpen die in het plan van aanpak naar voren komen zijn afgeleid uit de [Noordzij07] en [Grit0902]. Bij AedifiComm is geen standaardopmaak voor een plan van aanpak aanwezig.

5.1.1 Achtergrond

In het plan van aanpak worden de achtergrond van het bedrijf en het project besproken. Doel van de achtergrond is dat ook buitenstaanders zich een beeld kunnen vormen van het project en van het bedrijf waar het project wordt uitgevoerd.

5.1.2 Projectopdracht

Daarna komt het hoofdstuk "projectopdracht", waarin de opdracht uitgebreid besproken wordt. Het doel van dit hoofdstuk is dat de opdrachtgever en opdrachtnemer een duidelijk beeld krijgen van het project. Verder wordt duidelijk wat de hoofdvraag is van het project. Daarnaast wordt het doel van de opdrachtgever met deze opdracht duidelijk en wat er moet gebeuren om dit doel te bereiken. Er worden een aantal globale eisen genoemd en een lijst met op te leveren producten. Als laatste worden de projectgrenzen en de randvoorwaarden genoemd. Deze geven aan tot hoever het project doorgaat en wat wel en niet bij het project hoort.

5.1.3 Aanpak

Het hoofdstuk "aanpak" laat zien hoe het project wordt uitgevoerd. Er wordt stilgestaan bij de ontwikkelstrategie die tijdens het project wordt toegepast. Er zijn verschillende strategieën mogelijk en elke strategie heeft weer verschillende methoden. Tijdens het ontwikkelproces worden werkzaamheden uitgevoerd. Dit

hoofdstuk laat aan de hand van de strategie zien welke werkzaamheden tijdens welke fasen worden uitgevoerd. Verder komen de schema's en tekentechnieken en de planning aan de orde. Bij de aanpak is gekozen om voor elk increment een plan te maken. Het is mogelijk om voor elk fase een plan te maken, maar voor dit project is dat onnodig, omdat er kleine fasen zijn zorgt een plan per fase voor te veel documentatie. Daarnaast zijn de fasen goed te overzien, zodat een faseplan overbodig is. Het doel van de aanpak is om overeenstemming te krijgen over de te volgen weg naar het eindresultaat dat beschreven is in het hoofdstuk 'projectopdracht'.

5.1.4 Projectinrichting

Het volgende hoofdstuk gaat over de inrichting van het project. Het is opgenomen in het plan van aanpak zodat duidelijk wordt hoe de informatievoorziening is, wat de faciliteiten voor de opdrachtnemer zijn en hoe de kwaliteit van het resultaat gewaarborgd wordt. Dit hoofdstuk voorkomt onduidelijkheden tijdens het project, omdat opdrachtgever en opdrachtnemer weten wat er van hen verwacht wordt.

5.1.5 Kosten en baten

Tijdens elk project zijn er kosten en baten. Deze kan de opdrachtgever afwegen en aan de hand daarvan beslissen of het project aangenomen wordt. Bij veel projecten bepaalt de opdrachtgever aan de hand van het plan van aanpak of de opdracht wordt aangenomen zoals beschreven in dat plan. Voordat dit project begon is geen plan van aanpak gemaakt, omdat dat een onderdeel is van het afstuderen. Wel hebben de opdrachtgever en opdrachtnemer bij aanvang al overeenstemming bereikt over de kosten en baten van het project. Deze zijn terug te vinden in het plan van aanpak.

5.1.6 Risicofactoren

Het laatste hoofdstuk gaat over de risicofactoren. Daarin wordt beschreven welke factoren het slagen van het project in de weg kunnen staan en welke maatregelen genomen worden om dat te voorkomen. Op die manier is de kans dat een tegenslag het slagen van het project in de weg staat kleiner, omdat er maatregelen zijn genomen om tegenslagen te voorkomen en op te lossen. Na dit hoofdstuk komen de bronnen en bijlagen aan de orde.

5.2 Definitiefase

De definitiefase is bedoeld om te bepalen waarom het project nuttig is. Tijdens deze fase worden er cursussen gedaan met betrekking tot het IBM Lotus Notes en Domino pakket. Ook wordt onderzocht of er plug-ins zijn die vergelijkbaar zijn met de te maken plug-in.

5.2.1 Onderzoek vergelijkbare plug-ins

In dit gedeelte staan de resultaten van het onderzoek naar vergelijkbare plug-ins. Dat onderzoek is nodig om te bepalen of de plug-in helemaal ontwikkeld moet worden. Het kan zijn dat er een plug-in bestaat die de wensen van de opdrachtgever vervult, of dat er een plug-in bestaat die als basis voor het project

gebruikt kan worden. Eerst worden enkele voorbeelden van bestaande plug-ins besproken, daarna is er een conclusie.

Bestaande plug-ins

Bestaande plug-ins voor de sidebar van Lotus Notes, de eerste zeven plug-ins staan beschreven op [1OpenNTF1002].

Chart Plugin

Deze plug-in geeft een weergave in een tabel van een geselecteerde view.

Document property viewer

In deze plug-in staan instellingen van het huidige geselecteerde document.

Formul8

Deze plug-in maakt het mogelijk om formules te schrijven en uit te voeren op het geselecteerde document.

Secret Agent

De Secret Agent maakt het mogelijk om snel informatie te verkrijgen van tussenpersonen/vertegenwoordigers.

SideLog

Geeft snel inzicht in logs over de domino server.

Recent Contacts

Deze plug-in geeft recente contacten weer. Er kan aangegeven worden welke contacten belangrijk zijn, die staan bovenaan in de plug-in.

TwitNotes

Met deze plug-in kunnen twitterberichten gelezen worden, de status worden veranderd en er kunnen openbare en persoonlijke berichten verstuurd worden.

File Navigator

De File Navigator laat het operating system file system zien. Vanaf de sidebar kunnen files worden geopend, gedeletet, gekopieerd en geplakt. Files en folders kunnen met behulp van drag and drop in Notes document worden gezet. Meer informatie over deze plug-in is te vinden op [2Openntf1002].

Sidebar plug-in voor lotus enterprice integrator

Deze plug-in maakt het mogelijk om Lotus Enterprice Integrator te monitoren. Meer informatie over deze plug-in is te vinden op [3Openntf1002].

Lotus Quickr

Quickr is een applicatie, er is ook een plug-in van Quickr voor de sidebar van Lotus Notes. Met drag and drop kunnen files in de sidebar gezet worden, deze kunnen dan gedeeld worden met collega's of verwijderd worden uit de inbox. Ook kan de plug-in een attachment in een library zetten en het attachment vervangen voor een link. Meer informatie over deze plug-in is te vinden op [Lotus1002].

Side Note

Deze plug-in zorgt voor een notepad in de sidebar. Meer informatie over deze plug-in is te vinden op [4Openntf1002].

LinkedIn plug-in

De plug-in van LinkedIn is vergelijkbaar met de plug-in van twitter, het verschil is natuurlijk dat het bij deze plug-in om LinkedIn gaat. Meer informatie over deze plug-in is te vinden op [Zwart1002].

TripIt

Hier zijn aanbiedingen van TripIt te zien en informatie over reizen die gemaakt kunnen worden. Meer informatie over deze plug-in is te vinden op [Zwart1002].

Conclusie

Het onderzoek heeft veel plug-ins voor de sidebar van Lotus Notes opgeleverd. Er is een verschil te zien tussen plug-ins die samenwerken met modules in IBM Lotus Notes en Domino en plug-ins die samenwerken met andere platformen of stand-alone werken.

Tussen de plug-in die gemaakt moet worden voor het afstuderen en de onderzochte plug-ins zit een groot verschil. Het verschil is dat de plug-in van de opdracht samen moet werken met alle modules in CSP. De onderzochte modules werken allemaal niet met CSP en over het algemeen met hooguit één gegevensbron uit Lotus Notes Lotus Domino of een ander platform. De plug-in 'Document property viewer' is een uitzondering die eigenschappen van het huidig geselecteerde document laat zien. Deze plug-in heeft mogelijk enkele functies die nuttig zijn voor het project.

Een plug-in voor de Lotus Notes sidebar moet ook geïnstalleerd kunnen worden. De meeste onderzochte plug-ins hebben een website online staan waar een XML bestand op staat. Wanneer het XML bestand in de sidebar gesleept wordt begint de installatie. Voor de te maken plug-in is deze manier van installeren niet toe te passen, omdat de plug-in vanuit CSP geïnstalleerd gaat worden.

5.2.2 Cursus Lotus Notes en Domino en CSP

Tijdens de definitiefase heeft de opdrachtnemer kennisgemaakt met de omgeving waarin het project zich afspeelt. Er zijn twee cursussen gedaan met betrekking tot Lotus Notes en een cursus over CSP. De cursussen van Lotus Notes zijn van The Learning Continuum Company (TLCC) en te vinden op [TLCC1102], die van CSP heeft AedifiComm gemaakt. In verband met vertrouwelijkheid staat die niet in dit verslag.

Lotus Notes

Lotus Notes is de client kant van een client-server, groupware applicatie die ontwikkeld is door IBM Software Group. De client heeft de volgende functies:

1. (Zakelijke) email
2. Adresboek(en)
3. Agenda

4. IBM Lotus Domino server toepassingen (applicaties)
Standaard applicaties van IBM en mogelijkheden om zelf applicaties te ontwikkelen

Databases in Lotus Notes zijn niet relationeel, het is wel mogelijk om relationele databases aan toepassingen van Lotus Notes te koppelen.

IBM Lotus Notes en Domino is volledig opgebouwd uit notes. Notes zijn documenten. Een verzameling notes is een database en een note bevat items. Binnen Lotus Notes wordt onderscheid gemaakt tussen verschillende notes:

1. Design-notes
In deze notes staat een interface beschreven
2. Data-notes
Hierin staat de data beschreven.
3. Administration-notes
In de administration-notes worden rechten van gebruikers beschreven

Design-notes en data-notes zijn vaak gekoppeld aan elkaar. Het is mogelijk om data-notes weer te geven in een view. Een view is eigenlijk een soort datagrid en bestaat uit columns waarin item-waarden getoond worden, met mogelijkheden voor sortering en categorisering. De view zelf is gedefinieerd in een design-note.

CSP

De cursus over CSP gaat over de inhoud van verschillende modules. Elke module bevat veel informatie en er kan op verschillende manieren met deze informatie om worden gegaan. Een gedetailleerde beschrijving van elke module heeft geen toegevoegde waarde voor dit project.

Voor increment één is het goed om te weten dat onder de module Post de activiteiten per medewerker staan.

5.3 Analyse

In de analysefase staat centraal wat het resultaat van het project wordt. Dat is goed duidelijk te maken door eisen. Zo ook in dit project, daarom wordt eerst besproken hoe eisen te achterhalen zijn, vervolgens wordt het programma van eisen versie 1.0 besproken dat in bijlage 5 staat. Na het verzamelen van eisen is een incrementplan gemaakt voor increment één, dat is te vinden in bijlage 4.

5.3.1 Eisen achterhalen

Er zijn verschillende manieren om eisen te achterhalen. Er kunnen interviews gehouden worden met gebruikers en met de opdrachtgever, verder kan de ontwikkelaar zelf eisen voorleggen aan de opdrachtgever en gebruiker. Daarnaast kan er een prototype gemaakt worden, waardoor de gebruiker, opdrachtgever en de ontwikkelaar een beter beeld krijgen van het resultaat. Aan de hand van een prototype kunnen nieuwe eisen ontdekt of oude eisen aangepast worden. Tijdens dit project spelen de uiteindelijke gebruikers geen rol. De reden daarvoor is dat de oplossing deel gaat uitmaken van een standaardpakket. AedifiComm heeft een beschrijving van de gewenste oplossing aan verschillende klanten voorgelegd.

Iedere klant kijkt vanuit zijn eigen dagelijkse werkzaamheden naar de plug-in, maar houdt geen rekening met een generiek en herbruikbaar karakter dat voor het standaardpakket van belang is. AedifiComm kan zich verplaatsen in de werkzaamheden van de klant en daarmee de wensen ten aanzien van de plug-in formuleren.

Er zijn verschillende interviews gehouden met de opdrachtgever over het product. Er is voor gekozen om geen vragenlijsten uit te denken voor een interview met de opdrachtgever. Daar is voor gekozen, omdat de opdrachtgever nog niet duidelijk voor ogen had wat de plug-in moet gaan doen. Tijdens de interviews werd gebrainstormd over de functie van de plug-in en de eisen die daarbij horen. Een tweede argument om geen vragenlijsten te maken is dat de opdrachtgever op dezelfde afdeling werkt als de ontwikkelaar. Daardoor kunnen vragen meteen gesteld worden.

Na increment één wordt een prototype opgeleverd. Het prototype is bedoeld om de opdrachtgever een beter beeld te geven van het product. Aan de hand van het prototype wordt duidelijk of de opdrachtgever en opdrachtnemer hetzelfde beeld hebben over hoe het eindproduct eruit moet komen te zien. Door een prototype worden er vaak eisen aangepast of toegevoegd, daarom wordt de analysefase van increment twee gebruikt om eisen die bekend worden door het prototype vast te leggen.

5.3.2 Programma van Eisen (PvE)

In het PvE staat beschreven wat het uiteindelijke product kan. De eisen worden duidelijk op drie manieren, namelijk use-case-diagrammen, scenario's en uitgeschreven eisen.

De use-case-diagrammen zijn bedoeld om de functionele systeemeisen weer te geven. De gebruiker en opdrachtgever kunnen aan use-case-diagrammen zien wat de gebruiker van het systeem kan verlangen. Diagrammen spreken meer tot de verbeelding dan alleen tekst, daarom staan naast de uitgeschreven eisen ook use-case-diagrammen in het PvE.

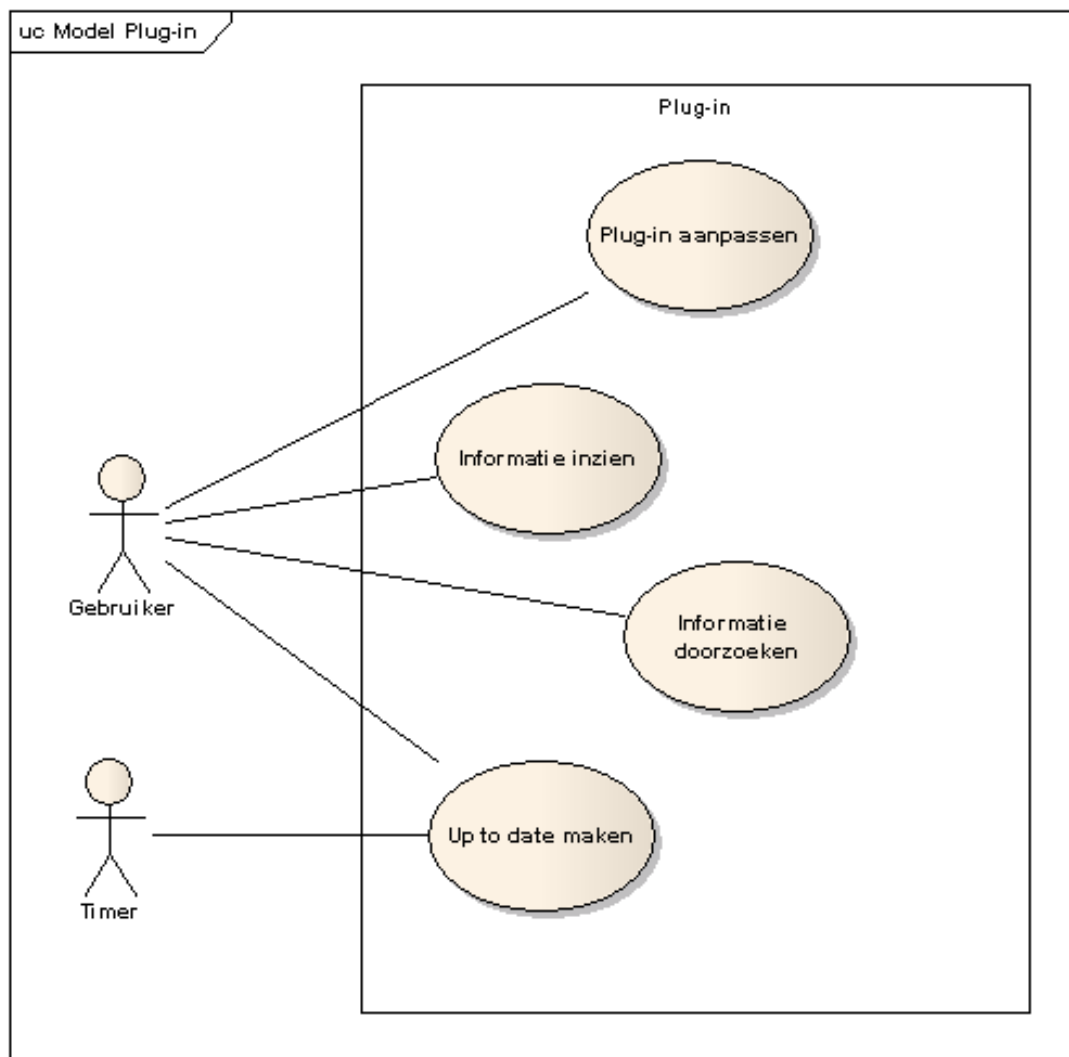
In eerste instantie was besloten geen scenario's in het PvE op te nemen. Maar door onduidelijkheid over de relaties tussen use-cases is toch gekozen om scenario's op te nemen in het PvE. De scenario's laten zien in welke volgorde de gebruiker acties kan uitvoeren op de plug-in en wat dat voor invloed heeft op de objecten van de plug-in.

Als laatste worden de eisen beschreven. Hierbij is gebruik gemaakt van [Dijkgraaf07].

De afkomst van de eisen is genoteerd. Dat is gedaan, zodat altijd duidelijk is wie, wanneer, welke eis heeft vermeld. Bij onduidelijkheid kan daar later op teruggegrepen worden. Het noteren van deze gegevens staat niet in het gebruikte boek.

Use-case-diagram

Om duidelijk te maken wat de globale eisen van de plug-in zijn is een use-case-diagram gemaakt, zie afbeelding 5.1. Het doel van de plug-in is relevante informatie altijd zichtbaar maken voor de gebruiker.



Afbeelding 5.1: global use-case-diagram van de plug-in

De gebruiker kan vier dingen doen met de plug-in. Allereerst kan de plug-in worden aangepast. Daarbij kan gedacht worden aan installeren en deïnstalleren, maar ook minimaliseren en maximaliseren.

Naast het aanpassen van de plug-in moet de gebruiker ook informatie kunnen inzien in de plug-in. Dat kan de gebruiker doen door een gegevensbron te selecteren. Als de gebruiker een gegevensbron heeft geselecteerd kan de informatie daarvan doorzocht worden met een filter of een zoekfunctie.

Als laatste kan de gebruiker de plug-in up to date maken door de refreshfunctie aan te roepen. Ook heeft de plug-in een timer die de plug-in regelmatig refreshed.

Met deze informatie is een globaal beeld beschreven van de plug-in. Het generieke aspect van de plug-in en hoe de plug-in geïnstalleerd en gedeïnstalleerd kan worden is nog niet duidelijk door het use-case-diagram. Daarom is naast de use-case-diagrammen ook een lijst van eisen opgesteld. In die lijst wordt duidelijk dat het generieke aspect van de plug-in meetbaar is, zie tabel 5.1. Wat plug-in enabled

betekent is nog niet te zeggen tijdens de analyse van increment één. Eerst moet duidelijk worden wat de kenmerken zijn van de modules. Met behulp van die kenmerken wordt de plug-in generiek gemaakt, dat zorgt ervoor dat modules met die kenmerken plug-in enabled zijn en modules zonder die kenmerken niet.

| Generiek | | | | |
|----------|--|---|----|----|
| | De plug-in werkt met alle modules in CSP die plug-in enabled zijn. | M | I2 | 26 |
| | De plug-in werkt in nog te maken modules van CSP die plug-in enabled zijn. | S | I2 | 27 |

Tabel 5.1: eisen over generieke aspect van de plug-in

Het installeren en deïnstalleren is ook meetbaar gemaakt door eisen zoals te zien is in tabel 5.1. Hoe de plug-in precies geïnstalleerd en gedeïnstalleerd wordt is nog niet duidelijk, dat wordt onderzocht tijdens de ontwerpfase.

| Performance | | | | |
|-------------|---|---|----|----|
| | De plug-in geeft informatie snel weer. <i>Snel = sneller dan huidige navigatie naar informatie</i> | M | I1 | 15 |
| | De plug-in is snel geïnstalleerd <i>Snel = korter dan 3 minuten</i> | M | I1 | 16 |
| | De plug-in is snel gedeïnstalleerd <i>Snel = korter dan 3 minuten</i> | M | I1 | 17 |

Tabel 5.2: eisen over installatie en deïnstallatie

5.3.3 Incrementplan

Tijdens de analysefase is een incrementplan gemaakt. In dat plan staat wat de doelen en werkzaamheden van increment één zijn. In het plan van aanpak is beknopt aangegeven dat tijdens het eerste increment de nadruk ligt op het onderzoeken van de platformen en het installeren en deïnstalleren van de plug-in. Daarnaast moet de plug-in werken met één gegevensbron, namelijk die van "persoonlijke activiteiten". Die gegevensbron is gekozen, omdat die een to-do list van de gebruiker weergeeft. Voor veel gebruikers is dat een handige gegevensbron om altijd zichtbaar te hebben.

De lay-out van de plug-in met tijdens het eerste increment aansluiten op Lotus Notes en CSP. Er is voor gekozen om dat meteen bij aanvang van de ontwikkeling van de plug-in te doen, zodat in een later stadium van het project geen andere lay-out ontwikkeld moet worden.

Na increment één is duidelijk met welke platformen de plug-in ontwikkeld wordt. Ook kan de plug-in geïnstalleerd en gedeïnstalleerd worden. Verder kan in CSP de gegevensbron "persoonlijke activiteiten" weergegeven worden en sluit de lay-out van de plug-in aan bij Lotus Notes en CSP.

Naast het prototype worden in het eerste increment ook documenten opgeleverd. Het eerste document is het plan van aanpak. Dit document is nodig om de opdracht goed in beeld te brengen. Ook staat in het plan van aanpak hoe het project

aangepakt wordt. Opdrachtgever, opdrachtnemer en in dit geval ook de begeleider van de afstudeerder hebben dan een goed beeld van de opdracht en de aanpak daarvan. Als bijlage van het plan van aanpak wordt een gedetailleerde planning opgeleverd.

Naast het plan van aanpak wordt er een programma van eisen gemaakt. In het programma van eisen worden ook de eisen die bekend zijn voor iteratie twee al meegenomen. Daar is voor gekozen, omdat anders moeilijk te overzien is hoe groot het project is. Ook kunnen de doelen van de incrementen niet bepaald worden als niet alle eisen bekend zijn. Het document over de eisen gaat dieper dan het plan van aanpak in op de eisen van de plug-in. In de analysefase van increment twee wordt het programma van eisen aangepast en uitgebreid met behulp van het gemaakt prototype.

Tijdens de ontwerpfase wordt een ontwerp gemaakt. Het ontwerp beschrijft de benodigde software om aan de eisen van increment één te voldoen. In de modelleerfase van increment twee wordt het ontwerp uitgebreid.

AedifiComm heeft weinig kennis over het ontwikkelen van een plug-in voor Lotus Notes, daarom is het van belang dat tijdens het implementeren goed gedocumenteerd wordt hoe alles geïmplementeerd wordt. Dat zorgt ervoor dat vergaarde kennis niet verloren gaat.

Aan het einde van increment één wordt de plug-in getest. De uitkomsten van de testen komen in een testrapport, deze bevindingen worden in increment twee gebruikt om de plug-in te verbeteren.

Tijdens het afstuderen wordt ook een tussenverslag gemaakt. Dit tussenverslag wordt tijdens increment één opgeleverd.

5.4 Ontwerp

Tijdens deze fase is een ontwerp gemaakt voor de te maken plug-in. Dat ontwerp staat beschreven in bijlage 6. In het ontwerp wordt beschreven hoe de plug-in gemaakt gaat worden. Zo denkt de ontwikkelaar na over de opbouw van de plug-in voordat deze geïmplementeerd wordt. Daarnaast staat op papier op welke manier de plug-in in elkaar zit, waardoor dat ook voor buitenstaanders snel duidelijk is. Het ontwerp bestaat uit platformen die gebruikt worden en diagrammen die de plug-in beschrijven. Daarnaast laat het ontwerp zien hoe de punten worden aangepakt die eerder dit verslag genoemd zijn in paragraaf 3.5.

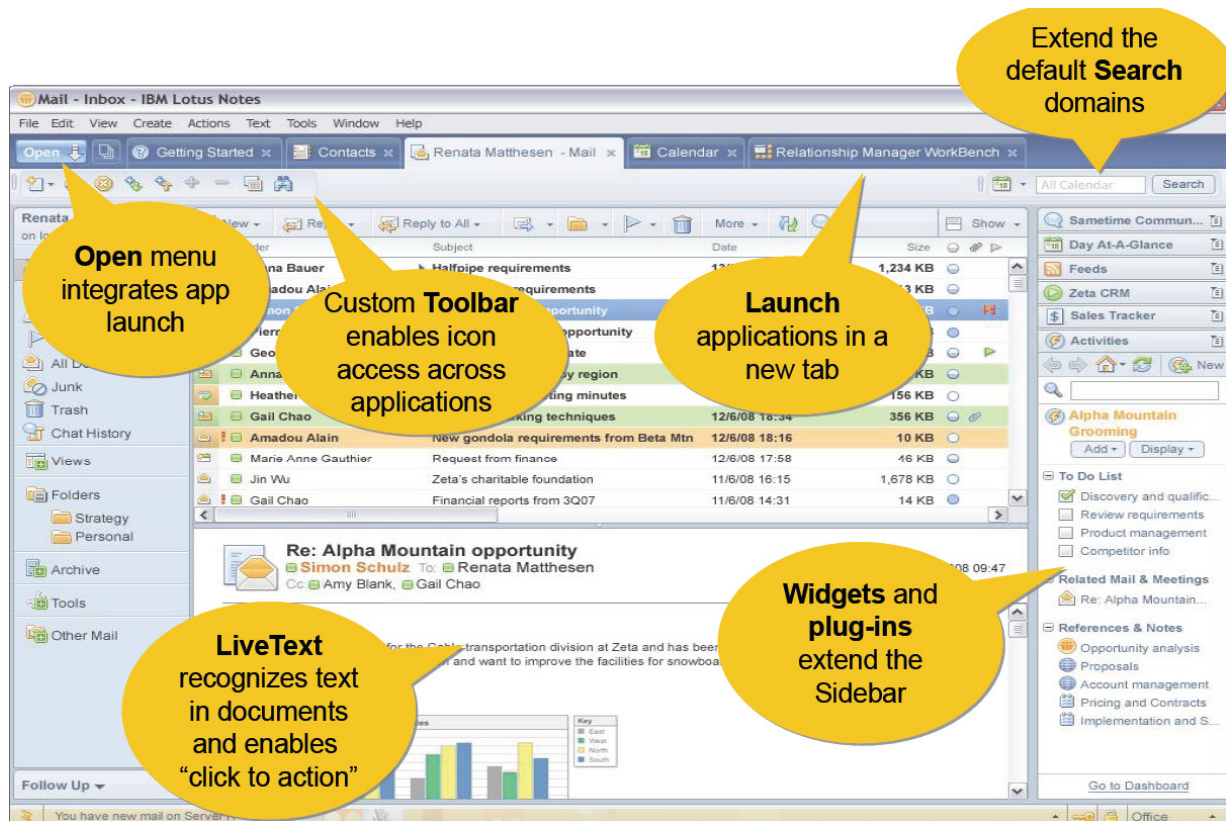
5.4.1 Platformen

Voordat begonnen werd met het ontwerpen van de plug-in is bepaald welke platformen en softwarepakketten een rol spelen bij de ontwikkeling van de plug-in. Allereerst wordt gewerkt met CSP, want daar wordt de plug-in voor ontwikkeld. Een screenshot van CSP is te zien in afbeelding 5.2. CSP is software die ondersteuning biedt voor de werkprocessen van accountants en belastingadviseurs. Het is gebaseerd op IBM Lotus Notes. AedifiComm heeft CSP in eigen beheer ontwikkeld en verzorgt zelf implementatie bij klanten. AedifiComm is al sinds de oprichting een IBM business partner die zich bezig houdt met software ontwikkeling op het IBM Lotus Notes platform, dus daarover was veel kennis aanwezig.

Afbeelding 5.2: screenshot van CSP

Verder is het ontwikkelen op Lotus Notes relatief gemakkelijk. Het ontwikkelen op bijvoorbeeld Microsoft Sharepoint kost meer tijd dan het ontwikkelen op Lotus Notes. Daar komt nog bij dat in de tijd dat AedifiComm met CSP begon niet veel platformen aanwezig waren, dus de keus was minder.

Nog een kenmerk van Lotus Notes is dat het is opgebouwd uit notes (documenten). De doelgroep van CSP (accountants en belastingadviseurs) werkt veel met documenten, daar sluit Lotus Notes goed bij aan. Verder is het platform veilig en wordt het ontwikkeld door een betrouwbaar en groot bedrijf, IBM. Een screenshot van Lotus Notes is te zien op afbeelding 5.3. Wat opvalt aan de afbeeldingen 5.2 en 5.3 is de sidebar aan de rechterkant. Dat is de sidebar van Lotus Notes en dus ook aanwezig in CSP. De te maken plug-in komt in die sidebar.



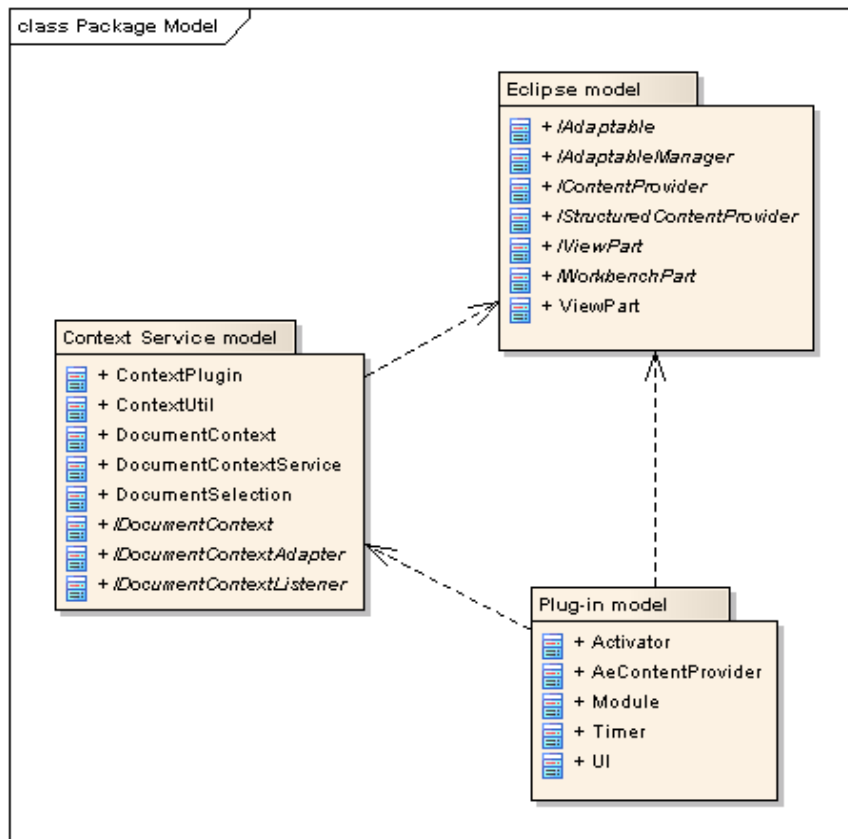
Afbeelding 5.3: screenshot van Lotus Notes

CSP, gebaseerd op Lotus Notes is de toepassing waar de plug-in voor gemaakt wordt. Verder moet bepaald worden in welke programma de plug-in ontwikkeld wordt. Dat kan op verschillende manieren. IBM heeft voor Lotus Notes een omgeving gemaakt waarin software ontwikkeld kan worden voor Lotus Notes, namelijk Domino Designer. Voor de plug-in van dit project is dit geen nuttige omgeving, omdat het daarin niet mogelijk is om een plug-in voor de sidebar van Lotus Notes te ontwikkelen. Een ander platform waarin ontwikkeld kan worden voor Lotus Notes is Eclipse. Sinds versie 8 van Lotus Notes is het gebaseerd op Eclipse. In Eclipse kan de IBM Lotus Expeditor geïnstalleerd worden, dat is de brug tussen Eclipse en Lotus Notes. Dat maakt het mogelijk om een plug-in te ontwikkelen voor de sidebar van Lotus Notes. In bijlage 14, 15 en 16 staat hoe in Eclipse een plug-in voor Lotus Notes gemaakt kan worden.

5.4.2 Packagediagram

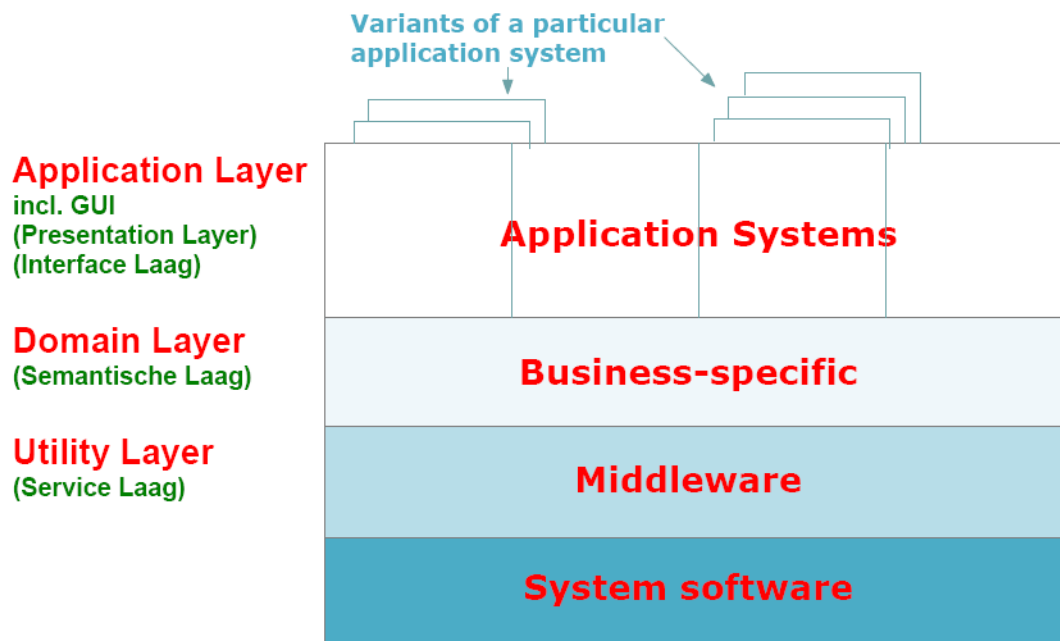
In de ontwerpfase zijn diagrammen gemaakt voor de plug-in, namelijk een klassediagram en een packagediagram. Het packagediagram is te zien op afbeelding 5.4. Er is voor gekozen om niet alleen de plug-in te modelleren, maar ook gedeelten van Eclipse en Context Service, omdat de plug-in daar gebruik van maakt. Context Service is software die gegevens van het huidige geselecteerde document ophaalt als de gebruiker een document selecteert. Het packagediagram laat zien hoe de afhankelijkheden tussen de packages zijn. Daardoor zijn verschillende lagen in het

model te herkennen. De plug-in is de interfacelaag en het minst stabiel. De Context Service is abstracter en stabiel dan de plug-in en valt onder de semantische laag. De meest abstracte package is Eclipse, deze bevat veel interfaces die geïmplementeerd worden in de twee andere packages van het model. Eclipse valt onder de semantische laag en de service laag, omdat het componenten biedt die ontwikkelaars in applicaties in kunnen vullen. In afbeelding 5.5 is de lagenstructuur te zien.



Afbeelding 5.4: packagediagram

Het doel van deze lagen is het hergebruik van software. De package Eclipse is abstract en stabiel, hij verandert zelden. Omdat Eclipse zelden wijzigt is het mogelijk om gebruik te maken van klassen in Eclipse. De plug-in verandert meer en is niet abstract en niet stabiel. Als de plug-in wijzigt moeten packages die van de plug-in afhankelijk zijn ook veranderen. Het is daarom niet handig om software te maken die afhankelijk is van de plug-in, want die verandert vaak. In het model is duidelijk te zien dat er geen package is die afhankelijk is van de plug-in en wel van Eclipse.



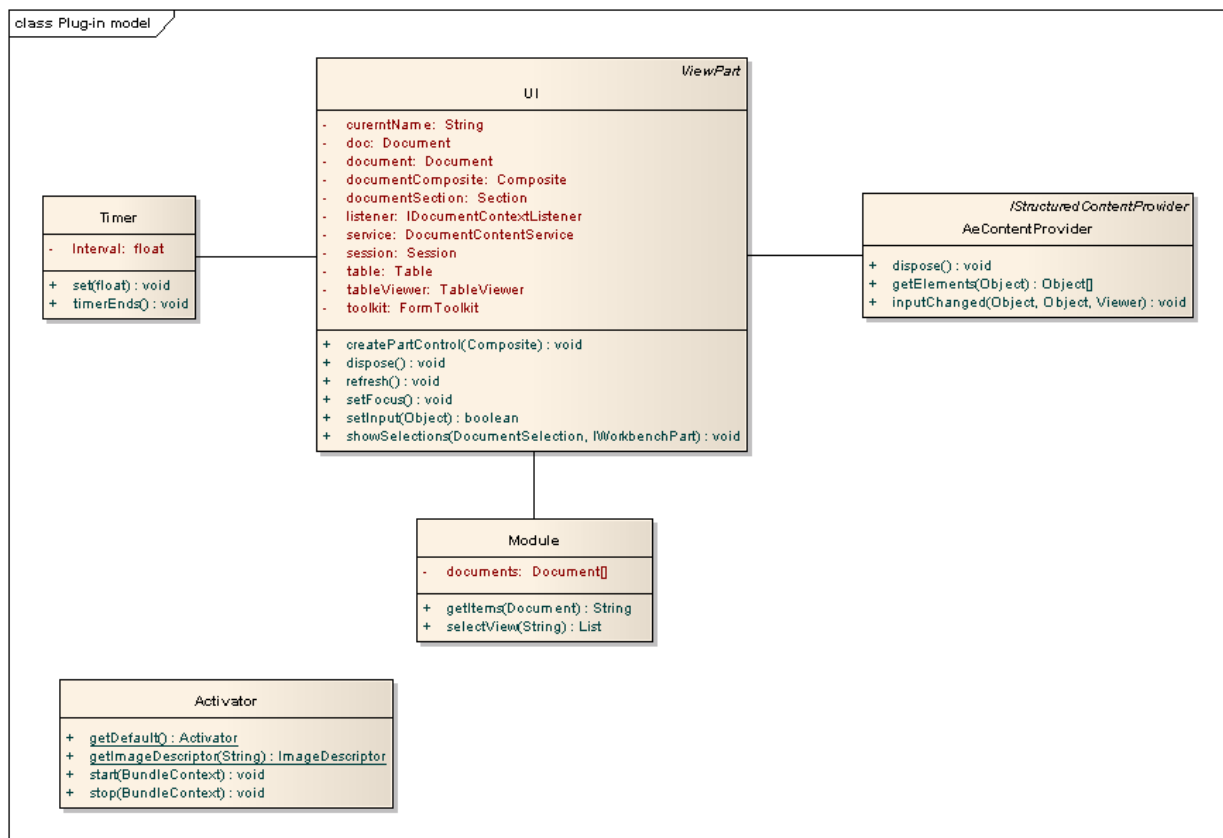
Afbeelding 5.5: lagenstructuur

5.4.3 Klassediagram

Het klassediagram van de plug-in voor increment één is te zien op afbeelding 5.6. Het diagram is gemaakt aan de hand van de eisen die staan in het incrementplan van increment één.

Wat opvalt aan het diagram is dat er niets over het installeren en het deïnstalleren staat. Dat komt omdat de plug-in dat niet zelf doet. In Eclipse is het mogelijk om van de gemaakte plug-in een updatesite te maken. Dan is een XML nodig die naar die updatesite wijst en als die XML in de sidebar van Lotus Notes geplaatst wordt installeert Lotus Notes de plug-in waar naar gewezen wordt en plaatst deze in de sidebar. Hoe dat precies werkt staat beschreven in bijlage 6, hoofdstuk 'Installatie'.

Als de plug-in is geïnstalleerd moet de interface worden opgebouwd, dat gebeurt in de klasse UI. Een belangrijk attribuut in UI is de `IDocumentContextListener`, uit de package Context Service. Die listener haalt gegevens op van het document dat de gebruiker op dat moment heeft geselecteerd in CSP en zet deze in een object van de klasse Module. Nu is de plug-in opgebouwd en krijgt het de gegevens van het geselecteerde document. De volgende stap is het weergeven van gegevens van het geselecteerde document. Dat gebeurt door een implementatie van `IStructuredContentProvider`, namelijk `AeContentProvider`.



Afbeelding 5.6: klassediagram van de plug-in

Een probleem tijdens de ontwerpfase was de beperkte kennis over Eclipse en Lotus Notes. Daardoor overlaptten de ontwerpfase en de implementatiefase elkaar voor een gedeelte, meer daarover staat in paragraaf 5.5.

5.5 Implementatie

Tijdens de implementatie is met behulp van het ontwerp de plug-in geïmplementeerd. Zoals in de vorige paragraaf al is vermeld overlaptten de ontwerpfase en de implementatiefase elkaar. Tijdens het ontwerpen werd duidelijk dat veel klassen van Eclipse goed gebruikt konden worden voor de plug-in. Voordat gemodelleerd kon worden hoe die klassen gebruikt gingen worden moest eerst de werking van die klassen onderzocht worden. Daarom werd al begonnen met het maken van een plug-in voor Lotus Notes. Door bevindingen tijdens het implementeren kon het model aangepast worden.

Aan het einde van de implementatiefase is het eerste prototype opgeleverd. De plug-in kan geïnstalleerd worden in de sidebar van Lotus Notes. Als de plug-in is geïnstalleerd wordt daarin een tabel weergegeven. Als de gebruiker een document selecteert laat de plug-in de kolommen van dat document zien, zonder inhoud.

5.5.1 Opbouw van plug-in

De weergave van de plug-in wordt gemaakt door de functie `createPartControl` in de klasse `UI`. Die functie komt uit `IViewPart`, hier is dus function overloading toegepast, zoals te zien is op afbeelding 5.7.

In `createPartControl` wordt allereerst een `FormToolkit` gemaakt. De `FormToolkit` maakt het mogelijk om in de plug-in te werken met veel functies uit de Standard Widget Toolkit (SWT). Daardoor kan binnen de `FormToolkit` onder andere secties, composites en tables gecreëerd worden. Hoe een `FormToolkit` en een sectie gecreëerd worden is te zien in code 5.1.

```
private Composite documentComposite;
private FormToolkit toolkit;

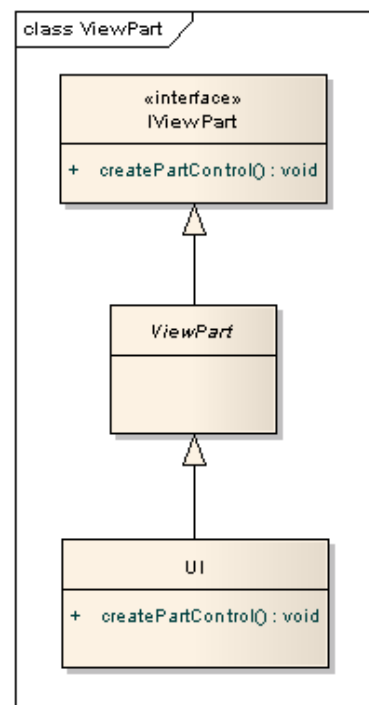
public void createPartControl(Composite parent) {
    toolkit = new FormToolkit(parent.getDisplay());

    documentSection = toolkit.createSection(parent, Section.TITLE_BAR);
    documentSection.setText("Plug-in");
    documentSection.marginWidth = 4;
    documentSection.marginHeight = 3;
    ...
}
```

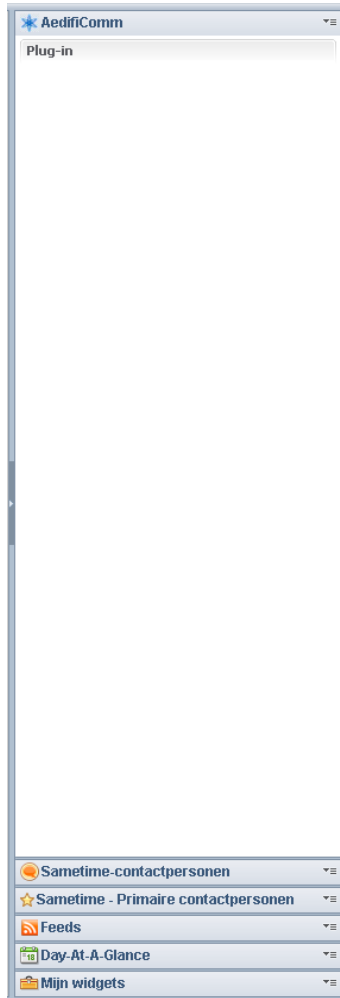
Code 5.1: creëren van `FormToolkit` en sectie

Een sectie is een gedeelte in de plug-in. Met verschillende secties kan onderscheid gemaakt worden tussen verschillende gedeeltes in een plug-in. Daarnaast ziet de plug-in er met een sectie beter uit, zie afbeelding 5.8 en 5.9.

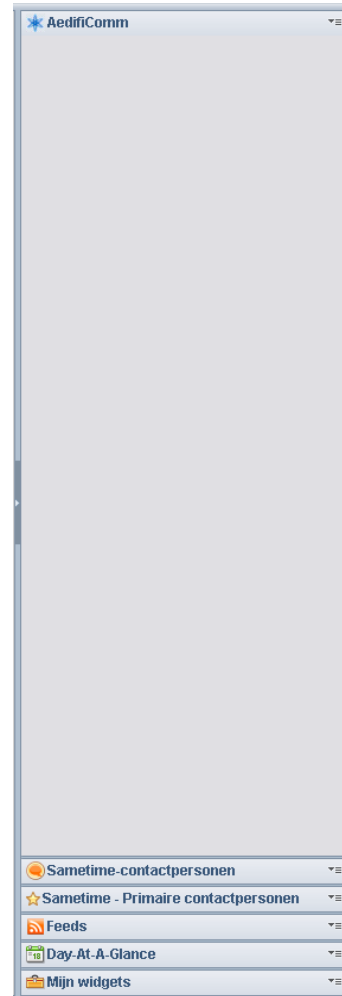
Tijdens het opbouwen van de plug-in wordt een listener gemaakt. De listener zorgt ervoor dat de `selectionChanged` functie wordt aangeroepen als de gebruiker van document wisselt. Hoe de listener aangemaakt en toegekend wordt staat in code 5.2, evenals de functie `selectionChanged`.



Afbeelding 5.7: function overloading van `createPartControl`



Afbeelding 5.8: plug-in met sectie



Afbeelding 5.9: plug-in zonder sectie

```
import com.ibm.lotuslabs.context.service.document.DocumentSelection;
import com.ibm.lotuslabs.context.service.document.IDocumentContextListener;
private IDocumentContextListener listener;
private DocumentContextService service;

public void createPartControl(Composite parent) {
    ...
    listener = new IDocumentContextListener(){
        public void selectionChanged(IWorkbenchPart part,
            DocumentSelection selection){
            showSelection(part, selection);
        }
    };

    service = DocumentContextService.getDefault();
    service.addSelectionListener(listener);
    ...
}
```

Code 5.2: aanmaken en toekennen van de IDocumentContextListener

5.5.2 Verandering van selectie

Als de plug-in opgebouwd is, gebeurt er niets, tot de gebruiker van document wisselt. Dan wordt de functie `selectionChanged` aangeroepen die in code 5.2 te zien is. Daarbij wordt de `IWorkbenchPart` meegegeven, dat is het gedeelte van CSP dat op dat moment geselecteerd is. Die is nodig om te checken wat de gebruiker selecteert, want de plug-in moet niet veranderen als de gebruiker de plug-in zelf selecteert. Ook wordt de `DocumentSelection` meegegeven, dat zijn gegevens over het document dat op dat moment geselecteerd is.

De plug-in gaat iets doen als de gebruiker van document wisselt, omdat op dat moment de selectie en dus ook de url verandert. Het is mogelijk om een functie aan te roepen als de selectie verandert. Het zou makkelijker zijn als het mogelijk is om een functie aan te roepen als de weergave verandert, want dan zou de plug-in alleen iets hoeven doen als de gebruiker van weergave verandert.

Met de `DocumentSelection` wordt bepaald wat het huidige geselecteerde document is. Met dat document kunnen alle gegevens die nodig zijn opgehaald en gebruikt worden. In Lotus Notes wordt onderscheid gemaakt tussen documenten en views. Met de huidige geselecteerde view is de informatie die de gebruiker op het scherm ziet op te halen, met het huidige geselecteerde document niet.

Om de view op te halen is gebruik gemaakt van de URL van de huidige selectie. De URL is altijd hetzelfde opgebouwd en bestaat uit de ID's van de server, de database, de view en het document. Via het document wordt het ID van de database opgehaald. Nu kan de ID van de view worden opgehaald, want bekend is dat deze altijd achter de ID van de database staat, zie code 5.3.

```
public String getViewKeyDB(String uri){
    String substring = uri;
    for (int i = 0; i<4; i++){
        int index1 = substring.indexOf("/");
        substring = substring.substring(index1+1);
    }

    int index2 = substring.indexOf("/");
    if (index2 == -1){
        index2 = substring.indexOf("?");
    }
    String substring2= substring.substring(0, index2);

    return substring2;
}
```

Code 5.3: bepalen van het ID van de view

Het is niet mogelijk om met het document de view op te halen, want een document kan in meerdere views tegelijk staan.

Uiteindelijk worden de kolommen uit de huidige view gehaald en geplaatst in de plug-in.

5.6 Testen

De testfase volgde op de implementatiefase. In deze fase werden een testplan en een testrapport gemaakt, deze zijn te vinden in bijlage 7 en bijlage 8. Het testplan is gemaakt aan de hand van de eisen die in het incrementplan van increment één zijn vastgelegd. De testen zijn uitgevoerd door de ontwikkelaar.

5.6.1 Testplan

Er is getest op drie verschillende operating systems, namelijk Windows XP, Windows Vista en Windows 7. Dat is gedaan, omdat in de eisen staat dat de plug-in op die operating systems moet werken.

Het testplan is zo opgebouwd dat de eisen die in het incrementplan van increment één staan getest worden. De testen zijn uitgevoerd door de ontwikkelaar zelf. De voornaamste reden om het niet te laten testen door een buitenstaander is dat het minder tijd kost.

Hierdoor is het wel noodzakelijk om de plug-in grondig door een buitenstaander te laten testen voordat deze in gebruik genomen wordt.

5.6.2 Testrapport

Het testrapport beschrijft de uitkomsten van de testen die in het testplan beschreven staan. Na het testen werd duidelijk dat de plug-in goed te installeren en te deïnstalleren is. Verder sluit het uiterlijk goed aan bij Lotus Notes en CSP en is de plug-in in de Nederlandse taal.

De eisen waar een negatief resultaat uit kwam waren het meerdere keren installeren, het weergeven van informatie en het vernieuwen van de plug-in. Het meerdere keren installeren is niet zo gemakkelijk als gedacht, daarom is ervoor gekozen die functie nog niet te implementeren. De weergave van informatie en het vernieuwen van de plug-in is ook nog niet geïmplementeerd vanwege gebrek aan tijd. Wel is tijdens increment één tijd bespaard voor increment twee, omdat de plug-in een generieke basis heeft. Op die manier kunnen de ontbrekende functies tijdens increment twee worden geïmplementeerd.

6 Increment 2

In dit hoofdstuk wordt het tweede increment van dit project beschreven. De fasen die doorlopen zijn worden afzonderlijk van elkaar besproken. De definitiefase die wel in increment één zit is niet doorlopen in increment twee. In deze fase gaat het om de reden en haalbaarheid van het project en die zijn niet veranderd sinds de definitiefase van increment één.

6.1 Analyse

In de analysefase wordt duidelijk wat het resultaat van het project wordt. In increment één is een begin gemaakt met de eisen. In deze paragraaf komt versie 2.2, zie bijlage 10, van het programma van eisen aan bod dat tijdens increment twee gemaakt is. Na de analyse is er een incrementplan gemaakt voor increment twee, dat is te vinden in bijlage 9.

6.1.1 Eisen achterhalen

Na increment één werd begonnen met de analysefase van increment twee. De manieren om eisen te achterhalen waren vaak hetzelfde als tijdens increment één. Dat betekent dat er interviews waren met de opdrachtgever waardoor eisen werden achterhaald.

Er is wel een verschil ten opzichte van de analysefase van increment één, namelijk dat er een prototype van de plug-in gemaakt is. Door het prototype is te zien of de ontwikkeling van de plug-in op de goede weg is. Als een ontwikkelaar eisen anders heeft geïnterpreteerd dan de opdrachtgever wordt dat vaak bij het prototype duidelijk. In dit project zijn verschillende eisen aangepast en toegevoegd. Het prototype liet verschillende functies zien die niet in het programma van eisen stonden, maar waar de opdrachtgever wel enthousiast over was. Alle aanpassingen en toevoegingen zijn te vinden in het programma van eisen.

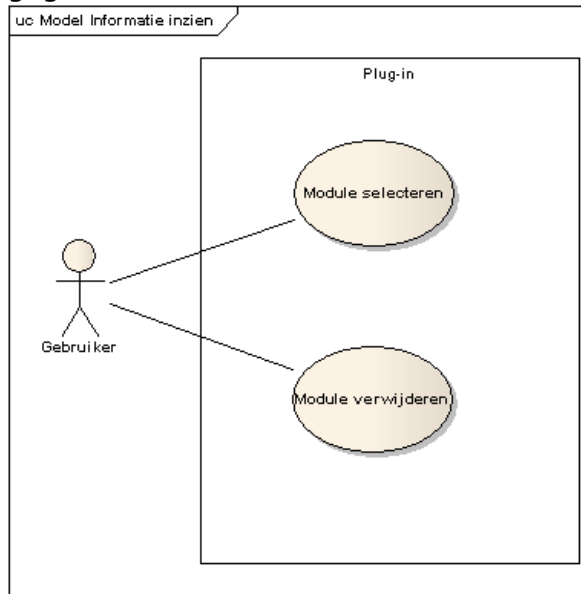
Daarnaast heeft de ontwikkelaar tijdens increment één meer inzicht gekregen in de mogelijkheden van de techniek. Het kan bijvoorbeeld zijn dat eisen onhaalbaar zijn, omdat de techniek het niet toelaat. Tijdens dit project was dat het geval bij de eis over het meerdere keren installeren van de plug-in, die was onhaalbaar in de beschikbare tijd.

6.1.2 Programma van eisen

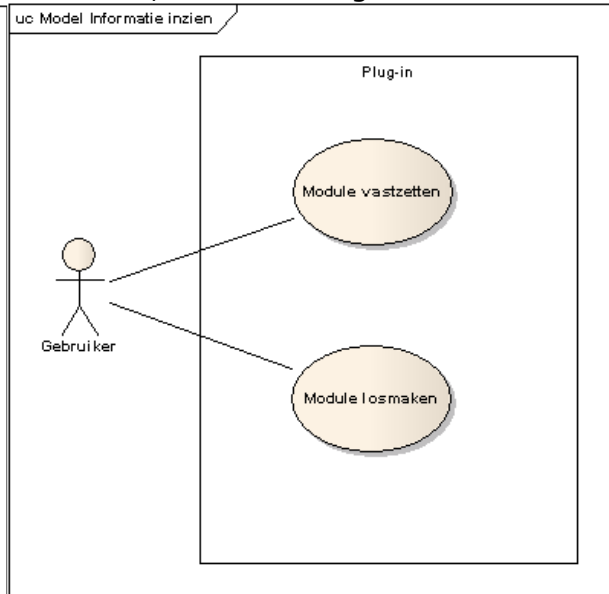
Na de analysefase van increment twee is het PvE versie 2.2 opgeleverd. De opbouw van dat document komt op veel punten overeen met versie 1.0. De use-case-diagrammen en de uitgeschreven eisen staan in beide versies. In versie 2.2 zijn de scenario's weggelaten. De reden daarvoor is dat deze geen bijdrage hadden aan het duidelijk maken van de eisen. Scenario's zijn bedoeld om duidelijk te maken in welke volgorde functies aangeroepen worden. De plug-in heeft geen functies in de user interface die in een bepaalde volgorde moeten worden aangeroepen. De user interface bevat wel functies die die scenario's veroorzaken, maar deze zijn niet van belang voor de gebruiker. Die functies laten zien hoe het systeem in elkaar zit en niet wat het systeem moet kunnen. Daarom zijn in het ontwerp wel scenario's opgenomen, deze worden sequentiediagrammen genoemd in het ontwerp.

Aanpassingen van eisen na increment één

Naast de opbouw van het PvE zijn er inhoudelijk ook aanpassingen geweest. Een daarvan is de manier waarop de gebruiker informatie over een gegevensbron in de plug-in kan inzien. In versie 1.0 staat dat de gebruiker een gegevensbron kan selecteren en verwijderen, zie afbeelding 6.1. In versie 2.2 staat dat de gebruiker gegevensbronnen vast kan zetten en los kan maken, zie afbeelding 6.2.



Afbeelding 6.1: versie 1.0



Afbeelding 6.2: versie 2.2

Tijdens increment één was het de bedoeling dat de plug-in niets weergaf tot de gebruiker een gegevensbron uit CSP selecteerde. Dan moest de plug-in de geselecteerde gegevensbron weergeven. Om de gegevensbron weer te geven moet de plug-in informatie uit CSP halen. Tijdens het implementeren zag de opdrachtgever dat het beter is om elke gegevensbron die de gebruiker in CSP selecteert weer te geven. Als de gebruiker dan een gegevensbron in de plug-in wil houden kan deze vastgezet worden. Het voordeel hiervan is dat de gebruiker altijd ziet hoe een module in de plug-in weergegeven wordt en een weergave kan vasthouden die handig lijkt.

In het verlengde van de aanpassing van de manier waarop de gebruiker een module in de plug-in kan krijgen is er een eis toegevoegd. In die eis staat dat de plug-in de gegevensbron die is vastgezet moet onthouden als CSP afgesloten wordt, zie tabel 6.1.

Gemak

| | | | |
|--|---|----|----|
| De plug-in onthoudt welke module vaststaat als Lotus Notes afgesloten wordt. | M | 12 | 46 |
|--|---|----|----|

Tabel 6.1: toegevoegde eis over onthouden van gegevensbron

Een andere aanpassing is dat het meerdere keren installeren van de plug-in geen 'Must have' meer is, maar een 'Should have'. Tijdens de implementatie van increment één bleek het erg lastig te zijn om deze eis te realiseren. De ontwikkelaar

en de opdrachtgever hebben toen besloten om de prioriteit van deze eis te verlagen, zodat niet alle tijd verloren ging aan het realiseren van de eis. Er is tevens met de opdrachtgever gesproken over mogelijke alternatieven, bijvoorbeeld door meerdere secties in de plug-in of meerdere versies van de plug-in, waardoor meer dan één gegevensbron in de sidebar weergegeven kan worden. De realisatie daarvan valt buiten de opdracht.

De laatste grote verandering in het PvE gaat over de weergave van de informatie in de plug-in. Het prototype van increment één laat de kolommen van de geselecteerde gegevensbron zien. Bij veel gegevensbronnen past het aantal kolommen niet goed in de plug-in, daarom zijn de eisen over de weergave uit versie 1.0, zie tabel 6.2, gespecificeerd zoals te zien is in tabel 6.3.

| Weergave | | | | |
|----------|---|---|----|----|
| | De gebruiker kan een weergave selecteren. | S | I2 | 18 |
| | De gebruiker kan een weergave aanpassen. | S | I2 | 19 |

Tabel 6.2: eisen uit versie 1.0 over de weergave

| Weergave | | | | |
|----------|--|---|----|----|
| | De gebruiker kan kolommen uit de weergave verbergen. | M | I2 | 51 |
| | De gebruiker kan kolommen uit de weergave tonen. | M | I2 | 52 |

Tabel 6.3: eisen uit versie 2.2 over de weergave

6.1.3 Incrementplan

Ook voor increment twee is een incrementplan gemaakt waarin staat wat er tijdens het increment gedaan wordt en wat het oplevert. Increment twee levert onder andere een plug-in op.

Het doel van de plug-in staat in het plan van aanpak. In het tweede deel van dit project wordt veel aandacht besteed aan het generiek maken van de plug-in. Na increment één is de plug-in voor een deel generiek. De kolommen van de huidige geselecteerde module zijn te zien in de plug-in. Na increment twee moet de plug-in alleen de relevante kolommen van de huidige geselecteerde module weergeven. Daarnaast moet ook de inhoud van de module weergegeven worden in de plug-in. De gebruiker moet de module die op dat moment geselecteerd is ook vast kunnen zetten in de plug-in.

Verder bevat de plug-in functies om door informatie in de vastgezette module te zoeken of deze informatie te filteren of sorteren.

Naast de plug-in worden in het tweede increment ook documenten opgeleverd. Het programma van eisen dat in increment één gemaakt is wordt aangepast en aangevuld. Dat gebeurt met behulp van het prototype en de opgedane kennis over de gebruikte platformen.

Tijdens de ontwerpfase wordt een nieuw ontwerp gemaakt. Tijdens increment één werd duidelijk dat het van toegevoegde waarde is dat ook gedeelten van Eclipse en een gebruikte plug-in worden gemodelleerd. Dat komt ook terug in het model van increment twee.

Aan het einde van increment twee wordt de plug-in getest. De uitkomsten van de testen komen in een testrapport, deze bevindingen kunnen eventueel na het project worden gebruikt om de plug-in te optimaliseren.

Tijdens het afstuderen wordt een eindverslag gemaakt. Dat verslag wordt tijdens iteratie twee gemaakt. Alle gemaakte verslagen staan als bijlage in dat verslag, daarom is het eindverslag ook het afstudeerdossier.

6.2 Ontwerp

De ontwerpfase van increment twee leverde een ontwerp en een document op met een beschrijving van dat ontwerp. De beschrijving van het ontwerp staat in bijlage 11. Voor het ontwerp van increment twee is het ontwerp van increment één als basis genomen. Het ontwerp is uitgebreid en verbeterd. In het ontwerpdocument van increment twee staat geen beschrijving van de platformen, omdat deze al in increment één staan en onveranderd zijn gebleven. Een ander verschil met het ontwerpdocument van increment één is dat sequentiediagrammen zijn opgenomen.

Het packagediagram en de klassediagrammen uit dit ontwerp zijn completer dan de diagrammen uit het ontwerp van increment één. In het packagediagram is een package toegevoegd, namelijk Lotus Notes. Klassen uit deze package worden in de plug-in gebruikt. Dat was in increment één ook het geval, maar toen zijn deze niet gemodelleerd. Nu bevatten het packagediagram en de klassediagrammen alle benoemswaardige klassen die gebruikt worden door de plug-in.

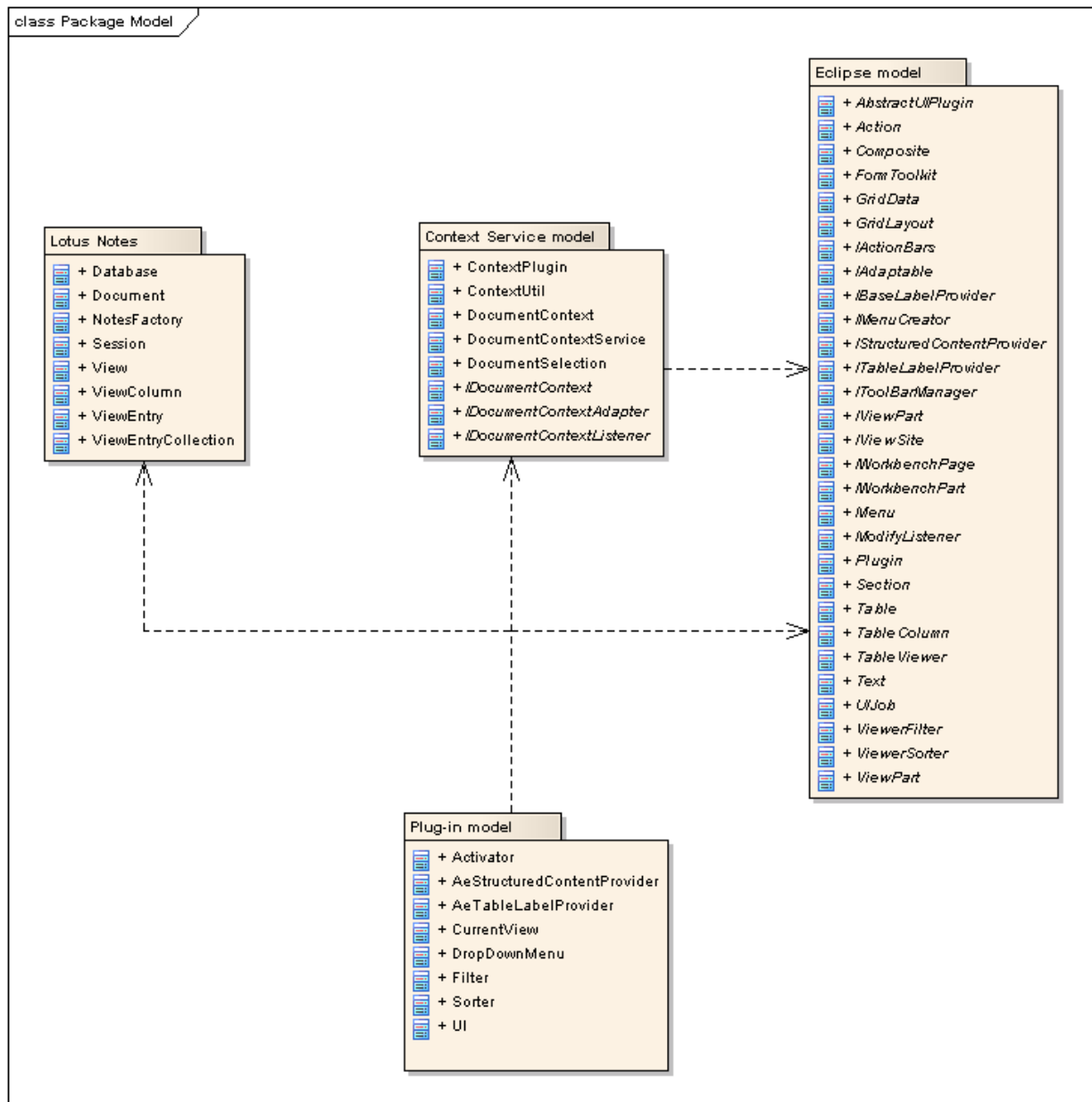
6.2.1 Packagediagram

Het packagediagram dat gemaakt is tijdens increment twee is te zien op afbeelding 6.3. Er zijn vier packages opgenomen in het diagram. Allereerst de te maken plug-in. Daarnaast de package Context Service, deze heeft dezelfde functie als in increment één, namelijk het opslaan van gegevens over het huidige geselecteerde document in Lotus Notes. Verder staat de package Eclipse in het diagram. Deze heeft meer klassen dan in increment één. Dat komt, omdat bij increment één alleen de belangrijkste klassen waar de plug-in gebruik van maakt waren opgenomen. In dit diagram staan alle klassen die gebruikt worden door de plug-in, daar is voor gekozen, omdat het diagram van vollediger is dan wanneer alleen de belangrijkste klassen worden opgenomen. Als laatste is de package Lotus Notes in het diagram opgenomen, omdat de plug-in gebruikt maakt van klassen uit Lotus Notes.

In het packagediagram zijn verschillende lagen te zien. De packages Lotus Notes en Eclipse zijn de meest stabiele packages en worden het minst vaak aangepast. Als een package stabiel is houdt dat in dat deze niet vaak verandert. Eclipse is stabiel, omdat veel mensen daar gebruik van maken. Als Eclipse niet stabiel zou zijn kunnen ontwikkelaars er niet vanuit gaan dat de klassen die gebruiker worden uit Eclipse hetzelfde blijven. Dan is het niet aantrekkelijk om het te gebruiken, want dan moet de ontwikkelaar zijn software testen en mogelijk aanpassen als Eclipse aangepast wordt.

De package Context Service is minder stabiel van Lotus Notes en Eclipse, maar stabielere dan de package plug-in. De plug-in is het minst stabiel.

Als een package niet stabiel is wordt bedoeld dat het gemakkelijk is om veranderingen toe te passen. De plug-in is ontwikkeld door AedifiComm en ook in beheer van hen. Als de plug-in verandert weet AedifiComm wat de gevolgen zijn, verder zijn er geen partijen afhankelijk van de plug-in. In paragraaf 5.4.2 van dit document staat meer uitleg over lagenstructuur in packagediagrammen.

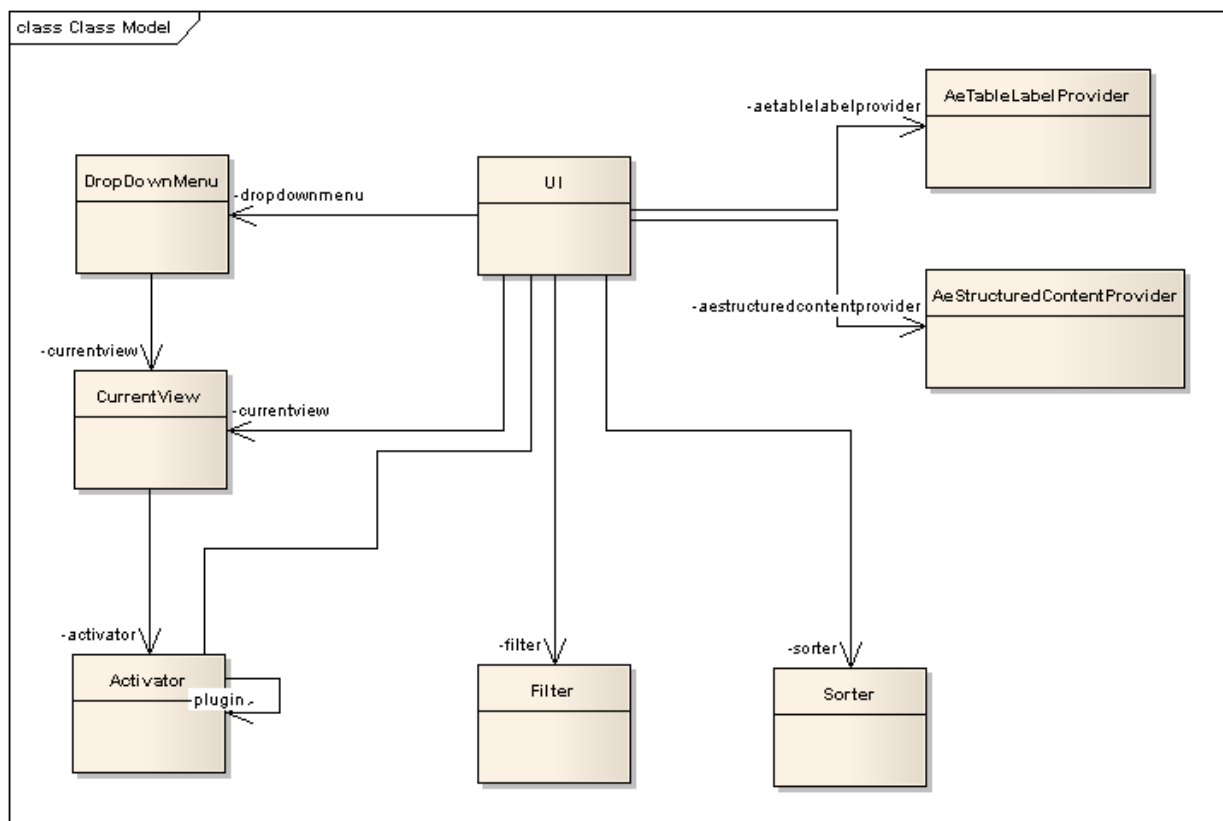


Afbeelding 6.3: packagediagram

6.2.2 Klassediagrammen

In het klassediagram, zie afbeelding 6.4, is te zien dat de klasse UI het middelpunt is van de plug-in. Deze klasse wordt aangemaakt bij het starten van de plug-in. Daardoor wordt de user interface van de plug-in opgebouwd. Voor die opbouw zijn veel klassen uit de package Eclipse nodig.

De klasse UI heeft een listener uit de package Context Service. Als de plug-in is opgebouwd zorgt die listener er voor dat een functie wordt aangeroepen op het moment dat de gebruiker een andere pagina selecteert in CSP. De functie die dan wordt aangeroepen haalt dan de view, die in de package Lotus Notes staat, op en geeft deze weer in de plug-in. Op deze manier laat de plug-in informatie zien over de huidige geselecteerde view.



Afbeelding 6.4: klassediagram van de plug-in

Tijdens het opbouwen van de plug-in wordt ook een toolbar gemaakt met acties daarin. De acties in de plug-in zijn:

- Huidige geselecteerde view vastzetten
- Informatie filteren (klasse Filter)
- Kolommen tonen of verbergen (klasse DropDownMenu)

De gebruiker kan op een van deze acties klikken, waardoor deze wordt uitgevoerd. Op die manier kan de huidige geselecteerde view worden vastgezet in de plug-in.

Als de gebruiker een andere pagina selecteert blijft de view die is vastgezet zichtbaar. Als een pagina is vastgezet kan de gebruiker de overige acties gebruiken, zolang een weergave niet is vastgezet zijn die acties niet beschikbaar. Met de filteractie kan de gebruiker een filter in- en uitschakelen en een waarde opgeven waarop gefilterd moet worden. De laatste actie in de toolbar is een drop-down menu met daarin acties. Als het menu wordt opengeklapt staan daar de kolommen van de vastgezette view. De gebruiker kan in het menu op een kolom klikken waardoor deze wordt getoond of verborgen.

Het vastzetten van de huidige geselecteerde view is geen aparte klasse, omdat bij die actie alleen een instelling van de plug-in wordt veranderd.

In het programma van eisen is te lezen dat de plug-in verschillende instellingen moet kunnen onthouden. Op die manier hoeft de gebruiker de plug-in niet elke keer opnieuw in te stellen als CSP wordt opgestart. De plug-in moet verschillende instellingen kunnen onthouden:

- De vastgezette view
- Zichtbare en onzichtbare kolommen
- Waarde van de filter

De plug-in bevat functies die het mogelijk maken om de instellingen te onthouden. Een andere eis is dat verschillende gebruikers op dezelfde computer gebruik kunnen maken van hun persoonlijke plug-in. Dat is te realiseren door de plug-in en de bijbehorende instellingen op te slaan in de persoonlijke map van Lotus Notes, waarop CSP is gebaseerd. Bij het installeren van de plug-in wordt deze standaard in de persoonlijke map gezet, dus daar hoeft in het ontwerp geen rekening mee worden gehouden.

6.2.3 Sequentiediagrammen

De manier waarop de sequentiediagrammen zijn gemaakt is te vergelijken met de use-case-diagrammen. Er is een globaal model en er zijn modellen die gedeelten van het globale model gedetailleerd weergeven.

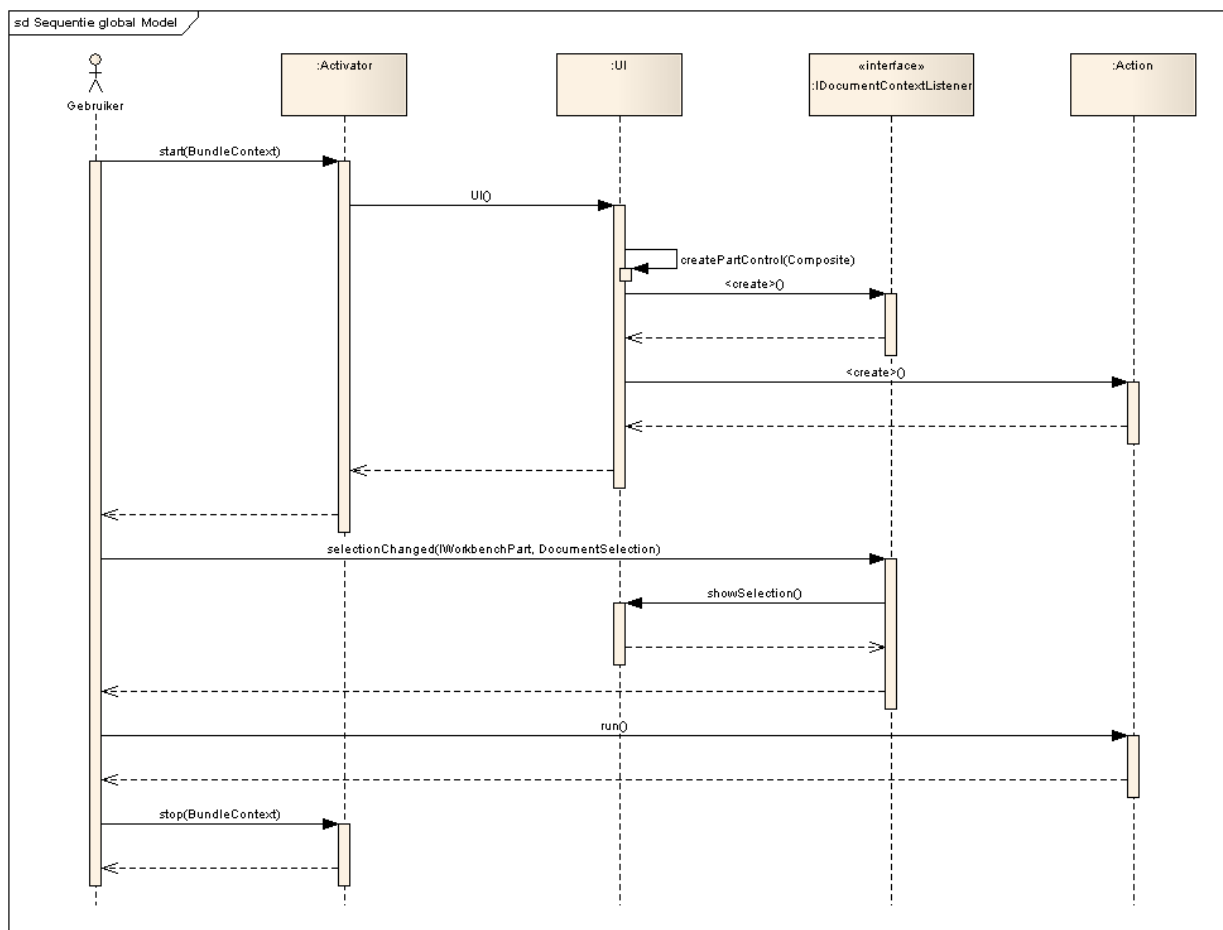
De diagrammen kloppen niet volledig met hoe UML het voorschrijft, maar daar is bewust voor gekozen. Het doel van de diagrammen die tijdens dit project gemaakt zijn is om duidelijk te maken wat de gebruiker met het systeem kan en wat er dan in het systeem gebeurt. Tijdens het maken van de diagrammen lag de hoogste prioriteit bij duidelijkheid en niet bij de voorschriften van UML.

Het globale sequentiediagram staat op afbeelding 6.5. Daarin is te zien dat het diagram is opgedeeld in vier delen:

1. Starten en opbouwen van de plug-in
2. Veranderen van de inhoud van de plug-in
3. Uitvoeren van acties
4. Sluiten van de plug-in

Het starten en opbouwen van de plug-in bestaat vooral uit het creëren van objecten. Daarbij worden veel klassen van Eclipse gebruikt, namelijk:

- IViewPart
- IWorkbenchPart
- Viewpart
- FormToolkit
- Section
- Composite
- GridData
- Table
- TableViewer
- TableColumn

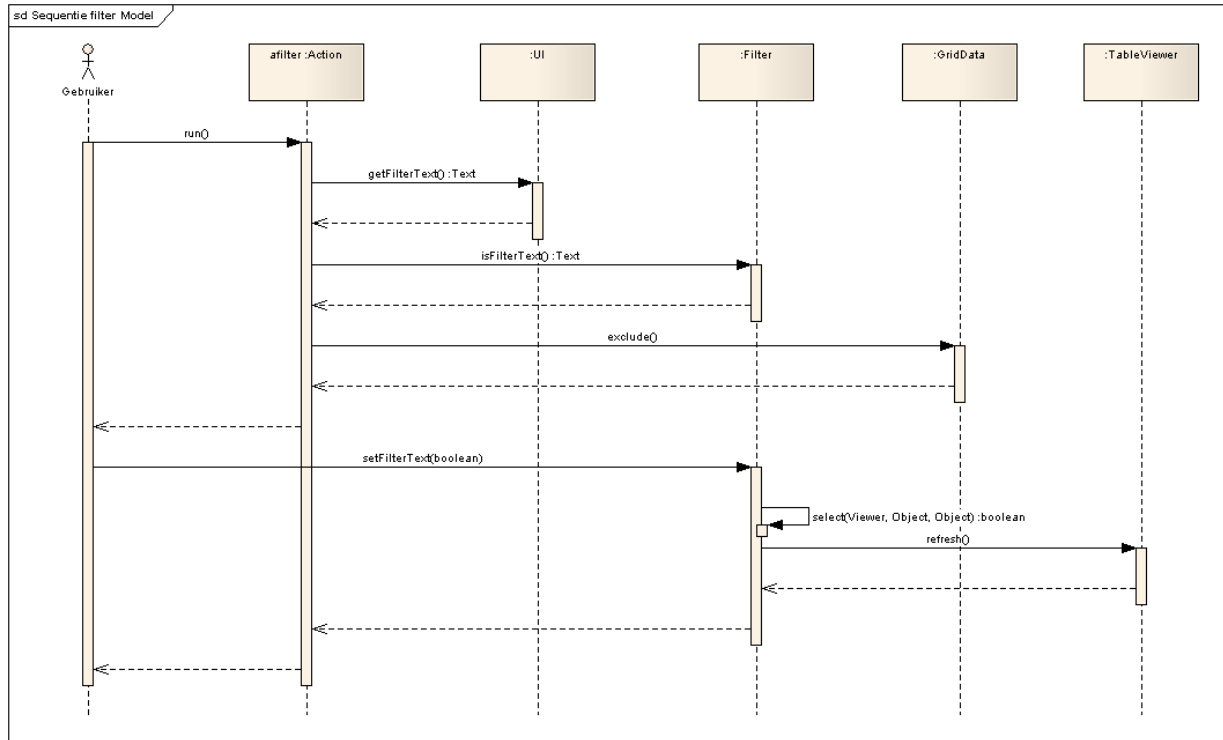


Afbeelding 6.5: globaal sequentiediagram van de plug-in

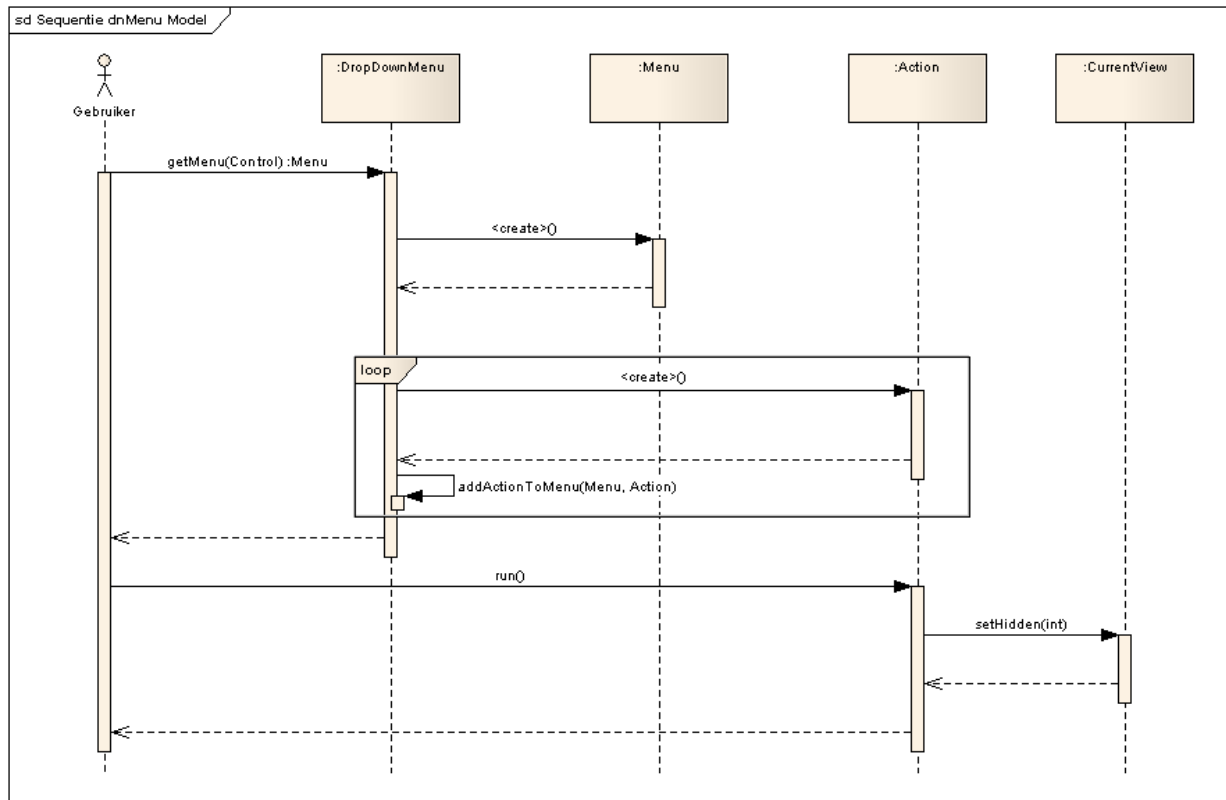
Het tweede deel van het globale diagram bestaat uit het veranderen van de inhoud van de plug-in. Dat gebeurt als de gebruiker een andere gegevensbron selecteert. De sequentiediagrammen van het opbouwen van de plug-in het het veranderen van de inhoud van de plug-in zijn te vinden in bijlage 11.

Zoals ook in paragraaf 6.2.1 staat kan de gebruiker drie acties in de plug-in uitvoeren. Het vastzetten van de view kan de gebruiker bereiken met een druk op de knop. Voor de filter en de zichtbaarheid van kolommen is een drop down menu gemaakt in de toolbar. Elk item van het menu is een aparte actie. Als de gebruiker

het menu uitklapt wordt deze gemaakt. Het diagram voor de filter is te vinden op afbeelding 6.6 en het diagram voor de zichtbaarheid van kolommen op afbeelding 6.7.



Afbeelding 6.6: drop down menu voor de filter



Afbeelding 6.7: drop down menu voor zichtbaarheid van kolommen

6.3 Implementatie

Tijdens de implementatie van increment twee is verder gegaan met het prototype dat tijdens increment één is opgeleverd. Dat prototype bouwt de plug-in op, waarna het wacht tot de gebruiker een gegevensbron selecteert. Op dat moment wordt een functie in de plug-in aangeroepen die de kolommen die zichtbaar zijn in de gegevensbron weergeeft in de plug-in. De inhoud van de gegevensbron wordt niet weergegeven in de plug-in.

6.3.1 Plug-in vullen met gegevens uit CSP

Zoals gezegd geeft de plug-in nog geen gegevens weer uit de geselecteerde gegevensbron. Dat realiseren was de hoogste prioriteit tijdens de implementatie van increment twee.

Om gegevens in de plug-in te zetten is gebruik gemaakt van de klasse `TableViewer` uit de SWT van Eclipse. Aan die klasse moet een tabel toegekend worden. Daarnaast kunnen er twee soorten providers aan de `TableViewer` worden toegekend, zoals te zien is in code 6.1. De providers zijn nodig om de tabel te vullen met gegevens uit de geselecteerde gegevensbron.

```
private TableViewer tableviewer;
private Table table;
```

```

public void createPartControl(Composite parent) {
    ...
    //give the tableviewer some providers
    tableviewer = new TableViewer(table);
    tableviewer.setContentProvider(new AeStructuredContentProvider());
    tableviewer.setLabelProvider(new AeTableLabelProvider());
    ...
}

```

Code 6.1: toekennen van Table en providers aan TableViewer

De TableViewer heeft een functie "setInput", die kan alleen gebruikt worden als de viewer een content provider heeft. Aan de functie "setInput" kan een object meegegeven worden waarin de input voor de plug-in staat. In dit geval gebeurt "onder de motorkap" van Eclipse wat, want "setInput" zorgt ervoor dat de functie "getElements" in de content provider wordt aangeroepen. Meer uitleg hierover staat op [Vogella1005].

De functie "setInput" wordt gevuld met de geselecteerde gegevensbron. Hoe die bron wordt opgehaald staat in hoofdstuk 5 van dit document. Als dat is gebeurd heeft de content provider de gegevens die in de plug-in moeten komen.

De functies van de content provider en de label provider zijn verschillend. De content provider (in de plug-in een implementatie van IStructuredContentProvider) zet de gegevens in een array van objecten. In de plug-in worden de gegevens in een meerdimensionale array gezet waarvoor geldt: array[aantal rijen][aantal items in de rij]. De functie "getElements" is te zien in code 6.2, daar wordt ook duidelijk hoe de gegevens uit de geselecteerde gegevensbron (variabele "view") worden gehaald.

```

public Object[] getElements(Object obj){
    List <Object> elements = new ArrayList();
    String[][] list = null;

    if ( obj instanceof View ) {
        View view = (View) obj ;

        try {
            //get all the entries of the view
            ViewEntryCollection collection = view.getAllEntries();
            ViewEntry entry = collection.getFirstEntry();
            Vector <ViewColumn> vectorcolumns = view.getColumns();
            List <ViewColumn> listcolumns = new ArrayList();
            int amountcolumns = 0;
            for (Enumeration <ViewColumn> e = vectorcolumns.elements();
                e.hasMoreElements(); ) {
                listcolumns.add(e.nextElement()) ;
            }

            //set every value of the view in the array
            list = new
            String[collection.getCount()][view.getColumnCount()];
            int count = 0;

```

```

        while (entry != null){
            Vector vals = entry.getColumnValues();
            int j = 0;
            int i = 0;
            for (i = 0; i<vals.size(); i++){
                if (!listcolumns.get(i).isHidden() &&
                    !listcolumns.get(i).isIcon()){
                    list[count][j] =
                        vals.elementAt(i).toString();
                    j++;
                }
            }
            count++;
            entry = collection.getNextEntry(entry);
        }
    }
    catch (NotesException e1) {
        e1.printStackTrace();
    }
}
return list;
}

```

Code 6.2: de functie "getElements" in de content provider

Als "getElements" klaar is zorgt de TableView er voor dat de functie "getColumnText" in de label provider wordt aangeroepen met als parameter de returnwaarde van "getElements". De label provider (in de plug-in een implementatie van ITableLabelProvider) zorgt er dan voor dat de gegevens op de juiste plek komen in de tabel.

Om bij elke verandering van gegevensbron de informatie van de gegevensbron in de plug-in te krijgen met de functie "setInput" van de TableView worden aangeroepen als de gebruiker een andere gegevensbron selecteert.

6.3.2 Acties in de plug-in

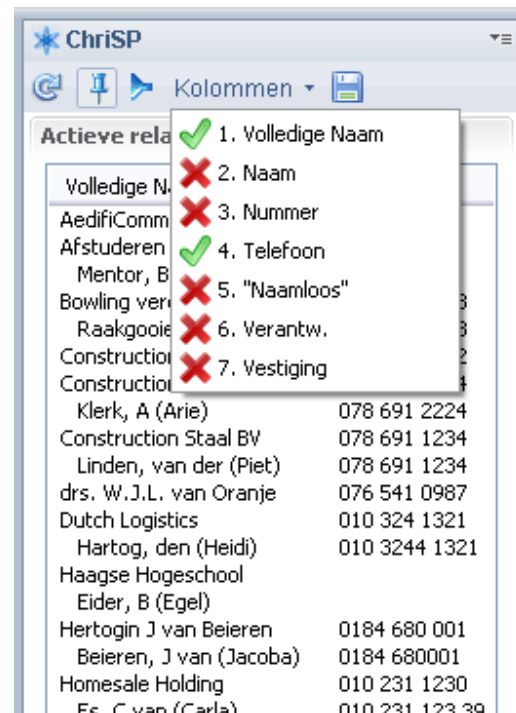
Op het moment dat de plug-in gevuld met gegevens kunnen er acties uitgevoerd worden op die gegevens. De acties die tijdens dit project zijn vastgelegd in de eisen zijn:

- De inhoud van de plug-in vastpinnen
- Informatie filteren
- Kolommen verbergen en tonen
- De plug-in vernieuwen
- Instellingen opslaan
- Informatie sorteren
- Informatie zoeken

Tijdens de implementatie zijn acties "vastpinnen", "filter", "kolommen", "vernieuwen" en "opslaan" geïmplementeerd. Als de plug-in wordt opgebouwd worden de acties gemaakt. Deze staan aan in de toolbar van de plug-in, zie afbeelding 6.8.



Afbeelding 6.8: toolbar met acties

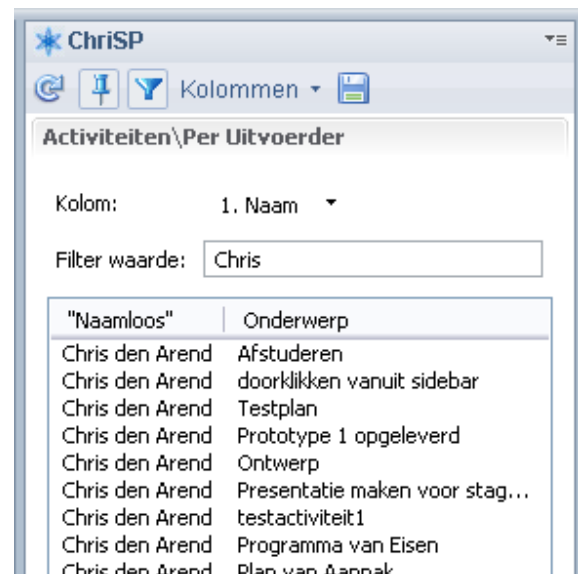


Afbeelding 6.9: actie "kolommen"

De actie "vastpinnen" zorgt ervoor dat de informatie die op dat moment in de lpug-in staat blijft staan, ook al selecteert de gebruiker een andere gegevensbron. Als de inhoud van de plug-in vastgezet is dan werken de acties "filter" en "kolommen". Als de inhoud niet is vergezet werken die acties dus niet.

Voor de acties "filter" en "kolommen" is een drop-down menu gemaakt, elk item in het menu is een actie. Daar is voor gekozen, omdat beide acties voor elke kolom apart gelden. Een kolom verbergen of zichtbaar maken kan nu doormiddel van het klikken op een menuitem, zie afbeelding 6.9.

De actie "filter" kan door de gebruiker aan en uit worden gezet. Als de actie aanstaat ziet de gebruiker een drop-down menu en een tekstveld. In het drop-down menu kan de kolom gekozen worden waarop gefilterd moet worden en in het tekstveld moet de filterwaarde worden neergezet, zie afbeelding 6.10.



Afbeelding 6.10: actie "filter"

Het vernieuwen van de plug-in gebeurt met de actie "vernieuwen". De gebruiker kan de plug-in op elk moment vernieuwen. Daarnaast heeft de plug-in een timer die elke vijf minuten afloopt en op dat moment wordt de plug-in ook vernieuwd, zie code 6.3.

```
private Timer timer;

public void createPartControl(Composite parent) {
    ...
    timer = new Timer();
    timer.scheduleAtFixedRate(new TimerTask() {
        public void run() {
            showSelection(currentpart, currentselection);
        }
    }, 300000, 300000);
    ...
}
```

Code 6.3: de timer

Het PvE heeft verschillende eisen die gaan over het gemak van de plug-in. Een daarvan is dat de plug-in gegevens moet onthouden, zodat bij het opnieuw opstarten van Lotus Notes de instellingen van de gebruiker niet verdwenen zijn. De gebruiker kan de plug-in instellen met de actie "filter" als de gewenste informatie in de plug-in staat kan de actie "opslaan" worden aangeklikt. Op dat moment worden de benodigde gegevens in de plug-in opgeslagen, met de functie:

```
plugin.getPluginPreferences().setValue("naam van de waarde", "waarde");
```

6.4 Testen

De testfase volgde op de implementatiefase. In deze fase werden een testplan en een testrapport gemaakt, deze zijn te vinden in bijlage 12 en bijlage 13. Het testplan is gemaakt aan de hand van de eisen die in het incrementplan van increment twee zijn vastgelegd. De testen zijn uitgevoerd door de ontwikkelaar.

6.4.1 Testplan

Er is getest op drie verschillende operating systems, namelijk Windows XP, Windows Vista en Windows 7. Dat is gedaan, omdat in de eisen staat dat de plug-in op die operating systems moet werken.

Het testplan is zo opgebouwd dat de eisen die in het incrementplan van increment één staan getest worden. Er is vanwege tijdgebrek gekozen om geen testcases te maken waar combinaties van eisen aan bod komen. Net als in increment één zijn de testen uitgevoerd door de ontwikkelaar zelf.

6.4.2 Testrapport

Het testrapport beschrijft de uitkomsten van de testen die in het testplan beschreven staan. Bij het maken van het testplan zijn de eisen die nog niet geïmplementeerd waren niet meegenomen. De eisen die wel geïmplementeerd waren zijn afzonderlijk van elkaar getest en alle uitkomsten waren goed.

6.5 Invoer

Bij de invoerfase is het de bedoeling dat bedacht wordt hoe een product het beste ingevoerd kan worden. Daarbij moet dan vooral rekening gehouden worden met de gebruikers, die moeten zo min mogelijk gehinderd worden in hun werkzaamheden. Als het plan klaar is wordt het product werkelijk ingevoerd.

Na de invoerfase van dit project is de plug-in nog niet voor gebruikers beschikbaar. AedifiComm brengt regelmatig nieuwe versies uit van CSP en de plug-in komt in de volgende versie van CSP.

Wel is bedacht hoe de plug-in ingevoerd kan worden. Lotus Notes biedt veel ondersteuning voor het ontwikkelen van plug-ins en ook voor de invoer daarvan. Het is namelijk mogelijk om een updatesite te maken op de Domino server die gekoppeld is aan de Lotus client van de gebruiker. Op de updatesite kan een plug-in worden geüpload.

Dan zijn er verschillende mogelijkheden voor de gebruiker om de plug-in te installeren. Dat kan doormiddel van het slepen van een XML-file in de sidebar van Lotus Notes, die manier werkt bij de plug-in.

Daarnaast kan de gebruiker via het menu van Lotus Notes een plug-in installeren, dan moet de URL van de updatesite wel bekend zijn. Deze manier werkt ook bij de plug-in, maar is niet erg gebruikersvriendelijk.

De ideale manier voor de gebruiker om de plug-in te installeren is een menu in CSP met daarin de plug-in. Als de gebruiker op de plug-in klikt wordt deze geïnstalleerd. Deze manier is de beste, omdat AedifiComm van plan is om in de toekomst meer plug-ins te maken voor CSP. Die plug-ins kunnen dan ook in dat menu.

7 Conclusies

In dit hoofdstuk worden de conclusies van het afstudeerproject besproken. Als eerste komen de conclusies van de aanpak van het project aan bod, daarna van de producten.

7.1 Aanpak

De aanpak was in twee incrementeel ingedeeld, in deze paragraaf worden de incrementen afzonderlijk besproken en daarna de complete aanpak.

7.1.1 Increment 1

Increment één verliep anders dan verwacht. Tijdens de ontwerp en implementatiefase is gedeeltelijk iteratief gewerkt, terwijl de planning was om dat volledig incrementeel te doen. Dat was nodig, omdat er nog te weinig kennis was over de functies van klassen in Eclipse, waarmee gewerkt moest worden. De geplande tijd voor elke fase in increment één is niet overschreden.

7.1.2 Increment 2

Increment twee verliep voorspoedig, maar liep een week uit. Tijdens het implementeren bleek dat er teveel informatie die in de plug-in aanwezig was. Daardoor hebben de ontwikkelaar en opdrachtgever besloten om meer prioriteit te geven aan de eis dat gebruikers informatie kunnen verbergen en zichtbaar kan maken. Daarnaast waren er veel vrije dagen waar in de planning geen rekening mee was gehouden. De aanpassing van de prioriteit van de eis en de vrije dagen zorgden voor een week uitloop.

7.2 Plug-in

Het praktische antwoord op de hoofdvraag die gesteld is in de inleiding "Hoe kan de sidebar van het IBM Lotus Notes en Domino platform worden ingezet om door middel van een configureerbare en generieke plug-in de gewenste informatie altijd zichtbaar te maken?" is de plug-in die tijdens dit project ontwikkeld is. In deze paragraaf komt de plug-in aan bod.

7.2.1 Prototype van plug-in na increment 1

Het doel was om na increment één een prototype af te hebben die geïnstalleerd en gedeïnstalleerd kan worden in de sidebar van Lotus Notes. Daarnaast moest het prototype informatie over een vastgestelde gegevensbron weergeven als de gebruiker deze had geselecteerd.

Het resultaat van increment één is een prototype die geïnstalleerd en gedeïnstalleerd kan worden in de sidebar van Lotus Notes. Het weergeven van de informatie is niet gerealiseerd zoals de bedoeling was. Als de gebruiker een willekeurige gegevensbron selecteert geeft het prototype een tabel weer met daarin de kolommen die ook in de geselecteerde gegevensbron staan. Het prototype is dus al generiek, maar laat nog geen informatie zien in de kolommen.

7.2.2 Plug-in na increment 2

Het doel was om na increment twee een plug-in af te hebben voor de sidebar van Lotus Notes. De gebruiker kan een gegevensbron vastzetten in de plug-in, zodat die altijd zichtbaar is.

Het resultaat van increment twee voldoet aan de doelstelling. De gebruiker kan een gegevensbron vastzetten in de plug-in. Als dat gebeurd is kan de gebruiker kolommen uit de plug-in verbergen en zichtbaar maken. Daarnaast kan de informatie in de plug-in gefiltert worden en wordt de plug-in regelmatig up to date gemaakt.

8 Aanbevelingen

In dit hoofdstuk staan verschillende aanbevelingen aan AedifiComm voor de toekomst van de gemaakte plug-in.

8.1 Toevoegingen voor de plug-in

De plug-in voldoet aan de gestelde eisen, maar dat betekent niet dat er geen nuttige toevoegingen bestaan voor de plug-in. Daarom staan in deze paragraaf enkele toevoegingen voor de plug-in:

- Informatie in de plug-in sorteren en zoeken
- De plug-in uitvoerig testen, ook op lagere versies van Lotus Notes en door andere medewerkers van AedifiComm en eindgebruikers van de plug-in
- De mogelijkheid implementeren dat gebruikers ook informatie kunnen bewerken in de plug-in en kunnen doorklikken om een document te openen

Sorteren en zoeken

Op dit moment is het voor de gebruiker mogelijk om informatie in de plug-in in te zien en te filteren. Het is niet mogelijk om informatie per kolom te sorteren of informatie te zoeken. Die twee functies zijn wel nuttig en maken de plug-in gebruikersvriendelijker.

Testen

Tijdens dit project was er niet veel tijd om de plug-in uitvoerig te testen. Elke eis is apart getest, maar het is aan te bevelen om de plug-in grondiger te testen, zodat de kwaliteit van de software gewaarborgd wordt.

Verder was een eis dat de plug-in moest werken met Lotus Notes 8.5, maar veel klanten van AedifiComm werken nog met oudere versies van Lotus Notes. Om die reden is het nuttig om de plug-in te testen op lagere versies van Lotus Notes. Vanaf versie 8.0 bevat Lotus Notes een sidebar, dus kan de plug-in vanaf die versie getest worden.

Informatie bewerken

Met de plug-in hoeft de gebruiker niet elke keer naar een gegevensbron te navigeren om informatie in te zien. Wat de plug-in echter niet mogelijk maakt is het aanpassen van informatie. Daardoor moet een gebruiker die aanpassingen wil maken op de gegevensbron die in de plug-in staat alsnog navigeren naar gegevensbron in CSP. Een goede uitbreiding voor de plug-in is de mogelijkheid om in de plug-in op een item te klikken waardoor het oorspronkelijke document in het hoofdscherm van CSP wordt geopend. De gebruiker kan dan aanpassingen maken in het document.

8.2 Aanpassing in CSP

Tijdens het implementeren van de plug-in kwam een probleem naar voren. Het probleem ontstaat als de inhoud van een kolom in CSP niet variabel is. Bij het ophalen van de kolommen worden alle kolommen opgehaald, ook de kolommen waarvan de inhoud niet variabel is. Echter als de inhoud van de kolommen

opgehaald wordt dan worden de kolommen met een constante waarde niet herkend. Het gevolg is dat in de plug-in meer kolommen zijn dan er inhoud is. Dus als er één kolom is waarvan de inhoud variabel is blijft de laatste kolom in de plug-in leeg. Dit probleem kan worden voorkomen door alle waarden in CSP variabel te maken. Als een kolom nu een constante waarde heeft dan moet die waarde aan een variabele worden toegekend. Vervolgens kan die variabele in de CSP view kolom weergegeven worden.

8.3 Nieuwe versie Eclipse

Om een plug-in te ontwikkelen met Eclipse en Lotus Notes is er enige gebondenheid aan platformen. Zo moet de Lotus Expeditor Toolkit gebruikt worden. De nieuwste versie van de Lotus Expeditor Toolkit (versie 6.2) werkt met Eclipse 3.4, terwijl de nieuwste versie van Eclipse versie 3.5 is. Om een plug-in voor Lotus Notes te maken kan dus niet gewerkt worden met de nieuwste versie van Eclipse. Dat is een nadeel, want Eclipse biedt in hogere versies meer ondersteuning voor bijvoorbeeld de klasse Table en TableColumn. Tijdens dit project was een oplossing voor handen om kolommen te verbergen en zichtbaar te maken, maar de header van de klasse Table kan niet gebruikt worden voor een menu, zie [Eclipse0705]. Daarom is het advies aan AedifiComm om rekening te houden met nieuwe versies van de Lotus Expeditor Toolkit en Eclipse. Dan is het in de toekomst mogelijk om gebruikersvriendelijkere functies te gebruiken dan dat er nu in de plug-in is geïmplementeerd.

9 Eindproduct

In dit hoofdstuk worden de opgeleverde producten besproken en wordt beschreven op welke manier ze hebben bijgedragen aan het resultaat. Daarnaast wordt elk product geëvalueerd.

9.1 Plan van Aanpak

Het eerste product dat ik heb opgeleverd was het plan van aanpak. In dat plan wordt verder gegaan met het afstudeerplan.

9.1.1 Eisen

In het plan staan een aantal eisen voor de plug-in. Aan het einde van het project kloppen de meeste eisen uit het plan van aanpak nog met de wens van de opdrachtgever. Een van de eisen die niet meer van toepassing is gaat over het slepen van een module in de plug-in. Het doel daarvan was om informatie uit die module weer te geven in de plug-in. Nu gebeurt het weergeven nog wel, maar de gebruiker kan de module vastzetten in de plug-in in plaats van slepen. Aan de functionaliteit is dus wat veranderd, maar het resultaat is hetzelfde.

9.1.2 Op te leveren producten

Naast eisen staan ook de op te leveren producten in het plan van aanpak. Die opsomming klopt niet goed met wat er werkelijkheid is opgeleverd tijdens het project. De plannen voor increment één en increment twee staan niet vermeld en documentatie over het testen mist. Er staat ook een tussenverslag tussen de op te leveren producten. Dat tussenverslag moet voortgangsverslag zijn. Mijn beeld van het voortgangsverslag was een verslag over de eerste helft van het afstuderen. Later bleek dat een korte beschrijving over de status van de afstudeeropdracht.

9.1.3 Planning

Tijdens het project heb ik me voor het grootste gedeelte aan de planning gehouden die in het plan van aanpak staat. Wel ervaarde ik dat het niet haalbaar is om over elk verslag feedback te krijgen van mijn bedrijfsmentor. Bij sommige verslagen heeft dat ook geen hoge prioriteit. De incrementenplannen zijn voor een groot gedeelte afgeleid uit het programma van eisen, dus zou daar twee keer dezelfde feedback op gegeven worden. Daarnaast moet ik aantonen dat ik zelfstandig ben. Als dan bij elk document veel feedback gegeven moet worden getuigt dat niet van zelfstandigheid. In praktijk werd op belangrijke punten in het project de stand van zaken geëvalueerd. Daarnaast gaf mijn bedrijfsmentor duidelijk aan wanneer feedback nodig was en maakte we een afspraak om die feedback door te nemen.

9.2 Programma van Eisen

Versie 1.0 van het PvE heb ik opgeleverd in de analysefase van increment één. Het document bestaat uit use-case-diagrammen, scenario's en een lijst met alle eisen. Daarnaast bevat het een uitleg over welke instellingen de plug-in moet onthouden als Lotus Notes wordt afgesloten.

De scenario's en de uitleg over het onthouden van instellingen zijn in het document opgenomen, omdat de samenhang tussen use-cases niet duidelijk werd door de use-case-diagrammen. De keuze om met scenario's de samenhang duidelijker te maken heeft niet veel nut gehad, want de scenario's maakten de samenhang niet duidelijker. Het was beter geweest om de lijst met eisen duidelijker te maken, want in de use-case-diagrammen stonden een aantal eisen die niet in de lijst met eisen stonden.

Versie 2.2 van het PvE die tijdens increment twee gemaakt is bevat use-cases en een lijst met alle eisen. Net als in versie 1.0 is ook in deze versie een hoofdstuk waarin uitgelegd wordt welke instellingen de plug-in moet onthouden als Lotus Notes afgesloten wordt.

De scenario's zijn weggelaten uit versie 2.2, omdat deze de eisen niet duidelijker maakten. Verder zijn de eisen in duidelijker en gedetailleerder in versie 2.2 dan in versie 1.0. Dat is logisch, want in increment twee is meer informatie beschikbaar over de plug-in en de platformen die gebruikt werden.

9.3 Ontwerp

In beide incrementen heb ik een ontwerp gemaakt. Die ontwerpen moeten ervoor zorgen dat de software die tijdens de implementatie gemaakt wordt van hoge kwaliteit is.

9.3.1 Increment 1

Het ontwerpdocument uit increment één bevat uitleg over de platformen die gebruikt zijn en de diagrammen die gemaakt zijn.

Het gedeelte over de platformen is niet erg technisch. Ik had er ook voor kunnen kiezen om meer technische details over de platformen te geven. Dat heb ik bewust niet gedaan, want mijn doel was een omgeving creëren waarmee ik een plug-in kon ontwikkelen. Om dat te bereiken is niet veel gedetailleerde kennis nodig over de platformen, dus leek het mij zonde van de tijd om daar dieper op in te gaan.

De diagrammen die in het document staan hebben weinig bijgedragen aan de implementatie van het prototype. Dat komt, omdat ik bij aanvang van het ontwerpen weinig kennis had over Eclipse en Lotus Notes. Die twee platformen bevatten veel klassen die ik kon gebruiken voor de plug-in. Om kennis te vergaren over die klassen heb ik voorbeelden van plug-ins gebruikt en ben ik gaan testen of de funties die ik wilde gebruiken nuttig waren. Daardoor was het prototype voor een groot deel af voordat het ontwerp af was.

9.3.2 Increment 2

Het ontwerpdocument uit increment twee bevat een packagediagram, klassediagrammen en sequentiediagrammen.

Ontwerpen in increment twee was een stuk eenvoudiger dan in increment één. Tijdens increment één heb ik veel kennis opgedaan over Eclipse en Lotus Notes.

Dat zorgt ervoor dat het ontwerp realistisch is en tijdens de implementatie heb ik het ontwerp een stuk minder aangepast dan ik in increment één deed. Het ontwerp is voor een groot gedeelte gerealiseerd tijdens de implementatie. Er zijn wel een paar verschillen zoals de klasse sorter die wel gemodelleerd is, maar is niet geïmplementeerd.

9.4 Plug-in

Na de implementatiefase van increment één heb ik een prototype van de plug-in opgeleverd. De implementatie van increment twee leverde de eerste versie van de plug-in op.

9.4.1 Prototype

Het prototype kwam niet overeen met het PvE uit increment één. De bedoeling was om de plug-in pas in increment twee generiek te maken. In increment één moest de plug-in werken met één gegevensbron uit CSP. Tijdens de implementatie merkte ik dat het even veel moeite kost om de plug-in te laten werken met één gegevensbron als om de plug-in te laten werken met alle gegevensbronnen in CSP. Om die reden heb ik de plug-in tijdens increment één al generiek gemaakt. Verder voldeed het prototype niet aan de eis dat het alle informatie liet zien uit een geselecteerde gegevensbron. Het prototype liet alleen de kolommen zien die ook in de geselecteerde gegevensbron aanwezig waren.

Een positief punt is dat de plug-in een generieke basis had. Een minder goed punt is dat de plug-in alleen de kolommen van de geselecteerde gegevensbron weergeeft en niet de inhoud.

9.4.2 Eerste versie van de plug-in

De eerste versie van de plug-in voldeed voor het grootste deel aan de eisen die in het PvE tijdens increment twee zijn vastgelegd. Alle eisen die als prioriteit 'Must have' hadden zijn geïmplementeerd, behalve het sorteren van kolommen. Verder onthoudt de pug-in de instellingen van de filter als Lotus Notes afgesloten wordt, die eis had als prioriteit 'Should have'.

9.5 Testen

De testen die ik tijdens increment één en increment twee heb uitgevoerd zijn vergelijkbaar. De plug-in is getest op drie operating systems, namelijk Windows XP, Windows Vista en Windows 7. De testplannen zijn gemaakt aan de hand van de PvE's.

Het testen kon een stuk grondiger, maar dat is in verband met de tijd niet gedaan. Nu zijn alle eisen apart getest. Het testen van verschillende eisen na elkaar is niet gebeurd.

9.6 Eindverslag

Het eindverslag van dit project heb ik opgedeeld in drie delen. Dat heb ik gedaan na het onderzoeken van de eisen voor het verslag en het bekijken van ongeveer tien

andere eindverslagen van afstudeerders. Daarbij zaten verslagen over technische opdrachten, maar ook verslagen over niet-technische opdrachten. Dat heb ik bewust gedaan, omdat in niet-technische verslagen misschien aspecten zitten die niet in technische verslagen zitten, maar wel nuttig zijn voor mij verslag. Een voorbeeld is dat in niet-technische verslagen over het algemeen meer wordt stilgestaan bij leerdoelen dan bij technische verslagen. In mijn eindverslag worden de leerdoelen wel besproken.

Over het algemeen zijn de eindverslagen opgedeeld in een aanpak, een beschrijving van de werkzaamheden en een terugblik op het afstuderen. Ook in mijn eindverslag zijn deze drie onderwerpen de rode draad door het verslag.

In de aanpak staat de situatie hoe die was voor het afstuderen en de strategie en methode waarmee ik het project aan heb gepakt. In de beschrijving van de werkzaamheden staat wat ik heb gedaan tijdens het project. Als laatste de terugblik, daarin wordt beschreven wat ik van het project vond en wat de conclusies van het project zijn.

10 Evaluatie

Voordat de opdracht begon heb ik leerdoelen opgesteld en de aanpak van het project bepaald. In dit hoofdstuk worden die twee geëvalueerd.

10.1 Leerdoelen

In paragraaf 3.6 van dit document staan de leerdoelen die bij aanvang van het afstuderen zijn bepaald. In deze paragraaf wordt duidelijk of ik die doelen heb gehaald en wat ik heb geleerd.

10.1.1 Technische doelen

Leren werken met Lotus Notes en Eclipse

Tijdens heel het afstuderen heb ik gewerkt met Lotus Notes en Eclipse en daardoor heb ik er mee leren werken. In de definitiefase van increment één heb ik een cursus over Lotus Notes gedaan, waardoor ik basiskennis over het platform kreeg. Wat niet in het doel staat, maar wat ik wel heb geleerd is het werken met CSP, het softwarepakket van AedifiComm. CSP is gebaseerd op Lotus Notes, dus technisch zit er niet veel verschil tussen die twee.

Wat tijdens het afstuderen vaak naar voren kwam was dat Lotus Notes is opgebouwd uit databases met daarin notes. In die notes zitten dan weer items. Wat dat precies betekent voor het ontwikkelen van de plug-in komt naar voren in het stuk over niet-relatieve databases.

Naast Lotus Notes heb ik ervaring opgedaan met het open source platform Eclipse. Het is een verzameling van plug-ins die met elkaar samenwerken. Het is mogelijk om extra plug-ins te installeren op Eclipse. Dat zorgt ervoor dat het platform heel flexibel is. Tijdens deze opdracht heb ik dat ook gemerkt, want er was software beschikbaar die ik op Eclipse heb geïnstalleerd waardoor ik kon ontwikkelen voor Lotus Notes. Die software is de Lotus Expeditor Toolkit. Verder biedt Eclipse tools voor grafische Java ontwikkeling, die heb ik ook gebruikt en in het stuk over ontwikkelen van een plug-in met Eclipse zal ik daar verder op ingaan.

De kracht dat Eclipse zoveel ondersteuning biedt is meteen ook de zwakte, want Eclipse is langzaam. Het opstarten en compilen gaat niet snel en het is ook te merken aan Lotus Notes, want dat platform is ook niet snel.

Leren werken met niet-relatieve databases

Lotus Notes en dus ook CSP werkt met niet-relatieve databases. Mijn ervaring ligt bij relationele databases, omdat daar tijdens de opleiding aandacht aan wordt besteed. Hoe Lotus Notes werkt met databases is een andere manier van denken dan wat ik heb geleerd, dus daar lag een uitdaging voor mij. Wel werd ik een aantal keer geconfronteerd met het feit dat ik nog "relationeel dacht".

Een logische vraag na deze ervaring met niet-relatieve databases is: wat is beter, een relationele database of een niet-relatieve database? Voordat ik die vraag kan beantwoorden is het goed om kort uiteen te zetten wat elke database is.

Een relationele database heeft tabellen met daarin records waarin informatie staat. Records uit een tabel kunnen worden verbonden met records uit een andere tabel. Dan moet in de tabel een kolom worden toegevoegd waarin per record een unieke sleutel opgenomen is. Aan de hand van identieke sleutels kunnen records uit verschillende tabellen aan elkaar gerelateerd worden.

Niet-relationele databases zijn opgebouwd uit documenten, bij Lotus Notes heten die notes, die geen relatie met elkaar hebben. Het databasemodel biedt niet zoals bij relationele databases de mogelijkheid om relatie aan te brengen op basis van unieke sleutels.

Mijn ervaring tijdens het afstuderen is dat voor elke situatie een andere soort database geschikt is. Bij het werken met relationele databases moeten in de software queries worden opgenomen die informatie ophalen uit de database. In niet-relationele databases moet de software een document uit de database ophalen en moeten acties geprogrammeerd worden om de bijbehorende informatie in de database te zoeken.

Als gewerkt wordt met een relationele database is het complexer om informatie op te halen, maar als ontwikkelaar kun je wel bepalen welke informatie je precies ophaalt. Gegevens ophalen uit een niet-relationele database is gemakkelijker, want er hoeft alleen een document opgehaald te worden, in dat document staat alle informatie.

Naar mijn mening is een niet-relationele database bij CSP beter dan een relationele. Dat komt, omdat CSP voor accountants en belastingadviseurs is ontwikkeld. Die doelgroep heeft een systeem nodig waar per klant of persoon veel ongestructureerde informatie moet worden opgeslagen. Het moet ook mogelijk zijn om informatie over een klant of persoon snel beschikbaar te hebben. In dat geval is het complexer om elke keer queries te maken dan om een document op te halen met daarin alle informatie.

Bij relationele databases moet elk veld, record en tabel goed gedefinieerd zijn. Er moet worden aangegeven van welk type een veld is en hoe lang het is, relationele databases hebben dus een datamodel. Dat is bij niet-relationele databases anders, daar maakt het niet uit hoe lang een veld is of van welk type het is, het kan altijd worden toegevoegd aan een document. Er staat nergens voorgeschreven op welke manier informatie gestructureerd is. Niet-relationele databases hebben geen datamodel, het kan dus zijn dat in een database dezelfde veldnamen staan met daarin verschillende types.

Gevolg voor een ontwikkelaar die niet-relationele databases gebruikt is dat er altijd documenten aan de database kunnen worden toegevoegd, zonder dat er iets aan de code hoeft te veranderen. Als bij een relationele database data wordt toegevoegd bestaat de kans dat relaties tussen tabellen en/of sommige queries niet goed meer werken.

Tijdens het afstuderen merkte ik dat het voor Lotus Notes en CSP een groot voordeel is dat gewerkt wordt met een niet-relationele database. Voor mij als programmeur was het ook makkelijker programmeren, want ik hoefde geen queries

meer te maken, maar kon een document uit de database halen waar alle informatie in stond die ik op dat moment nodig had.

Als ik verder denk aan gebruikers van CSP kan ik me voorstellen dat zij ook voordeel hebben van een niet-relatieve database. Als er een document ingescand moet worden kan die zonder moeite aan een database worden toegevoegd. Daarna kunnen op dat document alle soorten bewerkingen en aanvullende kenmerken worden toegevoegd.

Leren ontwikkelen van een plug-in met Eclipse

Zoals gezegd heb ik voor het ontwikkelen van de plug-in de Lotus expeditor Toolkit op Eclipse geïnstalleerd. Met die Toolkit is het mogelijk een plug-in te ontwikkelen voor Lotus Notes. Daarnaast biedt Eclipse veel ondersteuning voor grafische Java ontwikkeling. Die ondersteuning wordt vooral geboden door een The Standard Widget Toolkit (SWT) van Eclipse.

De genoemde aspecten zorgden dat het opzetten van een plug-in voor Lotus Notes geen groot probleem was. Daarnaast is het mogelijk om op vergelijkbare manier plug-ins voor andere plekken te ontwikkelen dan de sidebar Lotus Notes. Dit leerdoel is dus behaald.

Ervaring opdoen met de programmeertaal Java

De programmeertaal Java was nieuw voor mij, omdat daar op school geen aandacht aan wordt besteed en ik had er zelf nog nooit iets mee ontwikkeld. Voor het afstuderen werd tegen me gezegd dat het verschil tussen C++ (met die taal heb ik wel ervaring) en Java niet heel groot is. Tijdens het afstuderen bleek dat ook wel, de syntax van Java lijkt veel op die van C++, daarnaast zijn beide talen objectgeoriënteerd.

Een specifiek verschil wat ik tegenkwam is dat Java geen expliciet geheugenbeheer heeft, waardoor objecten opgeruimd worden als die geen referenties meer heeft. Bij Java wordt een object dus opgeruimd als het niet meer gebruikt wordt, bij C++ moet een object door de ontwikkelaar opgeruimd worden.

10.1.2 Organisatorische doelen

Leren bepalen van een ontwikkelstrategie en methode

Voordat een project begint moet bepaald worden op welke manier het project aangepakt wordt. Bij softwareontwikkeling gebeurt dat vaak aan de hand van een ontwikkelstrategie en een ontwikkelmethode.

Het afstuderen was mijn eerste project dat ik individueel moest doen. Het bepalen van een aanpak voor een project had ik voor het afstuderen nog nooit individueel gedaan.

Bij aanvang van het afstuderen had ik kennis over ontwikkelstrategieën en methoden, dat maakte het bepalen van de aanpak van het project eenvoudiger. Daarnaast heb ik ook vaak bepaald welke strategie of methode geschikt is voor virtueel project, maar nooit voor een echt project.

Dat zorgde ervoor dat ik de aspecten die de keuze over de strategie en methode beïnvloeden grondiger heb onderzocht dan ik voorheen altijd deed.

Wat is tijdens het project heb ervaren is dat documentatie erg duidelijk moet zijn en dat het soms bijzonder lastig is om het beeld dat ik had van de plug-in op papier te zetten. Daarnaast moest de opdrachtgever de documentatie op dezelfde manier interpreteren als ik had opgeschreven. Om die reden heeft dit project ervoor gezorgd dat ik een voorstander ben van het opleveren van prototypes tijdens een project. Een prototype laat zien hoe het product eruit komt te zien. Daarbij is de kans op een andere interpretatie door de opdrachtgever kleiner dan bij documentatie alleen.

Leren doorlopen van alle ontwikkelfasen in één project

Tijdens de opleiding zijn er vaak projecten waarbij ik samen met medestudenten een deel van een ontwikkeltraject moest uitvoeren. Dit afstudeerproject is een project waarbij alle ontwikkelfasen doorlopen moeten worden, omdat het geen deel van een ander project is en de oplevering moet een werkend product zijn. Daarnaast voer ik dit project individueel uit.

Mijn ervaring tijdens het project was dat het heel belangrijk is dat de documentatie klopt. Aan het begin van het project is het verloop gemakkelijk te overzien, maar naarmate het project vordert is dat moeilijker. Vaak moest ik oude documenten doornemen om te kijken wat de afspraken waren met de opdrachtgever. Tijdens het afstuderen is er geen onenigheid met de opdrachtgever geweest over de verwachtingen van de producten. Dat kwam mede dankzij de documentatie, maar vooral door het prototype.

Verder bleek de keuze van de ontwikkelstrategie bij aanvang van het project heel erg belangrijk. Naar mijn mening was het plan duidelijk en realistisch, mede daardoor verliep het doorlopen van de ontwikkelfasen goed.

Leren zoeken naar eisen met opdrachtgever

Het vastleggen van de eisen voor het eindproduct is bijzonder belangrijk tijdens een project. In dit project heb ik samen met de opdrachtgever eisen bepaald en vastgelegd. Een voordeel was dat de opdrachtgever op dezelfde afdeling werkt als waar ik afstudeerde. Vragen of ideeën kon ik meteen aan hem voorleggen.

De eerste manier waarop ik eisen heb verworven was door interviews te houden met de opdrachtgever.

De tweede manier om eisen boven tafel te krijgen was het maken van een prototype. Het prototype was, in overleg met de opdrachtgever, niet precies geworden zoals in de eisen stond, maar zorgde wel voor nieuwe inzichten bij de opdrachtgever en voor nieuwe eisen tijdens increment twee.

Na de oplevering van het prototype was er een bevestiging van de opdrachtgever dat de voorgang van het project op de goede weg was. Dat bevestigde mijn beeld dat bij onzekere systeemeisen een prototype een uitstekend middel is om duidelijkheid te krijgen over eisen.

Leren een product in te voeren

Eerder in deze paragraaf heb ik aangegeven dat ik tijdens de opleiding verschillende fasen van een ontwikkelproces heb doorlopen. De fase van het invoeren van een product is echter nooit aan bod geweest. Tijdens dit project moet de plug-in die gemaakt wordt ook ingevoerd worden in CSP.

In het geval van de plug-in heeft de gebruiker zelf de keus of deze wordt geïnstalleerd. In CSP moet dus een plek zijn waar de gebruiker dat kan bepalen. Om die plek te bepalen heb ik geleerd om meer te denken vanuit de gebruiker. Tijdens het project kwam ik met het idee om een XML-file te maken die de gebruiker dan in de sidebar van Lotus Notes kan slepen waardoor de plug-in geïnstalleerd kan worden. Waar ik niet aan gedacht had was dat de gebruiker CSP voor zich heeft en niet buiten CSP gaat zoeken naar de plug-in. Het is dus belangrijk vanaf welke plek in CSP de plug-in te installeren is. Wat ik ook heb geleerd is dat het van groot belang is wat voor soort gebruiker gebruik gaat maken van een product. De doelgroep van CSP zijn over het algemeen geen technische mensen, dus moet de plug-in en de installatie daarvan laagdrempelig zijn.

10.2 Aanpak

In deze paragraaf wordt de aanpak van de afstudeeropdracht geëvalueerd. Er wordt stilgestaan bij keuzes die gemaakt zijn en of er van de planning afgeweken is.

10.2.1 Aanvang afstuderen

Afstudeerplek

Aan het eind van de opleiding Technische Informatica voeren studenten een afstudeeropdracht uit. Begin 2010 mocht ik van school beginnen met afstuderen. Daarom ben ik eind 2009 op zoek gegaan naar een afstudeerbedrijf. Na contact en gesprekken met verschillende bedrijven kwam ik bij AedifiComm terecht. De opdracht die AedifiComm bood leek mij erg interessant en geschikt. Dat kwam omdat er een product gemaakt moest worden dat nog niet bestond en waar nog weinig over bekend was. Daarnaast was het een vereiste dat het product ingevoerd kon worden na de afstudeerperiode. Dat betekent dat alle ontwikkelfasen doorlopen moesten worden.

Het bedrijf AedifiComm is een partner van IBM en heeft veel kennis in huis. Dat gaf mij het vertrouwen dat ik tijdens mijn afstudeerperiode genoeg ondersteuning kon verwachten als dat nodig was. Daarnaast voelde de sfeer informeel aan, wat ik als erg positief ervaarde.

Nu het afstuderen bijna afgelopen is kloppen de verwachtingen die ik over de opdracht had goed met de werkelijkheid. Alle ontwikkelfasen zijn doorlopen en ik heb de eerste versie van een product gemaakt. Tijdens de opdracht werkte ik met Lotus Notes en Eclipse. AedifiComm heeft veel ervaring met Lotus Notes, waardoor ik met vragen over dat platform bij hen terecht kon. Daarnaast had AedifiComm niet veel ervaring met Eclipse en het ontwikkelen van plug-ins voor Lotus Notes, dat gebied moest ik zelf onderzoeken. Daardoor leerde ik veel van AedifiComm en leerde ik veel door zelf te onderzoeken. Ook help ik AedifiComm door deze opdracht

aan kennis over Eclipse en het ontwikkelen van plug-ins voor Lotus Notes. Dat ervaar ik als zeer prettig.

Start afstuderen

Toen het afstuderen begon was het zaak om een plan van aanpak en een planning te maken. Daarnaast kreeg ik van AedifiComm verschillende cursussen om de omgeving waarin de opdracht plaatsvond te verkennen.

Het plan van aanpak gaat door op het afstudeerplan, dat voor aanvang van het afstuderen gemaakt was. De hoofdstukken projectopdracht en aanpak hadden de hoogste prioriteit bij het schrijven, omdat deze belangrijk zijn om aan te tonen dat de opdracht HBO-waardig is. Uit de projectopdracht moet duidelijk zijn dat de opdracht op HBO-niveau is. In het hoofdstuk aanpak staat op welke manieren die opdracht aangepakt kan worden en welke manier mij het meest geschikt lijkt.

Het schrijven van het hoofdstuk aanpak kostte de meeste tijd. Dat kwam, omdat ik de exacte definitie van een iteratie en een increment niet goed voor ogen had. Na wat research in boeken en op internet was dat probleem snel verholpen.

De planning speelt een grote rol bij het bepalen van het aantal fasen dat tijdens het project doorlopen moet worden. Bij aanvang van het afstuderen ben ik niet tegen problemen aangelopen.

10.2.2 Increment 1

Definitie

Tijdens de definitiefase heb ik onderzocht of er plug-ins waren die leken op de plug-in uit de opdrachtoomschrijving. De plug-in moest gemaakt worden voor het softwarepakket CSP, dat AedifiComm heeft ontwikkeld. De kans dat er al een plug-in bestaat die goed werkt op dat pakket leek in eerste instantie te verwaarlozen. Toch leek het me goed om dat onderzoek te doen, omdat CSP gebaseerd is op Lotus Notes en voor Lotus Notes bestaan al wel plug-ins. Daarnaast moest de plug-in in de sidebar van Lotus Notes, die ook in CSP aanwezig is, komen. Tijdens het zoeken bleek dat het maken van plug-ins voor Lotus Notes sidebar nog in de kinderschoenen staat, want er waren niet veel plug-ins te vinden. Dat is niet heel gek, omdat het pas sinds Lotus Notes versie 8 mogelijk is om plug-ins voor de sidebar van Lotus Notes te maken. Met die wetenschap werd me ook duidelijker waarom AedifiComm deze opdracht had gegeven. Het bedrijf heeft nog weinig kennis over het maken van plug-ins voor de Lotus Notes sidebar, terwijl de sidebar veel mogelijkheden biedt om CSP gebruikersvriendelijker te maken. Door deze opdracht wilde AedifiComm meer kennis krijgen over het maken van plug-ins voor de Lotus Notes sidebar. Het onderzoek leverde geen plug-in op die voldeed aan de eisen van de opdrachtgever, maar door het onderzoek naar bestaande plug-ins werd wel duidelijk op welke manier een plug-in voor de Lotus Notes sidebar gebouwd kan worden. Deze fase verliep zoals gepland.

Analyse

In de analysefase heb ik het programma van eisen gemaakt. Eisen verzamelen kan op verschillende manieren en het hangt van de opdracht af op welke manier het gebeurt. Deze opdracht is voor een softwarepakket dat bij verschillende bedrijven in gebruik is. Daardoor heb ik er voor gekozen om geen interviews te doen bij de

uiteindelijke gebruikers. Elke gebruiker werkt in een andere omgeving en houdt geen rekening met andere gebruikers, waardoor het generieke aspect niet van belang is bij gebruikers.

Er zijn wel interviews geweest met de opdrachtgever. Voor de interviews zijn geen complete vragenlijsten uitgedacht en op papier gezet. Dat leek mij niet nodig, omdat de opdrachtgever op dezelfde afdeling zit als waar ik afstudeerde. Als er een vraag in me opkwam kon ik die meteen stellen. Dat betekent niet dat ik niet heb nagedacht over welke eisen de plug-in moet hebben, maar ik kon mijn vragen meteen stellen en hoefde ze niet op papier te verzamelen tot ik de opdrachtgever weer zag.

Tijdens de eerste analysefase heb ik de eisen voor heel het project verzameld, niet alleen voor increment één. Dat leek me het beste, omdat dan de werkzaamheden aan de hand van de eisen over twee incrementen verdeeld konden worden. Als ik alleen de eisen van increment één zou verzamelen dan kon ik geen inschatting maken welke werkzaamheden er in increment twee zouden zijn. Dan zou er geen betrouwbare planning gemaakt kunnen worden. Tijdens de analysefase was er geen vertraging.

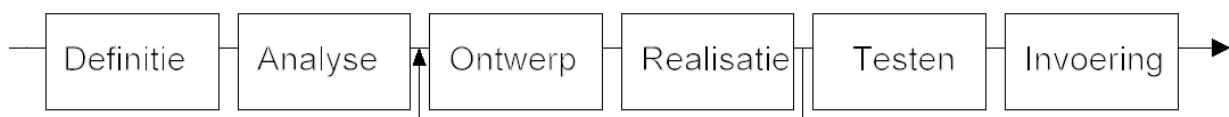
Ontwerp en implementatie

AedifiComm heeft weinig ervaring met het ontwikkelen van een plug-in voor de sidebar van Lotus Notes. Daardoor was het tijdens de ontwerpfase nodig om uit te zoeken welke platformen nodig waren om een plug-in te maken voor Lotus Notes. Daar was informatie over te vinden op internet, dus dat leverde geen problemen op. Mijn verwachting van de ontwerp- en implementatiefase was dat deze goed gescheiden konden blijven. De werkelijkheid was anders, want om een plug-in voor Lotus Notes te maken moet gebruik gemaakt worden van Eclipse en Eclipse heeft een platform API specification. Die API is een verzameling van klassen die gebruikt kunnen worden tijdens het programmeren in Eclipse. Zonder kennis over welke klassen in de API bestaan kon ik geen ontwerp maken. Het gevaar is dan dat er een ontwerp gemaakt wordt dat niet aansluit op Lotus Notes, dat gebaseerd is op Eclipse.

Het gevolg was dat ik de ontwerp- en implementatiefase iteratief doorlopen heb. Tijdens het ontwerp zocht ik op welke klassen en functies uit de API van Eclipse gebruikt konden worden. Om te testen of die klassen en functies ook echt werkten zoals gewenst implementeerde ik een deel van de plug-in.

Een andere keus had kunnen zijn om een ontwerp te maken aan de hand van theorie over de API van Eclipse. Dat leek mij geen goede keus, omdat er soms verschillende klassen voor hetzelfde doel gebruikt konden worden. Om dan te bepalen welke manier het beste is het goed om te testen. Afbeelding 8.1 geeft weer hoe increment één werkelijk verlopen is.

Increment 1:



Afbeelding 8.1: werkelijke verloop van increment één

Testen

Het prototype is na de implementatiefase is getest. Het leek mij goed om dat te doen, want daardoor wordt duidelijk wat het prototype wel en niet kan. Met die informatie wordt daarna bepaald wat er tijdens het volgende increment toegevoegd of verbeterd moet worden. Software kan getest worden volgens verschillende testmethodes. Bij het maken van het testplan van increment één was mijn doel om te testen aan welke eisen voldaan was en aan welke eisen niet voldaan was. Die manier van testen lijkt, zoals te lezen is in [Visser09], veel op blackbox testen. Nadeel van deze methode is dat input en output van functies in de software niet getest worden. Die testen worden ook wel whitebox testen genoemd. Voor whitebox testen is niet gekozen, omdat die testen geen rekening houden met de eisen. Behalve de implementatie kunnen de use-cases uit de analysefase en het ontwerp getest worden. Opleveringen uit de analyse en de ontwerpfase zijn niet volgens een methode getest, omdat dit een klein project is. Bij grotere projecten is het gebruikelijk om de eisen en het ontwerp te testen volgens een methode. Bij dit project is zijn die testen verband met tijd niet gedaan. Daarnaast heb ik ook geen kennis over het testen van eisen en het ontwerp, dus zou ik eerst kennis moeten opdoen over die testen.

10.2.3 Increment 2

Analyse

De analysefase van increment twee ben ik ingegaan met het prototype uit increment één. De interviews die gebruikelijk zijn om eisen boven water te krijgen zijn ook in deze fase gebruikt.

Het beeld in increment één was dat de gebruiker een gegevensbron in de plug-in kon slepen, zodat deze te zien was. Het prototype had de functie van het slepen nog niet, maar kon wel gegevens van een geselecteerde gegevensbron ophalen. Om het voor mezelf gemakkelijk te maken had ik het prototype zo gemaakt dat deze bij elke gegevensbron de gegevens ophaalde. Toen de opdrachtgever dat zag kwam hij met het idee om de functie van het slepen te vergeten en altijd elke geselecteerde gegevensbron weer te geven in de plug-in. Als de gebruiker dan een gegevensbron vast wilde zetten in de plug-in kon dat. Op die manier zorgde een functie die eigenlijk niet in de plug-in moest komen voor een idee bij de opdrachtgever.

Een minder positief punt was dat het erg lastig is om meerdere dezelfde plug-ins te installeren in de sidebar van Lotus Notes. Omdat het werk aan die eis niet alle tijd in beslag moest nemen hebben de opdrachtgever en ik besloten de prioriteit van die eis lager te maken.

Achteraf gezien zijn de eisen die tijdens deze fase naar voren zijn gekomen duidelijk en is er niets vergeten. De fase liep verliep volgens planning.

Ontwerp en implementatie

Het verloop van de ontwerp en implementatiefase ging ook in increment twee iteratief. Verschil met increment één was dat er een stuk minder iteraties doorlopen zijn.

Het ontwerp van increment twee was een stuk completer dan het ontwerp van increment één. Dat komt, omdat ik tijdens increment twee meer kennis had van de geschikte klassen in Eclipse voor de plug-in.

Een moeilijkheid tijdens het ontwerp en de implementatie was dat er veel gebeurd "onder de motorkap" van Eclipse. Bijvoorbeeld de TableViewer bevat een klasse die de inhoud van de plug-in verzorgt. Om input in de plug-in te krijgen moet een functie van de TableViewer worden aangeroepen. Eclipse zorgt er dan voor dat er een functie wordt aangeroepen in de klasse die de inhoud van de plug-in verzorgt. Het kostte veel tijd om uit te zoeken wat er in Eclipse gebeurde, terwijl, als we alleen naar de plug-in kijken, het er nu niet complex uit ziet.

Uiteindelijk heeft het project tijdens de ontwerp en de implementatiefase een week vertraging opgelopen. Dat is te wijten aan twee oorzaken. De eerste is dat tijdens deze fasen de prioriteit van een eis is aangepast. Dat kwam, omdat de informatie over de geselecteerde gegevensbron niet goed paste in de plug-in. De eis dat gebruikers kolommen kunnen verbergen en zichtbaar kunnen maken heeft toen hogere prioriteit gekregen. De tweede oorzaak is dat er veel feestdagen waren tijdens de ontwerp en implementatiefase, daar was in de planning geen rekening mee gehouden.

Testen

Ook tijdens increment is de plug-in getest. Als basis voor de testen is het PvE uit de analysefase genomen. Elke eis is afzonderlijk getest op Windows XP, Windows Vista en Windows 7. Dat gaf een positief resultaat.

Voor de manier van testen heb ik er voor gekozen om geen combinaties van eisen te testen, want dat kostte naar mijn mening teveel tijd.

Invoer

De werkelijke invoer van het product heb ik niet meegemaakt, omdat de plug-in in een van de volgende versies van CSP komt. Er is al wel bepaald hoe de plug-in ingevoerd moet worden. Lotus Notes biedt veel ondersteuning voor plug-ins en ook voor de manier waarop deze geïnstalleerd kunnen worden. Dat zorgde ervoor dat we tijdens dit project al een manier konden bepalen waarop de plug-in wordt ingevoerd.

10.3 Het project

Aan het einde van dit afstudeerproject is duidelijk dat het project geslaagd is. AedifiComm heeft mij goede ondersteuning gegeven en ik heb kennis overgedragen over Eclipse en het ontwikkelen van plug-ins voor Lotus Notes.

AedifiComm is van plan om na grondige testen de plug-in in te voeren in CSP. Dat feit is een bevestiging dat het project gelukt is en een bekroning op mijn werk tijdens het afstuderen. Naast het invoeren van de plug-in gaat AedifiComm ook nieuwe functies voor de plug-in ontwikkelen.

Tijdens het project heb ik het contact met mijn collega's als zeer positief ervaren. Als ik vragen had was er altijd interesse en namen de mensen veel tijd voor me. Verder was er naast het werk ook tijd voor informele omgang met elkaar. Dit alles

heeft ervoor gezorgd dat AedifiComm mij een contract aanbiedt om bij hen te komen werken en ik heb daar mijn handtekening onder gezet. Daaruit blijkt wel dat de samenwerking tussen AedifiComm en mij uitstekend is!

Mijn werkzaamheden zullen betrekking hebben op Lotus Notes en Exact. Praktisch zal ik als consultant optreden om bedrijven adviezen te geven over een totaaloplossing. Die oplossing ontwerp en implementeer ik dan zelf. AedifiComm, haar werknemers en mijn werkzaamheden zorgen ervoor dat ik met veel enthousiasme vooruit kijk naar mijn toekomst als software engineer.

11 Geraadpleegde literatuur

11.1 Boeken

[Dijkgraaf07] Dijkgraaf, W. en van Spall M., begin bij het eind, met SMART requirements, 1^e druk, Chaam, 2007.

[Larman04] Larman, C., Applying UML and patterns, and introduction to Object-Oriented analysis and design and iterative development. 3^e druk, Upper Saddle River, 2004. pp. 18-40

11.2 Websites

[1OpenNTF1002] OpenNTF, www.openntf.org/Internal/home.nsf (10 februari 2010)

[2Openntf1002] Openntf, <http://www.openntf.org/catalogs/a2cat.nsf/topicThread.xsp?action=openDocument&documentId=2DFAACEB82DE7876852576480020247F> (10 februari 2010)

[3Openntf1002] Openntf, <http://www.openntf.org/blogs/openntf.nsf/d6plinks/NHEF-7Y7AGX> (10 februari 2010)

[4Openntf1002] Openntf, <http://www.openntf.org/blogs/openntf.nsf/D6Plinks/NHEF-7YQBAQ> (10 februari 2010)

[Zwart1002] Sander Zwart, <http://www.socialsoftwareblog.nl/blog.nsf/dx/plugin-voor-lotus-notes.htm> (10 februari 2010)

[AedifiComm0902] AedifiComm, <http://www.aedificomm.nl/over-ons.aspx> (09 februari 2010)

[AedifiComm1102] AedifiComm, <http://www.csp.nu/over-csp.aspx> (11 februari 2010)

[Grit0902] Roel Grit, http://www.grit-projectmanagement.wolters.nl/7003_projectmanagement/_assets/7003d04.pdf (09 februari 2010)

In bijlage 17 is een pdf file van deze pagina opgenomen

[IBM0803] IBM, <http://www.ibm.com/developerworks/lotus/library/expeditor-toolkit/> (8 maart 2010)

In bijlage 18 is een pdf file van deze pagina opgenomen

[IBM0904] IBM, <http://www.ibm.com/developerworks/lotus/library/notes8-new/> (9 april 2010)

[Lotus1002] Lotus, <http://www-10.lotus.com/ldd/notestipsblog.nsf/dx/quickr-sidebar-plug-in?opendocument&comments> (10 februari 2010)

[Socialtext1002] Socialtext, www.socialtext.net/lotus/index.cgi?lotus_notes_sidebar_apps_and_plugins (10 februari 2010)

[TLCC1102] TLCC, <http://www.tlcc.com/> (11 februari 2010)

[Vogella1005] Vogella, <http://www.vogella.de/articles/EclipseJFaceTable/ar01s05.html> (10 mei 2010)

[Eclipse0705] Eclipse, https://bugs.eclipse.org/bugs/show_bug.cgi?id=17871 (7 mei 2010)

11.3 Overige

[Bosch07] Van den Bosch, H.C., *Sheets Systeemontwikkeling.pdf*, 2007

[Noordtzij04] Noordtzij, E., *Rapporteren, een handreiking voor het schrijven van rapporten*, 2004

[Visser09] Visser, J., *Testen, Embedded Software*, 2009

1 Begrippenlijst

| Begrip | Uitleg |
|--------------------|--|
| CSP | Softwarepakket van AedifiComm dat is gebaseerd op IBM Lotus Notes en Domino |
| Ontwikkelfasen | Definitie, analyse, ontwerpen, realisatie, testen en invoering |
| Ontwikkelstrategie | Manier waarmee een ontwikkelproces kan worden aangepakt, met globale indeling van ontwikkelfasen in tijd |
| Ontwikkelmethode | Een vaste manier van het ontwikkelproces om een resultaat te behalen |
| Back-end | Functies in de plug-in die niet zichtbaar zijn voor de gebruiker |

2 Afstudeerplan

Afstudeerplan

Informatie afstudeerder en gastbedrijf (*structuur niet wijzigen*)

Afstudeerblok: 2010-1.1
Startdatum: 8 februari 2010
Einddatum: 5 juni 2010
Inleverdatum: 4 juni 2010

Studentnummer: 60242

Achternaam: dhr. den Arend

(*) weghalen niet van toepassing

Voorletters: J.C.

Roepnaam: Chris

Adres:

Postcode:

Woonplaats:

Telefoon:

Mobiel:

Opleiding:

Locatie: Delft

(*) weghalen niet van toepassing

Variant: voltijd

Naam studieloopbaanbegeleider: Dave Stikkolorum

Naam begeleider/examinator: Dave Stikkolorum

Naam expert/examinator: Adri Pronk

Naam bedrijf: AedifiComm

Afdeling bedrijf: Softwareontwikkeling

Bezoekadres bedrijf: Londen 12

Postcode bezoekadres: 2993LA

Postbusnummer: Postbus 19

Postcode postbusnummer: 2990 AA

Plaats: Barendrecht

Telefoon bedrijf: 0180 555 111

Telefax bedrijf: 0180 555 115

Internetsite bedrijf: <http://www.aedificomm.com/>

Achternaam opdrachtgever: dhr. Spruit

(*) weghalen niet van toepassing

Voorletters opdrachtgever: F.J.C.

Titulatuur opdrachtgever: :ir.

Functie opdrachtgever: directeur

Doorkiesnummer opdrachtgever: 0180 555 111

Email opdrachtgever: fjc@aedificomm.com

Achternaam bedrijfsmentor: dhr Scheppink

(*) weghalen niet van toepassing

Voorletters bedrijfsmentor: N.G.

Titulatuur bedrijfsmentor: ing.

Functie bedrijfsmentor: Project manager

Doorkiesnummer bedrijfsmentor: 0180 555 111

Email bedrijfsmentor: nico.scheppink@aedificomm.com

Doorkiesnummer afstudeerder: 0180 555 111

Bedrijfsemail afstudeerder: chris.den.arend@aedificomm.com

Opdrachtomschrijving

Titel afstudeeropdracht: Ontwikkelen van generieke sidebar plug-ins voor IBM Lotus Notes & Domino

1. Bedrijf:

AedifiComm

2. Aanleiding:

AedifiComm ontwikkelt en implementeert volledig in eigen beheer een aantal standaard softwarepakketten, waaronder CSP. CSP is een integrale oplossing die de bedrijfsprocessen van financiële dienstverleners (accountants en belastingadviseurs) ondersteunt. CSP is gebaseerd op het IBM Lotus Notes & Domino platform. De meest recente releases van dit platform bieden geavanceerde mogelijkheden voor het toevoegen van plug-ins, die ontwikkeld worden met behulp van JAVA. AedifiComm ziet diverse toepassingen voor deze nieuwe mogelijkheden.

3. Probleemstelling:

Het softwarepakket CSP bestaat uit diverse modules en het is lastig om de meest relevante informatie ten allen tijde zichtbaar te maken voor de eindgebruiker. Hoe kan de sidebar van de applicatie worden ingezet om door middel van een configureerbare en generieke plug-in de gewenste informatie zichtbaar te maken?

4. Doelstelling van de afstudeeropdracht:

Onderzoeken hoe de sidebar van de applicatie kan worden ingezet om door middel van een configureerbare en generieke plug-in de gewenste informatie zichtbaar maakt. De eisen voor de plug-in liggen nog niet vast, dus het eerste deel van de opdracht is het vastleggen van de eisen voor de te maken plug-in. Deze eisen worden intern bepaald. Uiteindelijk wordt de plug-in gebruikt in CSP dat door ongeveer 30 bedrijven, die tussen twee en 600 medewerkers hebben, wordt gebruikt. Na het vastleggen van de eisen wordt er een functioneel ontwerp van de plug-in gemaakt. Bij dit ontwerp wordt rekening worden gehouden met uitbreidingen op dat model, bijvoorbeeld niet alleen het weergeven van de sidebar, maar ook het bewerken ervan. Daarnaast wordt er een sidebar plug-in gemaakt die overal in de suite van het bedrijf gebruikt/geplaatst kan worden. De moeilijkheid van de plug-in is dat AedifiComm werkt met een softwarepakket met 20 verschillende modules, de plug-in moet met elke module goed functioneren. Daarbij is het belangrijk dat de performance van de plug-in in elke omgeving en in elke toestand hoog is. De synchronisatie tussen de plug-in en samenwerkende modules moet soepel verlopen.

Een module uit CSP moet gesleept kunnen worden in de sidebar. De gebruiker moet dan kunnen selecteren welke informatie van belang is. Bijvoorbeeld alleen zijn of haar gerelateerde informatie, of informatie over alle medewerkers. De plug-in moet zo generiek zijn dat een nieuwe module in CSP ook meteen te gebruiken is met de plug-in.

5. Resultaat:

De eisen voor de plug-in vastleggen, dat is het eerste resultaat. Met de eisen wordt een functioneel ontwerp van de plug-in gemaakt. Als de eisen en het ontwerp gemaakt zijn wordt

een werkend prototype in een demonstratie opstelling gemaakt. Het prototype moet gewenste informatie zichtbaar maken en moet werken in verschillende omgevingen van CSP. De informatie wordt verzameld uit verschillende omgevingen en databases, van de klant. Sommige klanten hebben ongeveer 10.000 databases. De gebruiker kan de plug-in gebruiken, waardoor zijn of haar gewenste informatie zichtbaar wordt. Het prototype moet geplaatst worden in een demonstratieopstelling van het softwarepakket CSP. Het prototype moet werken in het IBM Lotus Notes & Domino platform, versie 8.5. Deze versie is in oktober 2009 op de markt gebracht.

Bij het resultaat hoort ook gedocumenteerde programmacode en documentatie over het implementeren van de plug-in.

6. Uitgangssituatie:

Een demonstratieopstelling van het softwarepakket CSP wordt ter beschikking gesteld. De plug-in zal ontwikkeld worden aan de hand van concrete business cases en wensen t.a.v. het softwarepakket CSP.

Er zijn diverse online resources over het ontwikkelen en toepassen van plug-ins. Op de site van IBM is veel documentatie beschikbaar over het IBM Lotus Notes & Domino platform.

Het is essentieel om de programmeertaal Java en het ontwikkelplatform Eclipse te beheersen.

7. Werkzaamheden:

Analyseren: probleemstelling analyseren

Modelleren: modellen maken via UML

Implementeren: programmeren in object oriented programming (Java)

Testen: testplan maken om de kwaliteit van de software te waarborgen.

Uit te voeren activiteiten: oriënteren, modelleren, programmeren, testen, documenteren en implementeren

8. Risico's en maatregelen

Er bestaat al een sidebar plug-in voor IBM Lotus Notes & Domino platform, versie 8.5.

Dit risico is reeds onderzocht door de student en het bedrijf. Allereerst is de versie van het IBM Lotus Notes & Domino platform waarmee gewerkt wordt recent, dus daar is nog weinig voor gemaakt. Daarnaast gebruikt AedifiComm het IBM Lotus Notes & Domino platform als basis, het bedrijf heeft zelf veel software geschreven, waardoor unieke omgevingen zijn ontstaan waar de plug-in moet komen.

De student en AedifiComm hebben kort gezocht naar een sidebar plug-in die toegepast kan worden binnen AedifiComm, maar niet gevonden.

Het maken van de opdracht kost minder dan 17 weken

De kans hierop is klein. De hoogste prioriteit is het weergeven van de sidebar plug-in. Als dit sneller klaar is dan 17 weken, dan moet de plug-in zo uitgebreid worden dat de gebruiker de sidebar ook kan bewerken.

Het maken van de opdracht kost meer dan 17 weken

Er is een risico dat het opzetten van het programma van eisen of het maken van het functioneel ontwerp meer werk in beslag neemt dan gepland. Dan worden er minder iteraties doorlopen.

9. Op te leveren (tussen)producten

- Plan van aanpak

- Planning
- Programma van eisen
- Voortgangsverslag (7e/8e week)
- Functioneel ontwerp
- Werkend prototype, zoals beschreven in het resultaat (punt 5)
- Gedocumenteerde programmacode
- Documentatie over het implementeren van de plug-in
- Eindpresentatie
- Afstudeerdossier (4 juni)

Benodigde competenties

- Inventariseren eisen en wensen en vertalen naar een systeemontwerp
- Programmeren
- Werken in een team
- Basis van object oriented programming in Java
- Werken met IBM Lotus Notes en IBM Domino Designer (gebaseerd op Eclipse)
- Testen
- Documenteren

Te demonstreren competenties en wijze waarop

- Systeemontwerp door oplevering van een functioneel / technisch ontwerp
- Programmeren, door oplevering prototype
- Planning voor project maken en uitvoeren, door eindverslag
- Beargumenteren van keuzes tijdens het project, door eindverslag
- Juiste methode en technieken gebruiken tijdens project, door eindverslag en prototype
- Testen door testplan en testverslag
- Documenteren door systeemdokumentatie en testverslag

Persoonlijke verantwoording van keuze opdracht/bedrijf/competenties**Bedrijf:**

Aedificomm is een professioneel bedrijf met veel kennis in huis. Daardoor is er goede begeleiding mogelijk tijdens het afstuderen. Het bedrijf heeft ook ervaring met afstudeerders en verwacht van de student altijd 100% inzet.

Opdracht:

De opdracht geeft de mogelijkheid om te doen wat ik leuk vind, namelijk programmeren. Binnen de opdracht is het mogelijk om software te analyseren, modelleren, implementeren en te testen.

Competenties:

Tijdens de opdracht leer ik een nieuwe programmeertaal, Java. Het is belangrijk om kennis van deze taal te hebben, omdat hij veel gebruikt wordt.

Daarnaast vind ik het leuk om met een nieuwe opdracht te beginnen en resultaat neer te zetten.

Nominale duur afstudeerperiode

16 weken

Activiteitenplanning

8 februari: begin afstuderen

week 8/9: bedrijfsbezoek van begeleider

week 12/13: inleveren voortgangsverslag

week 14/15: bespreken concept afstudeerdossier

week 19/20: tussentijds assessment

4 juni: inleverdatum afstudeerdossier

Naam opdrachtgever:
Handtekening voor akkoord:
Datum:

Naam bedrijfsmentor:
Handtekening voor akkoord
Datum:

Naam begeleider/examinator:
Handtekening voor akkoord:
Datum:

Naam expert/examinator:
Handtekening voor akkoord:
Datum:

3 Plan van Aanpak

Plan van Aanpak

Generieke sidebar plug-in
voor IBM Lotus Notes & Domino

AedifiComm

| | |
|---------------------|--------------------------------|
| Opdrachtgever: | F.J.C. Spruit |
| Bedrijfsmentor: | N.G. Scheppink |
| Begeleider: | D.R. Stikkolorum |
| Expert: | A.G.P. Pronk |
| Afstudeerder: | Chris den Arend |
| Email afstudeerder: | Chris.den.Arend@aedificomm.com |
| Plaats: | Barendrecht |
| Datum: | 05-03-2010 |
| Versie: | 2.1 |



Inhoudsopgave

| | |
|---|----|
| Inleiding | 4 |
| 1 Achtergrond | 5 |
| 2 Projectopdracht | 7 |
| 2.1 Opdrachtgever en opdrachtnemer | 7 |
| 2.2 Probleemstelling | 7 |
| 2.3 Doelstelling | 7 |
| 2.4 Opdrachtbeschrijving | 7 |
| 2.5 Op te leveren producten | 9 |
| 2.6 Projectgrenzen | 9 |
| 2.7 Randvoorwaarden | 10 |
| 3 Aanpak | 11 |
| 3.1 Ontwikkelstrategie afstudeerproject | 14 |
| 3.1.1 Lineaire strategie | 11 |
| 3.1.2 Iteratieve strategie | 12 |
| 3.1.3 Incrementele strategie | 13 |
| 3.1.4 Ongeschikte strategieën | 14 |
| 3.1.5 Geschikte strategie | 14 |
| 3.2 Werkzaamheden | 19 |
| 3.2.1 Definitiefase | 15 |
| 3.2.2 Increment 1 | 15 |
| 3.2.3 Increment 2 | 20 |
| 3.3 Schema's en tekentechnieken | 16 |
| 3.4 Planning | 20 |
| 4 Projectinrichting | 18 |
| 4.1 Informatievoorziening | 18 |
| 4.2 Faciliteiten | 18 |
| 4.3 Kwaliteit | 18 |
| 5 Kosten en baten | 19 |
| 5.1 Kosten | 19 |
| 5.2 Baten | 19 |
| 6 Risicofactoren | 20 |
| 6.1 Afwezigheid afstudeerder | 20 |
| 6.2 Miscommunicatie | 20 |
| 6.2.1 School | 20 |
| 6.2.2 Bedrijf | 20 |
| 6.3 Verkeerde ontwikkelstrategie | 20 |
| 6.4 Beperking IBM platform | 20 |
| Bronnen | 21 |
| 1. Boeken | 21 |
| Bijlagen | 21 |
| 1. Begrippenlijst | 1 |

| | |
|-------------------|----|
| 2. Planning | 21 |
|-------------------|----|

Inleiding

AedifiComm ontwikkelt software en biedt studenten de mogelijkheid bij hun af te studeren. Een van de softwarepakketten die AedifiComm ontwikkelt is CSP. Dat pakket is gebaseerd op Lotus Notes en Domino. De Lotus Notes software heeft een sidebar, nu wil AedifiComm die sidebar gebruiken om daar een plug-in in te installeren. Die plug-in moet gemaakt worden. In dit document wordt beschreven hoe de aanpak is van het project: ontwikkelen van generieke sidebar plug-in voor IBM Lotus Notes. De beschrijving van de aanpak is belangrijk, omdat daarmee gecontroleerd wordt of de indeling van het project goed is. Dit document is gemaakt voor de opdrachtgever.

Als eerste worden de achtergrond van het project en het bedrijf beschreven. Dan wordt uitgelegd wat de opdracht is en wat AedifiComm met de opdracht wil bereiken. In hoofdstuk drie staat hoe de opdracht aangepakt wordt, o.a. met welke methode. Het hoofdstuk projectinrichting gaat over de plek die iedereen binnen het project heeft, de kwaliteit die gewaarborgd moet worden en de kosten en baten. Het laatste hoofdstuk gaat over de mogelijke risico's in het project en de maatregelen daartegen. De opbouw van dit plan is gemaakt aan de hand van Noordtzij (2007) en Roel Grit, http://www.grit-projectmanagement.wolters.nl/7003_projectmanagement/_assets/7003d04.pdf (09 februari 2010).

In bijlage 1 staat een tabel met begrippen en de uitleg daarvan.

1 Achtergrond

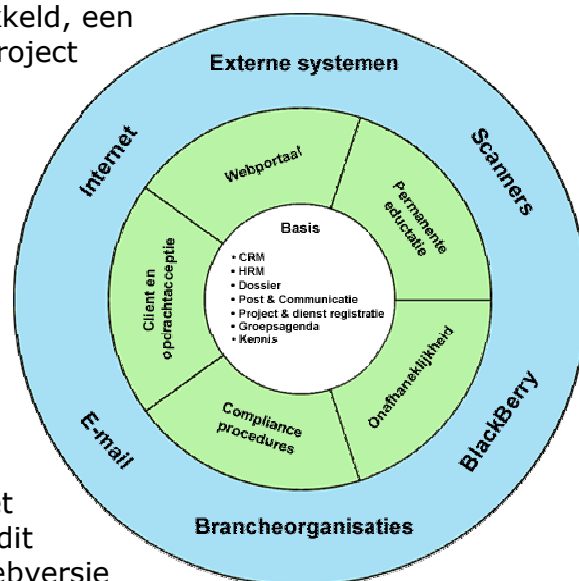
De afstudeeropdracht wordt uitgevoerd bij het bedrijf AedifiComm. Dit bedrijf is gevestigd in Barendrecht. AedifiComm is een onafhankelijke leverancier van diensten en producten op het gebied van kennismanagement, workflowmanagement, messaging-technologie, real time communication en collaboration oplossingen. Binnen het bedrijf wordt gebruik gemaakt van oplossingen van IBM (Lotus, Websphere) en Microsoft (sharepoint en .NET). De focus van het bedrijf is op kennis en software-projecten. Samen met de zusterorganisatie NedPortal (netwerkinfrastructuur) is AedifiComm in staat oplossingen te ontwerpen, implementeren en ook te beheren. AedifiComm heeft in eigen beheer producten ontwikkeld, variërend van complete dossier-oplossingen tot tools voor Lotus Domino. Deze producten worden over de hele wereld geleverd. Het bedrijf verdeelt per project werkzaamheden over medewerkers. Het bedrijf heeft de afdeling secretariaat, administratie en softwareontwikkeling. Alle projecten die met software te maken hebben worden op de afdeling softwareontwikkeling gedaan, zo ook de uitvoering van het afstudeerproject.

AedifiComm heeft verschillende producten ontwikkeld, een daarvan is CSP voor accountants. Het afstudeerproject heeft betrekking op dat pakket.

CSP biedt ondersteuning voor de werkprocessen van accountants, en belastingadviseurs.

CSP is modulair opgebouwd, zoals te zien is in afbeelding 1.1.

De basismodule van CSP beschikt over verschillende functies, enkelen daarvan zijn CRM functionaliteit, HRM functionaliteit, elektronisch dossier, post en communicatie, persoonlijke agenda en groepsagenda. AedifiComm biedt ook een pakket met uitbreidingen op de basismodule. Daarnaast is het mogelijk om maatwerk op CSP te verkrijgen. Op dit moment is CSP een standalone applicatie, een webversie van CSP wordt ontwikkeld.



Afbeelding 1.1: modules van CSP

Verschiede ontwikkelaars zijn voortdurend bezig met het uitbreiden en verbeteren van CSP. Tot 2009 werd CSP aangeboden zonder onderhoudscontract. Gevolg daarvan was dat bij nieuwe of andere wensen van de klant maatwerk gedaan moest worden. Situatie was toen dat bij elke klant een eigen versie van CSP stond. In 2009 is AedifiComm begonnen met het leveren van een onderhoudscontract, het bedrijf heeft er toen voor gezorgd dat veel maatwerk dat bij verschillende klanten stond in het standaardpakket van CSP opgenomen werd. Alle klanten kregen vanaf toen een standaard versie van CSP.

Recentelijk is er een nieuwe versie van IBM Lotus, Notes en Domino op de markt gekomen. AedifiComm is met deze versie gaan werken. Deze nieuwe versie heeft veel functies die nog niet gebruikt worden door CSP. Tijdens deze opdracht wordt een van die ongebruikte functies gebruikt een nuttige toevoeging aan CSP.

2 Projectopdracht

In dit hoofdstuk wordt beschreven wie de opdrachtgever en opdrachtnemer zijn. Verder wordt duidelijk waarom dit project wordt uitgevoerd en wat het project is. Als laatste staat in dit hoofdstuk binnen welke grenzen en randvoorwaarden het project uitgevoerd wordt.

2.1 Opdrachtgever en opdrachtnemer

Het bedrijf dat de opdracht geeft is AedifiComm. De opdrachtgever is dhr. F.J.C. Spruit. De bedrijfsmentor van de afstudeerder is N.G. Scheppink(NIS). Tijdens het afstuderen treedt de bedrijfsmentor op namens opdrachtgever. De opdracht is aangenomen en wordt uitgevoerd door Chris den Arend, TI-student van de Haagse Hogeschool, vestiging Delft.

2.2 Probleemstelling

Het softwarepakket CSP bestaat uit diverse modules en het is lastig om de meest relevante informatie te allen tijde zichtbaar te maken voor de eindgebruiker. Elke module bevat informatie en voor elke gebruiker heeft relevante informatie een andere invulling.

Voor dit project is een structureel probleem geconstateerd. Namelijk dat relevante informatie niet op elk moment in CSP weergegeven wordt. Om informatie te vinden moet eerst genavigeerd worden naar de module waar de gezochte informatie in staat. Deze actie kost veel tijd en is ongemakkelijk.

De hoofdvraag van dit project is dus: hoe kan de sidebar van het IBM Lotus Notes en Domino platform worden ingezet om door middel van een configureerbare en generieke plug-in de gewenste informatie zichtbaar te maken?

Generiek betekent hier dat de plug-in met elke willekeurige module van CSP, dat zijn er nu 20, moet werken. Configureerbaar betekent dat de gebruiker kan bepalen welke informatie uit de gekozen module gewenst is.

2.3 Doelstelling

De doelstelling van het project is het uitbreiden en daarmee gebruikersvriendelijker maken van het softwarepakket CSP. Het vriendelijker maken van de software gebeurt op twee manieren. Als eerste kost het de gebruiker minder tijd voordat de gewenste informatie zichtbaar is. Het tweede is dat de gebruiker meer en vaker op de hoogte is doordat de relevante informatie te allen tijde zichtbaar is. Dit zorgt ervoor dat het werk efficiënter gedaan wordt. Werknemers kunnen eerder actie ondernemen als dat nodig is, waardoor de klant beter geholpen wordt.

Daarnaast is het doel van de opdrachtgever om meer kennis te krijgen over het ontwikkelen van een plug-in voor de sidebar van Lotus Notes.

2.4 Opdrachtbeschrijving

De opdracht bestaat uit verschillende delen. Het eerste deel is het vastleggen van de eisen voor de plug-in. De eisen komen voor een deel in dit document, maar voor een uitgebreide beschrijving van de eisen wordt een apart document gemaakt. Het

vastleggen van de eisen gebeurt met de MoSCoW methode. Daarnaast wordt beschreven waar de eisen vandaan komen.

Als de eisen vastgelegd zijn dan wordt het functioneel en technisch ontwerp gemaakt. Het functionele ontwerp beschrijft hoe de plug-in gaat werken. Het is goed mogelijk dat de opdrachtgever door het functioneel ontwerp wat aanpassingen heeft op de eisen. In dat geval worden de eisen aangepast. Het ontwerp wordt gemaakt in UML, door middel van een use-case-diagram een sequentiediagram en een klassediagram. Een eis aan het ontwerp is dat het uitgebreid moet kunnen worden.

Als laatste wordt het eindproduct ontwikkeld, de eerste versie van de plug-in. De plug-in moet voldoen aan het ontwerp en aan de eisen die vastgelegd zijn. Zoals gezegd staan alle eisen in een apart document, hier staan de meest abstracte en algemene eisen voor de plug-in. Achter elke eis staat bij wie de eis vandaan komt en wanneer deze kenbaar is gemaakt.

Functioneel

- De plug-in functioneert in de sidebar van Lotus Notes
NIS, schriftelijk bij de opdrachtschrijving
- De plug-in functioneert met alle modules binnen CSP
NIS, telefonisch bij de opdrachtschrijving
- Er kan een module van CSP in de plug-in gesleept worden
NIS, telefonisch bij de opdrachtschrijving
- De plug-in geeft alle geselecteerde informatie weer over de module die in de plug-in is gesleept
NIS, schriftelijk bij de opdrachtschrijving
- De gebruiker kan bepalen welke categorie informatie weergegeven wordt
NIS, telefonisch bij de opdrachtschrijving
- De plug-in is in het Nederlands
NIS, mondeling bij bespreken van het plan van aanpak
- De gebruiker kan instellen hoe vaak de plug-in refreshed wordt
NIS, mondeling bij bespreken van het plan van aanpak

Niet-functioneel:

- De plug-in werkt op het platform IBM Lotus Notes en Domino
NIS, schriftelijk bij de opdrachtschrijving
- De plug-in werkt in de standalone versie van CSP
NIS, schriftelijk bij de opdrachtschrijving
- De plug-in is zo generiek dat een nieuwe module in CSP ook functioneert met de plug-in
NIS, schriftelijk bij de opdrachtschrijving
- De informatie in de plug-in wordt sneller weergegeven dan wanneer naar diezelfde informatie genavigeerd wordt
NIS, mondeling bij bespreken van het plan van aanpak
- De lay-out moet passen in CSP
NIS, mondeling bij bespreken van het plan van aanpak

2.5 Op te leveren producten

Tijdens het project worden de volgende producten opgeleverd:

- Plan van Aanpak
- Planning
- Programma van Eisen
- Tussenverslag
- Ontwerp (technisch en functioneel)
- Eerste versie van de plug-in
- Gedocumenteerde programmacode
- Documentatie over het implementeren van de plug-in
- Documentatie over de invoer van de plug-in
- Eindpresentatie
- Afstudeerdossier (alle opgeleverde producten bij elkaar)

2.6 Projectgrenzen

De opdracht is dat er een eerste versie ontwikkeld wordt van een plug-in voor de sidebar van Lotus Notes. Dat houdt in dat aan het einde van dit project een werkend en getest product is. Dat product moet voldoen aan de eisen die als "Must have" staan gedefinieerd in het document met de eisen.

Het eindproduct moet generiek zijn. Dat betekent dat de plug-in met alle modules moet werken, ook met eventueel nieuwe modules in CSP. Om de plug-in te laten werken met modules die later gebouwd worden moeten er eisen zijn aan de modules binnen CSP. De plug-in heeft als doel informatie weergeven, die informatie kan in de plug-in niet aangepast worden. De gebruiker krijgt wel de mogelijkheid om voorwaarden te stellen aan de weer te geven informatie. Er kan bijvoorbeeld gekozen worden om alleen informatie uit een module weer te geven die gerelateerd is aan de gebruiker zelf. Een ander voorbeeld is de informatie die te maken heeft

met een bepaalde klant. Het project is geslaagd als het eindproduct werkend is in een demonstratieopstelling.

2.7 Randvoorwaarden

Binnen het project zijn er verschillende beperkende condities die vooraf gesteld zijn. De plug-in moet functioneren in de sidebar van CSP dat gebaseerd is op IBM Lotus Notes en Domino vanaf versie 8.5. Daarnaast moet de plug-in in de programmeertaal Java geschreven worden. Er is gekozen voor Java, omdat het bedrijf dat als eis stelt. Daarnaast is Java geïntegreerd in Lotus Notes en de ondersteuning voor Java is beter dan bijvoorbeeld C of C++. Ook zijn de meeste plug-ins voor de sidebar van Lotus Notes in Java geschreven, daaruit kan afgeleid worden dat die taal het best geschikt is voor een plug-in voor de sidebar van Lotus Notes.

3 Aanpak

In dit hoofdstuk wordt beschreven hoe het project aangepakt wordt. Daarbij wordt stilgestaan bij de manier waarop het product ontwikkeld wordt, de werkzaamheden die uitgevoerd worden en de globale planning van het project.

3.1 Ontwikkelstrategie afstudeerproject

Het project duurt niet lang, ongeveer vier maanden. De uitvoerder van het project is een student die afstudeert en het project wordt niet uitgevoerd in een projectgroep. Al deze aspecten wegen mee in de afweging welke ontwikkelstrategie gebruikt gaat worden.

Ontwikkelstrategieën zijn opgebouwd uit verschillende fasen. De verschillende fasen zijn:

- Definitie
- Analyse
- Ontwerp
- Realisatie
- Testen
- Invoering

Er zijn een aantal ontwikkelstrategieën mogelijk:

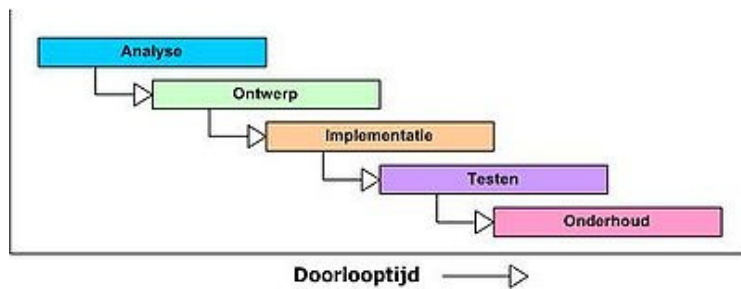
4. Lineaire ontwikkelstrategie
5. Iteratieve ontwikkelstrategie
6. Incrementele ontwikkelstrategie

3.1.1 Lineaire strategie

Bij de lineaire ontwikkelstrategie worden alle ontwikkelfasen achter elkaar doorlopen. Na het doorlopen van alle fasen is het product af. Deze strategie is goed toe te passen op kleine projecten waar de opdrachtgever de systeemeisen goed kan definiëren. Ook is het belangrijk dat de ontwikkelaars de systeemeisen goed begrijpen, dus ervaring van de ontwikkelaar is wenselijk. Het nadeel van de strategie is dat de opdrachtgever de eisen niet kan aanpassen na de definitie en analyse fase.

Watervalmethode

De watervalmethode maakt gebruik van de lineaire ontwikkelstrategie. Verschil is dat bij deze methode fasen kunnen overlappen, zie afbeelding 3.1. Vroeger werd deze methode veel gebruikt, ook voor grote projecten. Deze methode is niet geschikt voor projecten waar systeemeisen tijdens het project veranderen.



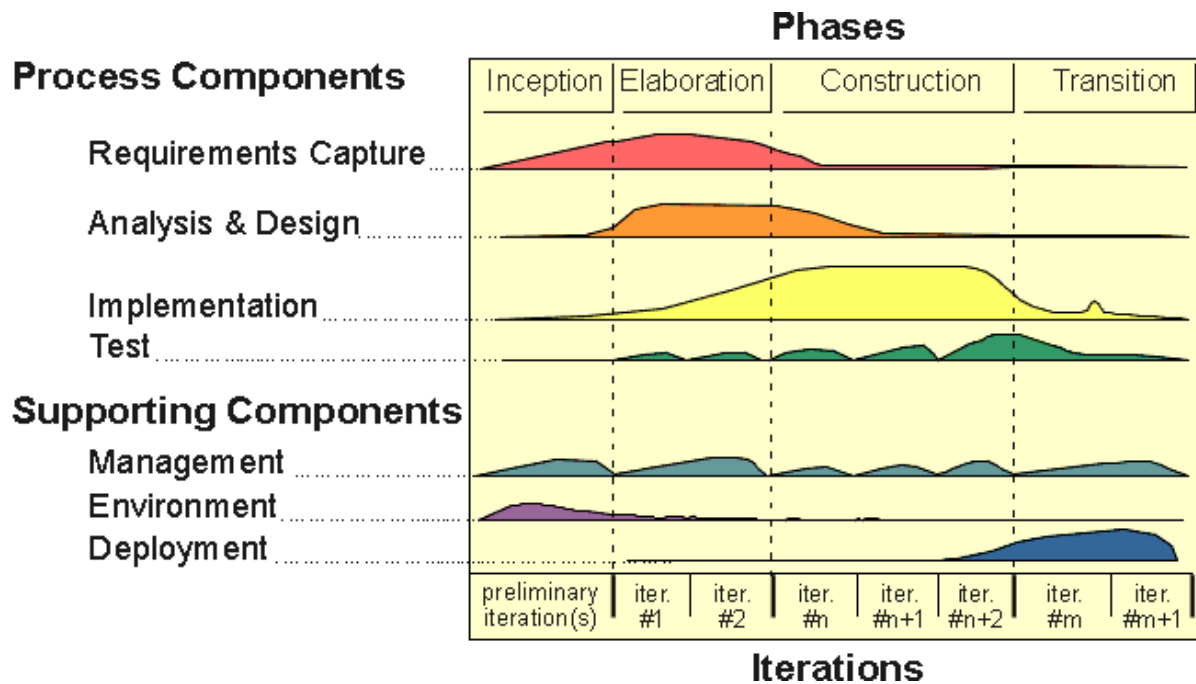
Afbeelding 3.1 de watervallmethode

3.1.2 Iteratieve strategie

Bij de iteratieve ontwikkelstrategie worden net als bij de lineaire strategie alle ontwikkelfasen doorlopen, maar hier kan teruggedaan worden naar een eerdere fase. Dat kan nodig zijn om een fase te verbeteren. Net zoals bij de lineaire strategie wordt het eindresultaat in een keer opgeleverd. Deze strategie wordt gebruikt bij grote projecten en projecten waar de probleemruimte niet duidelijk is. De nadelen zijn dat het proces moeilijker te beschrijven is en de oorspronkelijke doelstellingen kunnen vervagen.

Rational Unified Process (RUP)

Er zijn verschillende mogelijkheden als het gaat om incrementele en iteratieve ontwikkelmethoden. Een van de bekendste is RUP. Bij RUP wordt vastgehouden aan een aantal best practices. Een daarvan is het iteratief ontwikkelen, RUP kan worden ingericht als combinatie van incrementeel en iteratief ontwikkelen. Er kan ook voor gekozen worden om met RUP alleen iteratief te ontwikkelen. Tijdens een iteratie is het voor de opdrachtgever en gebruiker mogelijk om de probleemstelling en de eisen aan te passen. In elke iteratie zijn de ontwikkelfasen weer anders verdeeld. Tijdens de eerste iteratie wordt meer tijd besteed aan de eisen en de analyse, in latere iteraties verschuift dat naar implementeren en testen. Dit is goed te zien in afbeelding 3.2. RUP valt onder de agile methoden.



Afbeelding 3.2: RUP diagram

Meer uitleg over bovenstaande strategieën en methoden die daar gebruik van maken staan in Larman (2004: 18-40).

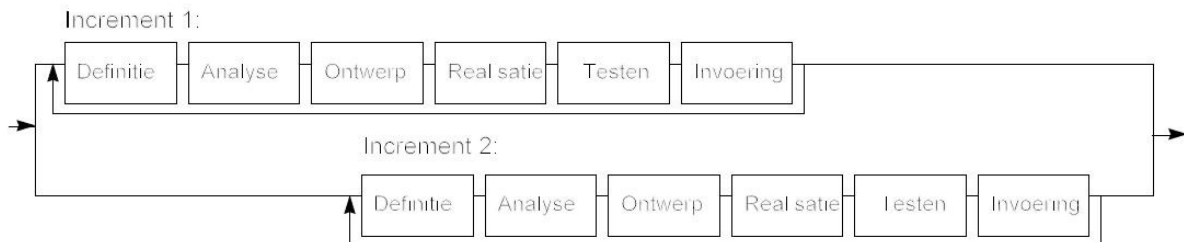
3.1.3 Incrementele strategie

Als laatste de incrementele ontwikkelstrategie. Bij deze strategie worden verschillende incrementen doorlopen, zoals te zien is in afbeelding 3.3. Binnen een increment worden alle ontwikkelfasen doorlopen. Na elk increment is er een resultaat, dat kan bijvoorbeeld een verslag, een prototype of een versie van het product zijn. Er is dus een duidelijk verschil tussen de iteratieve strategie en de incrementele strategie. Bij iteratief worden de fasen verbeterd, terwijl aan heel het project gewerkt wordt, bij incrementeel worden telkens gedeelten van het project gemaakt, waarbij de evaluatie van het vorige deel meegenomen wordt om het product te verbeteren. Er kan dus gezegd worden dat bij iteratief één increment doorlopen wordt, waarbij fasen tijdens dat increment verbeterd kunnen worden. Een combinatie van deze strategieën is mogelijk door elk increment iteratief te doorlopen.

Incrementeel ontwikkelen wordt veel gebruikt bij grote projecten waar de systeemeisen onzeker zijn. Door verschillende incrementen is er al tijdens het project resultaat waardoor de opdrachtgever een gedeelte van het eindresultaat ziet. Als er een verkeerde interpretatie is geweest van de eisen komt dat vaak bij een prototype aan het licht.

Nadeel is dat het ontwikkelproces lastig te doorgronden is en dat de doelstellingen tijdens het project onbedoeld kunnen verschuiven.

Onder deze strategie vallen de agile methoden. Deze methoden zijn heel 'behendig' en 'lenig', omdat er veel verandering binnen een project mogelijk is. Goede voorbeelden van een agile methoden zijn Scrum en RUP.



Afbeelding 3.3: incrementele ontwikkelstrategie

3.1.4 Ongeschikte strategieën

Zoals eerder genoemd is dit een klein project waarbij de systeemeisen niet vastliggen en de ontwikkelaar niet ervaren is. Een lineaire ontwikkelstrategie is niet gepast, omdat de eisen nog niet vastliggen. De eisen moeten nog duidelijk worden en na een resultaat kunnen de eisen veranderen. Dat is niet mogelijk bij de lineaire strategie.

Een iteratieve strategie is ook niet goed toepasbaar op dit project. De reden is grotendeels dezelfde als bij de lineaire strategie. Bij die twee strategieën wordt het eindresultaat in een keer opgeleverd, waardoor de eisen moeilijk kunnen veranderen aan de hand van een resultaat.

3.1.5 Geschikte strategie

De incrementele ontwikkelstrategie sluit wel goed aan bij dit project. Tijdens het project wordt een resultaat opgeleverd. Aan de hand van het resultaat kan de opdrachtgever de eisen bijstellen of verduidelijken. Dat is van groot belang, omdat de systeemeisen onzeker zijn en de ontwikkelaar is onervaren met dit soort projecten.

Bij incrementeel ontwikkelen is het mogelijk om componenten uit eerdere incrementen te verbeteren, dat zorgt voor hogere kwaliteit van het product. Ook is met verschillende incrementen de haalbaarheid van het project eerder in te schatten. Na een increment wordt geëvalueerd of de doelstellingen nog haalbaar zijn.

Daarnaast wordt tijdens het project een gedeelte van het product opgeleverd, dus als het project veel vertraging oploopt is er nog wel een bruikbaar deel van het product.

Aantal incrementen

Tijdens het project worden twee incrementen doorlopen. Bij het kiezen voor een incrementele ontwikkelstrategie is minder dan twee incrementen al uitgesloten, omdat één increment doorlopen geen incrementele ontwikkelstrategie is. Meer dan twee incrementen is gezien de duur van het project niet praktisch. Er worden

documenten opgeleverd tijdens het project, per increment wordt het aantal opleveringen van documenten groter. De verwachting is dat met drie of meer incrementen teveel tijd verloren gaat aan het maken en verbeteren van de documenten. Daarnaast worden de documenten beoordeeld door de opdrachtgever en de begeleider. Bij drie of meer incrementen is het mogelijk dat de periode waarin een document beoordeeld wordt langer is dan een increment. Dan is het project toe aan de volgende versie van een document, maar de feedback is nog niet beschikbaar. Bij twee incrementen is dat risico kleiner.

Voor de werkzaamheden van het project is het niet van belang of er twee of drie incrementen doorlopen worden. De opdracht is goed in twee incrementen op te delen, maar dat is ook het geval bij drie incrementen.

Tijdens dit project wordt een voorbeeld genomen aan het incrementeel ontwikkelen. Een verschil is dat de invoerfase van het afstudeerproject klein is, omdat het om een plug-in gaat die in een bestaand systeem komt. De plug-in komt in de sidebar, de invoer daarvan is erg makkelijk, omdat er geen veranderingen optreden in het systeem. De plug-in is een toevoeging op het systeem. In de planning wordt meer duidelijk over hoe de indeling is van de ontwikkelstrategie.

Risico's

Een risico van deze keuze is dat de eisen tijdens het project onbedoeld kunnen verschuiven. Het is mogelijk om eisen tijdens het project aan te passen of toe te voegen. Deze mogelijkheid kan verkeerd uitpakken, waardoor een eis veranderd terwijl dat niet moet.

3.2 Werkzaamheden

Het project bestaat uit de opstart, twee incrementen en een uitloop. De uitloop wordt gebruikt als er vertraging is tijdens het project. De overige fasen worden per stuk beschreven en er wordt verteld wat de werkzaamheden en de belangrijke beslismomenten zijn. Documentatie over het project wordt tijdens alle fasen gemaakt.

3.2.1 Definitiefase

Tijdens de definitiefase van het project wordt het plan van aanpak en de planning gemaakt. Daarnaast worden er cursussen gedaan met betrekking tot het IBM Lotus, Notes en Domino pakket. Ook wordt onderzocht of er plug-ins zijn die vergelijkbaar zijn met de te maken plug-in. Deze fase is niet lang, omdat bij dit project al door de opdrachtgever onderzocht is of er een vergelijkbaar product op de markt is, dat bleek niet zo te zijn. Daarom is het niet nodig om nog uitgebreid te onderzoeken of de plug-in al bestaat.

3.2.2 Increment 1

Het resultaat van increment één is een plug-in in de sidebar van de Lotus Notes software. In de plug-in kan één module geslept worden en daarvan wordt informatie weergegeven.

Analyse

Tijdens de analyse van increment 1 wordt gezocht naar wat de opdrachtgever

precies wil. Daar zijn interviews voor nodig en onderzoek naar bestaande plug-ins met andere functies. Aan de hand van bestaande plug-ins kan bepaald worden wat gangbaar is voor een plug-in in IBM Lotus Notes en Domino. Aan het einde van de analysefase wordt een programma van eisen opgeleverd.

Ontwerp

Daarna begint de ontwerpfase. Tijdens deze fase wordt een systeem ontworpen. Dit wordt gedaan met de modelleertaal UML. Aan het einde van deze fase wordt het functioneel en een technisch ontwerp opgeleverd. Na het ontwerp wordt begonnen met de realisatie.

Realisatie

Tijdens de realisatie wordt gewerkt in een demonstratieopstelling van CSP. Als de realisatie af is wordt er getest.

Testen

Voordat getest wordt moet eerst een testplan gemaakt worden. De uitkomsten worden gedocumenteerd en gebruikt in increment twee. De oplevering van increment één is een prototype van de plug-in.

3.2.3 Increment 2

Het resultaat van increment twee is een plug-in in de sidebar van de Lotus Notes software. In de plug-in kunnen alle modules van CSP gesleept worden en de gebruiker kan de plug-in zo configureren dat de gewenste informatie weergegeven wordt.

De werkzaamheden van increment twee liggen minder vast dan die van het eerste increment. Dit komt, omdat de resultaten van increment één invloed hebben op increment twee.

Analyse

De analysefase bestaat uit een evaluatie met de opdrachtgever over het resultaat van increment één. Daaruit blijkt of de eisen of de doelstellingen aangepast worden.

Ontwerp

Na de analyse wordt het ontwerp aangepast aan de hand van de bevindingen tijdens de evaluatie en het testen. Het kan zijn dat de opdrachtgever tevreden is met het bestaande model, dan wordt het model verbeterd aan de hand van de uitkomsten van de testen. Ook wordt het ontwerp uitgebreid aan de hand van de eisen die voor increment twee staan.

Realisatie

De realisatiefase bestaat uit het toepassen van de veranderingen en toevoegingen die gedaan zijn in het model. Hierbij wordt verder gegaan met het prototype.

Testen

Als laatste is er een testfase. Hiervoor wordt een testplan geschreven, en er worden testen uitgevoerd. De uitkomsten van de testen worden gedocumenteerd. Na increment twee wordt de eerste versie van de plug-in opgeleverd, tevens het eindresultaat.

3.3 Schema's en tekentechnieken

Alle op te leveren schema's en tekeningen moeten voldoen aan de UML specificaties. UML is een modelleertaal waarmee systemen op een object

georiënteerde manier kunnen worden geanalyseerd en ontworpen. UML is een veelgebruikte taal met een duidelijke standaard. Om deze reden is UML gekozen als standaard modelleertaal voor dit project. UML biedt ook schema's die makkelijk zijn uit te leggen aan klanten. Met behulp van de UML diagrammen is het voor de klant makkelijk om te zien welke kant het project op gaat.

3.4 Planning

In de planning zijn, zoals eerder aangegeven, twee incrementen opgenomen. De globale planning van de afstudeerperiode staat in tabel 3.1.

| | | | | |
|------|---------|-------------|-------------|----------|
| Week | 1 | 2 t/m 9 | 10 t/m 15 | 16 en 17 |
| | Opstart | Increment 1 | Increment 2 | Uitloop |

Tabel 3.1: globale planning

Naast de globale planning is er een gedetailleerdere planning, met milestones per week. Deze staat in tabel 3.2. In de planning vallen de volgende niet genoemde documentaties onder het afstudeerdossier: gedocumenteerde programmacode, documentatie over het implementeren van de plug-in en documentatie over de invoer van de plug-in.

| Week project (weeknummer) | fase | Milestone |
|------------------------------|-------------------------|--------------------------------------|
| 1 (6) | Opstart / Definitie | Plan van Aanpak, Planning |
| 2 (7) | Definitie / Analyse | |
| (8) | Afstudeerder is afwezig | |
| 3 (9) | Analyse | Programma van eisen |
| 4 (10) | Ontwerp | |
| 5 (11) | Ontwerp | Functioneel ontwerp |
| 6 (12) | Realisatie | Voortgangsverslag |
| 7 (13) | Realisatie | Oplevering prototype 1 |
| 8 (14) | Testen | |
| 9 (15) | Analyse | Aangepast programma van eisen |
| 10 (16) | Ontwerp | |
| 11 (17) | Ontwerp / Realisatie | Aangepast functioneel ontwerp |
| 12 (18) | Realisatie | |
| 13 (19) | Realisatie | Oplevering eerste versie van plug-in |
| 14 (20) | Testen / Invoer | Testrapport |
| 15 (21) | uitloop | Afstudeerdossier |
| 16 (22) | uitloop | |

Tabel 3.2: planning op weekbasis

In bijlage 2 staat de uitgebreide planning. Naast de planning die in dit plan van aanpak is opgenomen komt er voor elk increment één apart plan. Dat is nodig, omdat de planning die nu gemaakt is voor increment twee niet zeker is. Ook wordt dan duidelijk wat er na increment één en na increment twee opgeleverd wordt.

4 Projectinrichting

In dit hoofdstuk staat hoe de informatievoorziening is en hoe de faciliteiten zijn. Daarna wordt verteld hoe de kwaliteitsborging is.

4.1 Informatievoorziening

De communicatie met de opdrachtgever is niet complex. De opdrachtgever en opdrachtnemer werken in hetzelfde pand, dus als er informatie nodig is kan dat altijd gevraagd worden, mondeling, via chat of via mail.

Wekelijks is er een evaluatie met de bedrijfsmentor en de opdrachtgever over de voortgang van het project.

De student houdt zelf zijn gewerkte uren bij. Deze kunnen door de opdrachtgever en door de Haagse Hogeschool worden opgevraagd.

4.2 Faciliteiten

Het bedrijf AedifiComm heeft een werkplek voor de afstudeerder. Daarbij hoort een P.C. met toegang tot het netwerk van het bedrijf en internet, een telefoon en een bureau. Verder staat IBM Lotus Notes en Domino op de P.C.. Om te ontwikkelen en te testen is een testomgeving gecreëerd.

4.3 Kwaliteit

De kwaliteit van het eindproduct wordt op verschillende manieren gewaarborgd. Allereerst is er een programma van eisen, deze eisen beschrijven waar het product aan moet voldoen en helpen zo de kwaliteit van het product controleerbaar te maken. Als de controle wordt geëvalueerd en de bevindingen worden toegepast verhoogt dat de kwaliteit.

Daarnaast wordt de ontwikkeling gedaan in twee incrementen volgens een incrementele ontwikkelstrategie. Daardoor kan na elke fase het resultaat geëvalueerd worden of de opleveringen kwaliteit hebben. De opdrachtgever is een expert op het gebied van CSP, dus een evaluatie van de opdrachtgever is betrouwbaar. Ook wordt gewerkt met een ontwikkelstrategie die zichzelf vaak bewezen heeft. Dit is geen garantie voor kwaliteit, maar is wel een hulpmiddel daartoe.

Verder is niet alleen de opdrachtgever expert, maar ook de medewerkers van AedifiComm hebben ervaring met CSP, deze mensen worden regelmatig gevraagd om hun mening over de ontwikkeling van de plug-in.

Als laatste wordt na de realisatie getest. Hiervoor wordt een testplan gemaakt en de uitkomsten van de testen in increment één worden gebruikt om in increment twee het product te verbeteren.

5 Kosten en baten

Tijdens een afstudeerperiode wordt er een opdracht gemaakt. De opdracht moet natuurlijk zorgen voor baten, maar brengt ook kosten met zich mee. In dit hoofdstuk staat beschreven wat de kosten en de baten zijn.

5.1 Kosten

De kosten voor het project zijn:

- Vergoeding voor de student
- Een werkplek
- Tijd van medewerkers van AedifiComm

De student neemt tijdens het afstuderen een werkplek in bij het bedrijf. Daardoor wordt gebruik gemaakt van energie, opslag, printpapier en inkt. Daarnaast wordt tijd gevraagd van medewerkers van het bedrijf, als er uitleg nodig is over CSP of de ontwikkeling van de plug-in.

5.2 Baten

De baten van het project zijn:

- Meer kennis over sidebar plug-ins in Lotus Notes
- Ontwikkeling van CSP
- Een potentiële nieuwe werknemer

De manier waarop de plug-in ontwikkeld wordt maakt gebruik van de nieuwste functies van IBM Lotus Notes en Domino. Deze functies worden besproken en uitgelegd in het eindverslag, daardoor krijgt AedifiComm meer kennis. Na een goed verlopen afstudeerperiode is het niet ongebruikelijk dat de afstudeerder blijft werken bij het afstudeerbedrijf, daarom een potentiële nieuwe werknemer bij de baten.

6 Risicofactoren

In dit hoofdstuk staan de risico's van het project beschreven. Er wordt begonnen met de meest relevante risico's daarna worden telkens minder relevante risico's besproken.

6.1 Afwezigheid afstudeerder

Het kan zijn dat de opdrachtnemer afwezig is door ziekte of andere omstandigheden, waardoor het project vertraging oploopt. De uitloop binnen de planning zorgt ervoor dat die vertraging opgevangen wordt. Dat neemt niet weg dat door langdurige afwezigheid de deadline van het project niet gehaald wordt door de afstudeerder. In dat geval kan AedifiComm voor vervanging zorgen, zodat het project vervolgd wordt.

6.2 Miscommunicatie

6.2.1 School

Het kan zijn dat er miscommunicatie is met de begeleiders vanuit school. Op school zijn meerdere mensen op de hoogte van de opdracht. Daarnaast kan via mail en desnoods telefonisch contact gezocht worden. Dit risico is niet heel groot, ervaring van afgestudeerde studenten leert dat afstudeerders makkelijk contact krijgen met begeleiders vanuit school.

6.2.2 Bedrijf

Miscommunicatie met de opdrachtgever is ook een risico. Bij AedifiComm zijn meerdere mensen op de hoogte van de opdracht waardoor niet alleen de opdrachtgever advies kan geven, maar ook andere medewerkers. Verder wordt er een prototype gemaakt, als na het prototype blijkt dat de opdrachtgever en de opdrachtnemer het resultaat anders interpreterden is er nog een tweede increment om dat recht te zetten.

6.3 Verkeerde ontwikkelstrategie

In hoofdstuk 3 wordt al een risico van de gekozen ontwikkelstrategie genoemd, namelijk dat eisen onbedoeld kunnen verschuiven. Dit wordt tegen gegaan door niet zomaar een eis te veranderen, maar het doel goed voor ogen te houden. Dit risico is niet groot, omdat er maar twee incrementen zijn en de opdrachtgever is ervaren met dit soort projecten.

6.4 Beperking IBM platform

Er is een risico dat wat de opdrachtgever voor ogen heeft niet mogelijk is op het te gebruiken platform van IBM. In dat geval wordt, binnen de mogelijkheden, naar een vergelijkbare oplossing gezocht die wel mogelijk is.

Bronnen

1. Boeken

Larman, C. Applying UML and patterns, and introduction to Object-Oriented analysis and design and iterative development. 3e druk, Upper Saddle River, 2004.

Noordtzijs, E., Plan van Aanpak, een handreiking voor het schrijven van het plan van aanpak, 2004

2. Websites

Roel Grit, http://www.grit-projectmanagement.wolters.nl/7003_projectmanagement/_assets/7003d04.pdf (09 februari 2010).

Bijlagen

1. Begrippenlijst

| Begrip | Uitleg |
|----------------|---|
| CSP | Softwarepakket van AedifiComm dat is gebaseerd op ICM Lotus Notes en Domino |
| Ontwikkelfasen | Definitie, analyse, ontwerpen, realisatie, testen en invoering |

2. Planning

Deze bijlage wordt digitaal gestuurd, met dit document.

4 Incrementplan increment 1

Increment 1

Generieke sidebar plug-in
voor IBM Lotus Notes & Domino

AedifiComm

| | |
|---------------------|--------------------------------|
| Opdrachtgever: | F.J.C. Spruit |
| Bedrijfsmentor: | N.G. Scheppink |
| Begeleider: | D.R. Stikkolorum |
| Expert: | A.G.P. Pronk |
| Afstudeerder: | Chris den Arend |
| Email afstudeerder: | Chris.den.Arend@aedificomm.com |
| Plaats: | Barendrecht |
| Datum: | 08-03-2010 |
| Versie: | 1.0 |



Inhoudsopgave

| | |
|--|----|
| Inleiding..... | 3 |
| 1 Resultaat..... | 4 |
| 1.1 Prototype..... | 4 |
| 1.2 Documenten..... | 4 |
| 2 Eisen van prototype..... | 5 |
| 3 Planning..... | 6 |
| 4 Werkzaamheden | 8 |
| Bijlagen..... | 9 |
| 1. Onderzoek vergelijkbare plug-ins..... | 9 |
| Bestaande plug-ins | 9 |
| Conclusie..... | 10 |

Inleiding

In dit document wordt besproken wat de planning is voor increment één van het project: ontwikkelen van een configureerbare en generieke plugin voor de lotus notes sidebar.

Er wordt ook verteld welke werkzaamheden worden uitgevoerd in increment één. Dit document biedt uitbreidingen op de gedeelten over increment één in het plan van aanpak en het programma van eisen.

Het document wordt aan het einde van de analysefase opgeleverd. Daar is voor gekozen, omdat de eisen dan bekend zijn, waardoor de werkzaamheden gedetailleerder in de planning van het increment gezet kunnen worden.

Het doel van dit verslag is uiteenzetten van de werkzaamheden in increment één. Daarbij hoort een planning en afbakening van de werkzaamheden. Als eerste wordt het beoogde resultaat van het increment behaldeld. Daarna de eisen, de planning en de werkzaamheden.

1 Resultaat

In dit hoofdstuk wordt besproken wat de geplande resultaten zijn van dit increment. Eerst wordt het prototype besproken en vervolgens de documenten die worden opgeleverd.

1.1 Prototype

Increment één levert onder andere een prototype op. Dat prototype wordt in increment twee verbeterd en afgemaakt. De eisen waar het prototype na increment één aan moet voldoen staan in hoofdstuk drie van dit document. De volledige lijst van eisen staat in het programma van eisen.

Het resultaat van increment één is een plug-in voor de sidebar van Lotus Notes. Deze plug-in is snel te installeren en te verwijderen. Als de plug-in geïnstalleerd is kan er een module geselecteerd worden waarvan de plug-in informatie laat zien. Belangrijk is dat de plug-in na dit increment nog niet generiek is. Alleen de module "persoonlijke activiteiten" functioneert met de plug-in. Het uiterlijk van de plug-in sluit aan bij Lotus Notes en CSP en de plug-in draait op de platformen die in de eisen staan.

1.2 Documenten

Naast het prototype worden in het eerste increment ook documenten opgeleverd. Het eerste document is het plan van aanpak. Dit document is nodig om de opdracht goed in beeld te brengen. Ook staat in het plan van aanpak hoe het project aangepakt wordt. Opdrachtgever, opdrachtnemer en in dit geval ook de begeleider van de afstudeerder hebben dan een goed beeld van de opdracht en de aanpak daarvan. Als bijlage van het plan van aanpak wordt een gedetailleerde planning opgeleverd.

Naast het plan van aanpak wordt er een programma van eisen gemaakt. In het programma van eisen worden ook de eisen die bekend zijn voor iteratie twee al meegenomen. Daar is voor gekozen, omdat anders moeilijk te overzien is hoe groot het project is. Ook kunnen de doelen van de incrementen niet bepaald worden als niet alle eisen bekend zijn. Het document over de eisen gaat dieper dan het plan van aanpak in op de eisen van de plug-in. In de analysefase van increment twee wordt het programma van eisen aangepast en uitgebreid met behulp van het gemaakt prototype.

Tijdens de ontwerpfase wordt een ontwerp gemaakt. Het ontwerp beschrijft de benodigde software om aan de eisen van increment één te voldoen. In de modelleerfase van increment twee wordt het ontwerp uitgebreid.

AedifiComm heeft weinig kennis over het ontwikkelen van een plug-in voor Lotus Notes, daarom is het van belang dat tijdens het implementeren goed gedocumenteerd wordt hoe alles geïmplementeerd wordt. Dat zorgt ervoor dat vergaarde kennis niet verloren gaat.

Aan het einde van increment één wordt de plug-in getest. De uitkomsten van de testen komen in een testrapport, deze bevindingen worden in increment twee gebruikt om de plug-in te verbeteren.

Tijdens het afstuderen wordt ook een tussenverslag gemaakt. Dit tussenverslag wordt tijdens iteratie een opgeleverd.

2 Eisen van prototype

In dit hoofdstuk staan de eisen die van toepassing zijn op increment één. De eisen komen uit het programma van eisen, maar wel gefilterd op increment één.

| Eisen | | Prio | Tijd | Nr. | |
|------------------|---------------------------------------|---|------|-----|----|
| Functioneel | | | | | |
| | Informatie | | | | |
| | | De gebruiker kan de module "persoonlijke activiteiten" selecteren. | M | I1 | 1 |
| | | De gebruiker kan informatie inzien over de module die geselecteerd is. | M | I1 | 2 |
| | (de)installatie | | | | |
| | | De gebruiker kan de plug-in installeren. | M | I1 | 7 |
| | | De gebruiker kan de plug-in deïnstalleren. | M | I1 | 8 |
| | Vernieuwen | | | | |
| | | De plug-in heeft een timer | M | I1 | 33 |
| | | De gebruiker kan de plug-in vernieuwen. | M | I1 | 11 |
| | Gemak | | | | |
| | De plug-in is in de Nederlandse taal. | M | I1 | 14 | |
| Niet-functioneel | | | | | |
| | Performance | | | | |
| | | De plug-in geeft informatie snel weer. <i>Snel = sneller dan huidige navigatie naar informatie</i> | M | I1 | 15 |
| | | De plug-in is snel geïnstalleerd <i>Snel = korter dan 3 minuten</i> | M | I1 | 16 |
| | | De plug-in is snel gedeïnstalleerd <i>Snel = korter dan 3 minuten</i> | M | I1 | 17 |
| | Uiterlijk | | | | |
| | | De user-interface van de plug-in moet aansluiten bij CSP. | M | I1 | 20 |
| | | De user-interface van de plug-in moet aansluiten bij Lotus Notes. | M | I1 | 21 |
| | | De plug-in past in de sidebar van Lotus Notes. | M | I1 | 22 |
| | Platform | | | | |
| | | De plug-in werkt op het platform IBM Lotus Notes en Domino 8.5. | M | I1 | 23 |
| | | De plug-in werkt op de versie 8 van CSP. | M | I1 | 24 |
| | | De plug-in werkt op windows 2003 en hoger, 32 en 64 bits versie | M | I1 | 25 |
| | Gemak | | | | |
| | | De gebruiker kan de te maken plug-in meerdere keren in de sidebar van Lotus Notes installeren | M | I1 | 28 |

3 Planning

In dit hoofdstuk staan korte werkzaamheden met daarbij de deadline. Het deel van increment één uit de planning die bij het plan van aanpak als bijlage is opgeleverd is in zijn geheel van toepassing op dit increment. Voor increment twee is de planning minder zeker, omdat dat increment later is dan increment één.

| Taak | Deadline |
|--|--------------------|
| Plan van aanpak (PvA) versie 0.1 af | 10 februari |
| Feedback PvA opdrachtgever en bedrijfsmentor | 11 februari |
| Planning versie 0.1 af | 11 februari |
| Feedback planning bedrijfsmentor | 11 februari |
| Plannen interview opdrachtgever | 11 februari |
| PvA versie 0.2 af | 12 februari |
| Onderzoek naar vergelijkbare plug-ins | 12 februari |
| Cursus IBM Lotus Notes en Domino | 12 februari |
| Planning versie 1.0 af | 12 februari |
| Plannen eerste afspraak met begeleider | 17 februari |
| Interview opdrachtgever | 16 februari |
| Feedback PvA begeleider | 16 februari |
| Onderzoek eisen voor plug-ins voor de sidebar | 16 februari |
| PvA versie 1.0 af | 17 februari |
| Programma van Eisen (PvE) versie 0.1 af | 19 februari |
| Feedback Programma van Eisen opdrachtgever | 19 februari |
| PvE versie 0.2 af | 2 maart |
| Feedback Programma van Eisen begeleider | 4 maart |
| PvE versie 1.0 af | 5 maart |
| Afspraak begeleider maken voor voortgangsgesprek | 5 maart |
| Functioneel ontwerp versie 0.1 af | 10 maart |
| Feedback functioneel ontwerp bedrijfsmentor | 12 maart |
| Functioneel ontwerp versie 0.2 af | 15 maart |
| Bedrijfsbezoek begeleider | 15 maart |
| Feedback functioneel ontwerp begeleider | 18 maart |
| Functioneel ontwerp versie 1.0 | 19 maart |
| Samenvoegen en afmaken van voortgangsverslag versie 0.1 | 22 maart |
| Plannen bespreking concept afstudeerdossier | 26 maart |
| Voortgangsgesprek begeleider | 22 maart – 2 april |
| Feedback voortgangsverslag opdrachtgever en bedrijfsmentor | 26 maart |
| Voortgangsverslag versie 0.2 | 26 maart |
| Feedback voortgangsverslag begeleider | 31 maart |
| Implementeren prototype 1 | 2 april |
| Voortgangsverslag versie 1.0 | 2 april |
| Maken van testplan | 6 april |

| | |
|------------------------------------|--------------------|
| Uitvoeren van testplan | 8 april |
| Uitkomsten van testen documenteren | 9 april |
| Afstudeerdossier versie 0.1 | 9 april |
| Bespreken concept afstudeerdossier | 5 april – 16 april |

4 Werkzaamheden

Het laatste hoofdstuk gaat over de werkzaamheden. Hierin wordt per fase beschreven wat de werkzaamheden zijn tijdens increment één.

Definitie

In de definitiefase wordt het plan van aanpak met planning gemaakt en er wordt een onderzoek gedaan naar vergelijkbare bestaande plug-ins. Tijdens dit schrijven is het onderzoek naar vergelijkbare plug-ins afgerond en de documentatie daarvan is te vinden in bijlage 1.

Analyse

Tijdens de analyse van increment 1 wordt gezocht naar wat de opdrachtgever precies wil. Daar zijn interviews voor nodig en onderzoek naar bestaande plug-ins met andere functies. Aan de hand van bestaande plug-ins kan bepaald worden wat gangbaar is voor een plug-in in IBM Lotus Notes en Domino. Aan het einde van de analysefase wordt een programma van eisen opgeleverd.

Ontwerp

Daarna begint de ontwerpfase. Tijdens deze fase wordt een systeem ontworpen. Dit wordt gedaan met de modelleertaal UML. Aan het einde van deze fase wordt het functioneel en een technisch ontwerp opgeleverd. Na het ontwerp wordt begonnen met de realisatie.

Realisatie

Tijdens de realisatie wordt gewerkt in een demonstratieopstelling van CSP. Als de realisatie af is wordt er getest.

Testen

Voordat getest wordt moet eerst een testplan gemaakt worden. De uitkomsten worden gedocumenteerd en gebruikt in increment twee. De oplevering van increment één is een prototype van de plug-in.

Bijlagen

1. Onderzoek vergelijkbare plug-ins

In dit hoofdstuk staan de resultaten van het onderzoek naar vergelijkbare plug-ins. Eerst worden voorbeelden van bestaande plug-ins besproken, daarna is er een conclusie.

Bestaande plug-ins

Bestaande plug-ins voor de sidebar van Lotus Notes:

- **File Navigator**

De File Navigator laat het operating system file system zien. Vanaf de sidebar kunnen files worden geopend, gedelete, gekopiëerd en geplakt. Files en folders kunnen door middel van drag en drop in Notes document worden gezet.

- **Chart Plugin**

Deze plugin geeft laat een grafiek of tabel van een geselecteerde view zien.

- **Document property viewer**

In deze plug-in staan instellingen van het huidige geselecteerde document.

- **Formul8**

Deze plug-in maakt het mogelijk om formules te schrijven en uit te voeren op het geselecteerde document.

- **Secret Agent**

De Secret Agent maakt het mogelijk om snel informatie te verkrijgen van tussenpersonen/vertegenwoordiger.

- **SideLog**

Geeft snel inzicht in logs over de domino server.

- **Recent Contacts**

Deze plug-in geeft recente contacten weer. Er kan aangegeven worden welke contacten belangrijk zijn, die staan bovenaan in de plug-in.

- **Sidebar plug-in voor lotus enterprise integrator**

Deze plug-in maakt het mogelijk om Lotus Enterprise Integrator te monitoren.

- **Lotus Quickr**

Quickr is een applicatie, er is ook een plugin van Quickr voor de sidebar van Lotus Notes. Met drag & drop kunnen files in de sidebar gezet worden, deze kunnen dan gedeeld worden met collega's of verwijderd worden uit de inbox. Ook kan de plug-in een attachment in een library zetten en het attachment vervangen voor een link.

- **Side Note**

Deze plug-in zorgt voor een notepad in de sidebar.

- **TwitNotes**

Met deze plug-in kunnen twitterberichten gelezen worden, de status worden veranderd en er kunnen openbare en persoonlijke berichten verstuurd worden.

- **LinkedIn plugin**

De plug-in van LinkedIn is vergelijkbaar met de plug-in van twitter (TwitNotes), het verschil is natuurlijk dat het bij deze plug-in om LinkedIn gaat.

- **TripIt**

Hier zijn aanbiedingen van TripIt te zien en informatie over reizen die gemaakt kunnen worden.

Conclusie

Het onderzoek heeft veel plug-ins voor de sidebar van Lotus Notes opgeleverd. Er is een verschil te zien tussen plug-ins die samenwerken met modules in IBM Lotus Notes en Domino en plug-ins die samenwerken met andere platforms.

Tussen de plug-in die gemaakt moet worden voor het afstuderen en de onderzochte plug-ins zit een groot verschil. Het verschil is dat de plug-in van de opdracht samen moet werken met alle modules in CSP. De onderzochte modules werken allemaal niet met CSP en met hooguit één module uit Lotus Notes, Lotus Domino of een ander platform.

Bronnen

OpenNTF, www.openntf.org/Internal/home.nsf (10 februari 2010)

Socialtext,

www.socialtext.net/lotus/index.cgi?lotus_notes_sidebar_apps_and_plugins (10 februari 2010)

5 Programma van eisen versie 1.0

Programma van Eisen

Generieke sidebar plug-in

voor IBM Lotus Notes & Domino

AedifiComm

| | |
|---------------------|--------------------------------|
| Opdrachtgever: | F.J.C. Spruit |
| Bedrijfsmentor: | N.G. Scheppink |
| Begeleider: | D.R. Stikkolorum |
| Expert: | A.G.P. Pronk |
| Afstudeerder: | Chris den Arend |
| Email afstudeerder: | Chris.den.Arend@aedificomm.com |
| Plaats: | Barendrecht |
| Datum: | 08-03-2010 |
| Versie: | 1.0 |



Inhoudsopgave

| | |
|---|----|
| Inleiding..... | 3 |
| 1 Use-case-diagrammen..... | 4 |
| 1.1 Globaal use-case-diagram | 4 |
| 1.2 Gedetailleerde use-case-diagrammen..... | 6 |
| 1.2.1 Plug-in aanpassen | 6 |
| 1.2.2 Informatie inzien..... | 10 |
| 1.2.3 Informatie doorzoeken | 12 |
| 1.2.4 Up to date maken | 16 |
| 2 Eisen | 18 |
| 3 Onthouden van instellingen..... | 20 |
| Bijlagen..... | 21 |
| 1. Afkomst eisen | 21 |

Inleiding

Aedificomm heeft het softwarepakket CSP ontwikkeld. CSP bestaat uit diverse modules en het is op dit moment nog niet mogelijk om de meest relevante informatie van die modules te allen tijde zichtbaar te maken voor de eindgebruiker. Elke module bevat informatie en voor elke gebruiker heeft relevante informatie een andere invulling.

In CSP wordt de relevante informatie niet op elk moment weergegeven. Om informatie te vinden moet eerst genavigeerd worden naar de module waar de gezochte informatie in staat. Deze actie kost veel tijd en is niet gebruikersvriendelijk. Daarom wordt er een plug-in gemaakt voor de sidebar van Lotus Notes. Deze plug-in geeft te allen tijde de gewenste informatie weer.

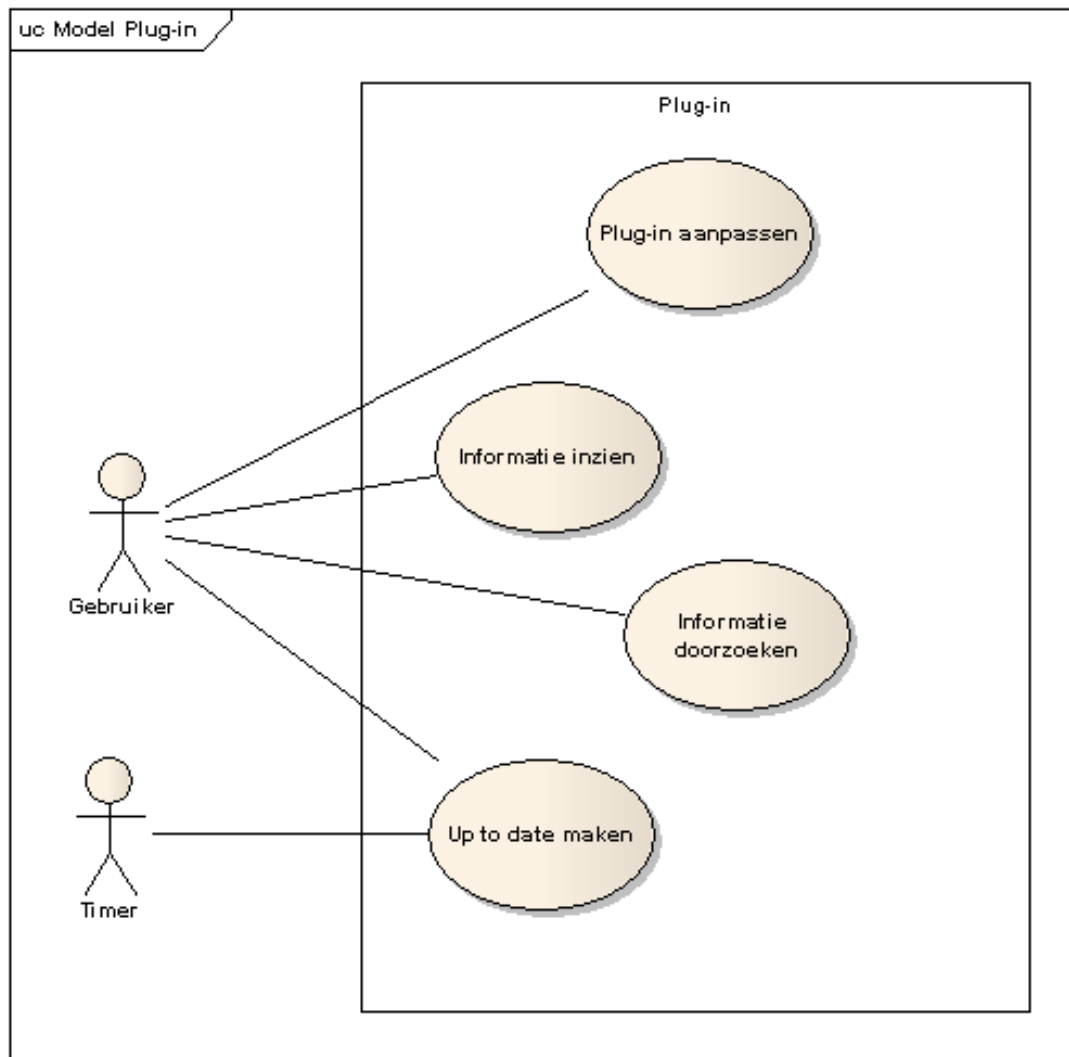
In dit document staat aan welke eisen de te maken plug-in moet voldoen. Als eerste wordt stilgestaan bij de use-case-diagrammen. Er is een globaal diagram en er zijn meerdere specifieke diagrammen. Er is voor gekozen om scenario's te maken, omdat de scenario's het verloop van acties duidelijker maken voor de gebruiker. Na de diagrammen komen de eisen aan bod.

1 Use-case-diagrammen

In dit hoofdstuk staan de use-case-diagrammen. Als eerste wordt een globaal use-case-diagram weergegeven waar in één oogopslag is te zien wat de gebruiker met het systeem kan. Daarna is er per use-case een gedetailleerd model gemaakt. De use-case-diagrammen zijn bedoeld om samen met de gebruikers en opdrachtgever vast te leggen hoe de gebruiker met de plug-in om moet gaan. Bij elk diagram staan ook beschrijvingen per use-case en scenario's. Bij de scenario's zijn de acties van de gebruiker niet in chronologische volgorde. Daar is voor gekozen, omdat er geen eenduidige volgorde bestaat van de mogelijke acties. Er wordt in dit hoofdstuk niet stilgestaan bij de implementatie van de plug-in.

1.1 Globaal use-case-diagram

In dit use-case-diagram, zie afbeelding 1.1, staan de globale use-cases die betrekking hebben op de gehele plug-in.



Afbeelding 1.1: globaal use-case-diagram van de plug-in

| | |
|---------------------|--|
| Naam | Plug-in aanpassen |
| Samenvatting | De instellingen van de plug-in worden aangepast |
| Aanname | De plug-in is beschikbaar |
| Preconditie | Geen |
| Postconditie | De plug-in is aangepast |
| Omschrijving | De gebruiker gebruikt een functie van de plug-in of de sidebar van Lotus Notes om de plug-in aan te passen |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Informatie inzien |
| Samenvatting | De plug-in laat informatie zien |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes |
| Preconditie | Geen |
| Postconditie | De plug-in laat informatie zien |
| Omschrijving | De gebruiker raadpleegt de plug-in |
| Uitzondering | Geen |

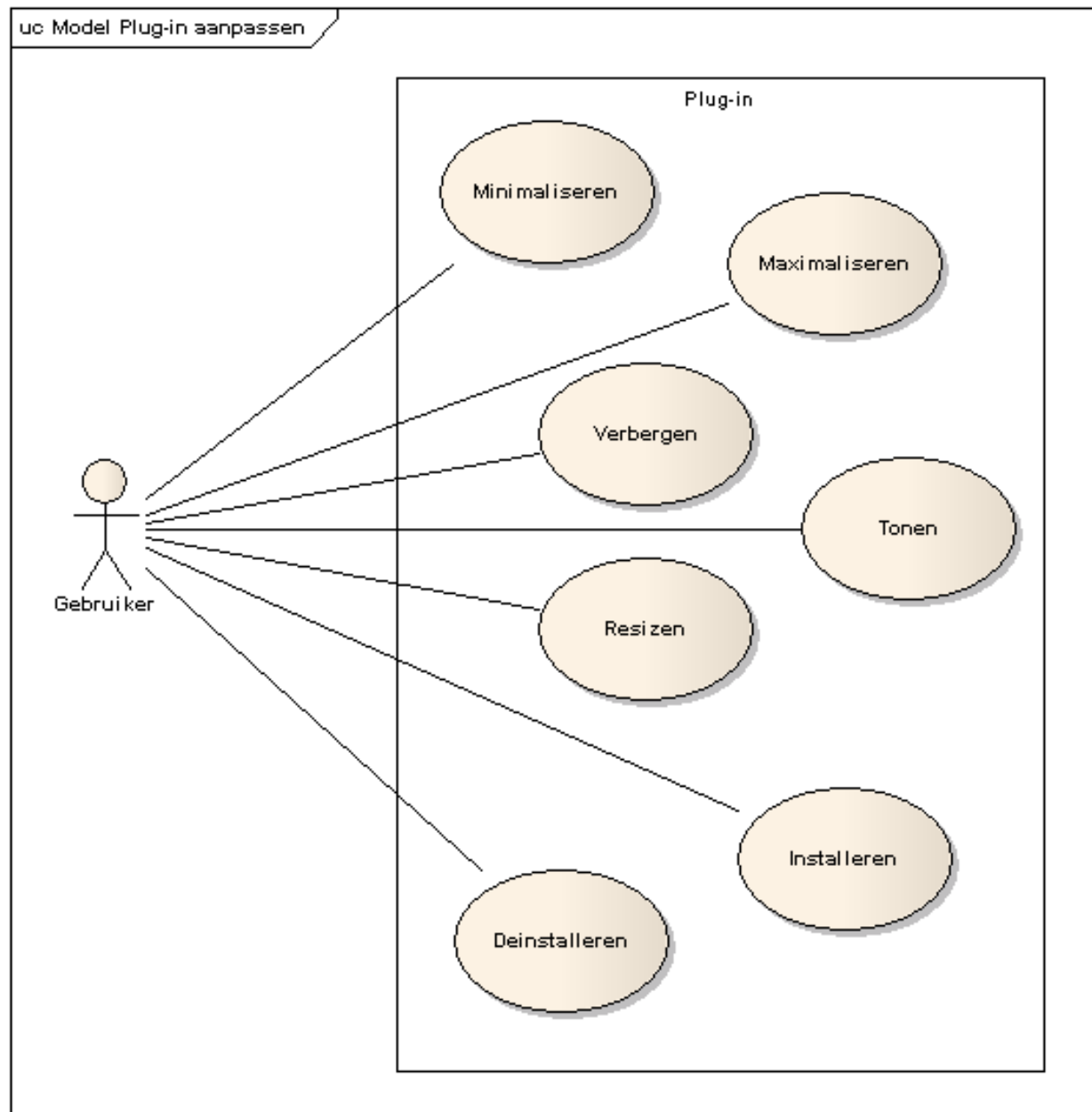
| | |
|---------------------|--|
| Naam | Informatie doorzoeken |
| Samenvatting | In de plug-in wordt naar informatie gezocht |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes |
| Preconditie | Er is een module geselecteerd |
| Postconditie | De plug-in laat gezochte informatie zien |
| Omschrijving | De gebruiker geeft criteria op waaraan de informatie moet voldoen. |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Up to date maken |
| Samenvatting | De plug-in haalt informatie op die op dat moment in de database staat |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes |
| Preconditie | Er is een module geselecteerd |
| Postconditie | De plug-in laat actuele informatie zien |
| Omschrijving | De gebruiker geeft aan dat de actuele informatie opgehaald moet worden, of er loopt een timer af |
| Uitzondering | Geen |

1.2 Gedetailleerde use-case-diagrammen

In het dit gedeelte van het document staat per use-case van het globale diagram een use-case-model, daarnaast zijn ook scenario's gemaakt voor elk model.

1.2.1 Plug-in aanpassen



Afbeelding 1.2: use-case-diagram "plug-in aanpassen"

| | |
|---------------------|---|
| Naam | Minimaliseren |
| Samenvatting | De plug-in wordt geminimaliseerd in de sidebar |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes en er staan naast deze plug-in één of meer andere plug-ins in de sidebar |
| Preconditie | De plug-in is niet geminimaliseerd |
| Postconditie | De plug-in staat geminimaliseerd in de sidebar |
| Omschrijving | De gebruiker klikt met de linkermuisknop op de titel van de plug-in |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Maximaliseren |
| Samenvatting | De plug-in wordt gemaximaliseerd in de sidebar |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes |
| Preconditie | De plug-in is niet gemaximaliseerd |
| Postconditie | De plug-in staat gemaximaliseerd in de sidebar |
| Omschrijving | De gebruiker klikt met de linkermuisknop op de titel van de plug-in |
| Uitzondering | Geen |

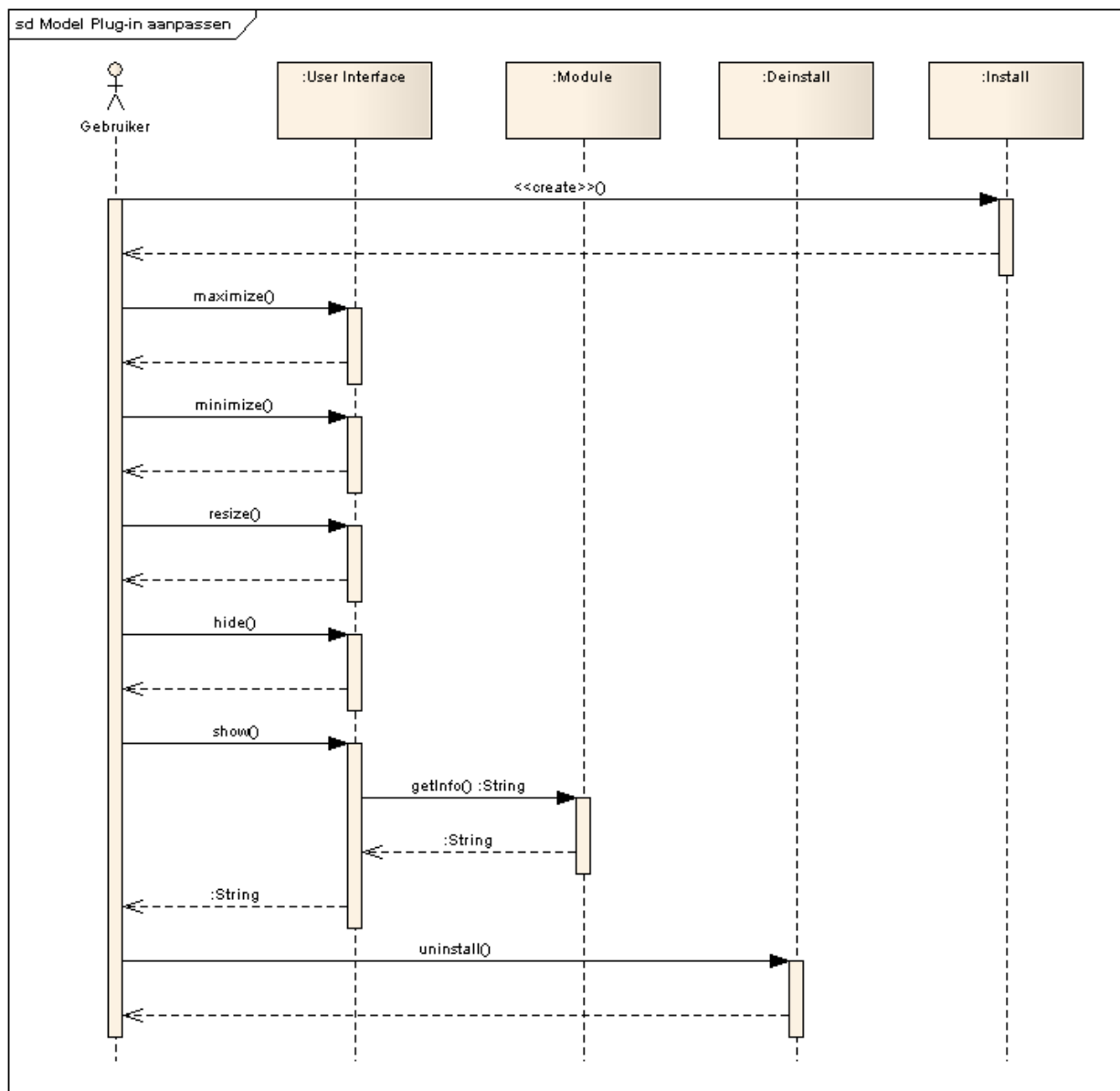
| | |
|---------------------|---|
| Naam | Verbergen |
| Samenvatting | De plug-in verdwijnt uit de sidebar van Lotus Notes, maar is nog wel geïnstalleerd |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes |
| Preconditie | Geen |
| Postconditie | De plug-in is niet zichtbaar in de sidebar van Lotus Notes |
| Omschrijving | De gebruiker klikt met de rechtermuisknop op de titel van de plugin en kiest in het menu dat openklapt: Hide panel: "plug-in naam" Of de gebruiker navigeert in het menu naar view->right sidebar panels en vinkt daar de plug-in uit. |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Tonen |
| Samenvatting | De plug-in wordt zichtbaar in de sidebar van Lotus Notes |
| Aanname | De plug-in is geïnstalleerd in de sidebar van Lotus Notes en is verborgen |
| Preconditie | Geen |
| Postconditie | De plug-in is zichtbaar in de sidebar van Lotus Notes |
| Omschrijving | De gebruiker navigeert in het menu naar view->right sidebar panels en vinkt daar de plug-in aan. |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Resizen |
| Samenvatting | De afmetingen van de plug-in worden aangepast |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes |
| Preconditie | Er staat naast deze plug-in één of meer andere plug-ins in de sidebar |
| Postconditie | De afmetingen van de plug-in zijn aangepast |
| Omschrijving | De gebruiker sleept de onderkant van de plug-in omhoog of omlaag. Of de gebruiker sleept de linker zijkant van de sidebar naar links of naar rechts. |
| Uitzondering | Geen |

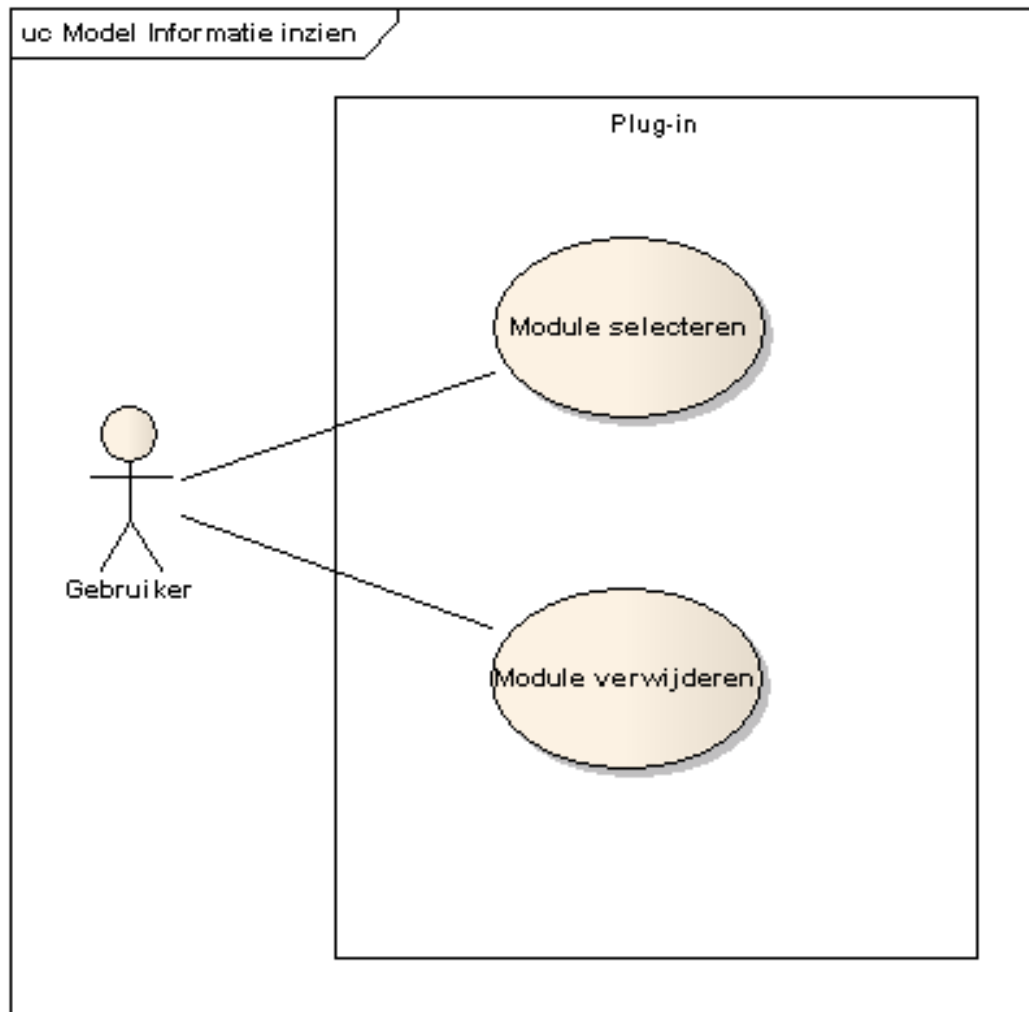
| | |
|---------------------|---|
| Naam | Installeren |
| Samenvatting | De plug-in wordt in de sidebar van Lotus Notes geïnstalleerd |
| Aanname | De plug-in is beschikbaar |
| Preconditie | Geen |
| Postconditie | De plug-in is geïnstalleerd en te zien in de sidebar van Lotus Notes |
| Omschrijving | De gebruiker sleept het icoon van de plug-in in de sidebar van Lotus Notes. Of de gebruiker navigeert in het menu naar de plug-in om deze te installeren. Ter verduidelijking: de gebruiker kan de plug-in meerdere keren installeren in de sidebar van Lotus Notes |
| Uitzondering | Geen |

| | |
|---------------------|---|
| Naam | Deïnstalleren |
| Samenvatting | De plug-in wordt verwijderd uit Lotus Notes |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes |
| Preconditie | Geen |
| Postconditie | De verwijderde plug-in is niet meer aanwezig in Lotus Notes |
| Omschrijving | De gebruiker gaat naar de plug-in "My Widgets", daar doet de gebruiker rechtermuisknop op de te verwijderen plug-in en klikt dan op remove. |
| Uitzondering | Geen |



Afbeelding 1.3: scenario "plug-in aanpassen"

1.2.2 Informatie inzien

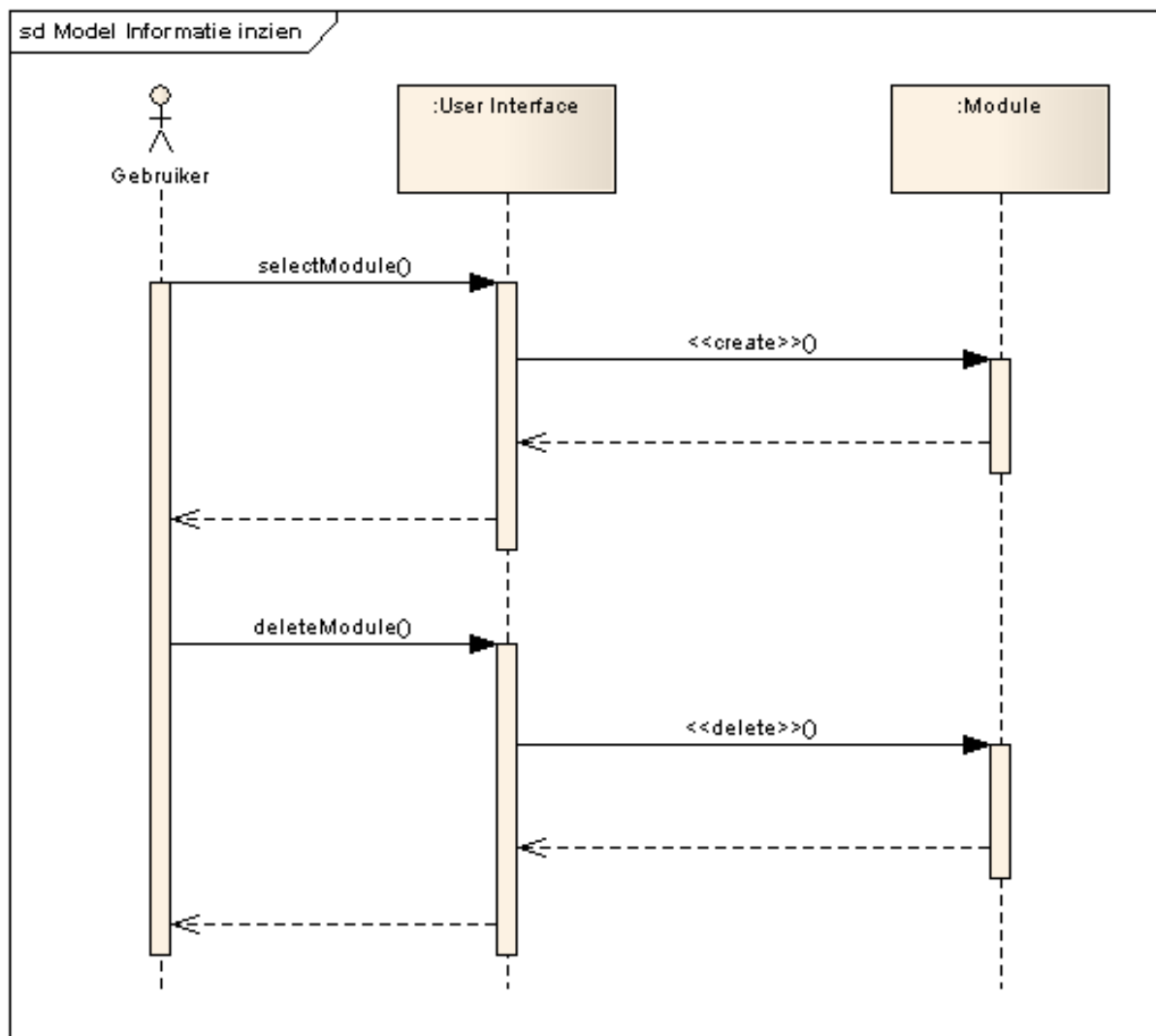


Afbeelding 1.4: use-case-diagram "informatie inzien"

| | |
|---------------------|--|
| Naam | Module selecteren |
| Samenvatting | Er wordt een module voor de plug-in geselecteerd |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes |
| Preconditie | Geen |
| Postconditie | Informatie over de geselecteerde module wordt weergegeven in de plug-in |
| Omschrijving | <p>De gebruiker kan een module openen, deze kan dan in de plug-in gesleept worden.</p> <p>De plug-in geeft dan de informatie weer van de module die in de plug-in gesleept is. De informatie is de informatie die in de laatst geselecteerde view staat. Als er geen view geselecteerd is wordt de default view weergegeven.</p> |

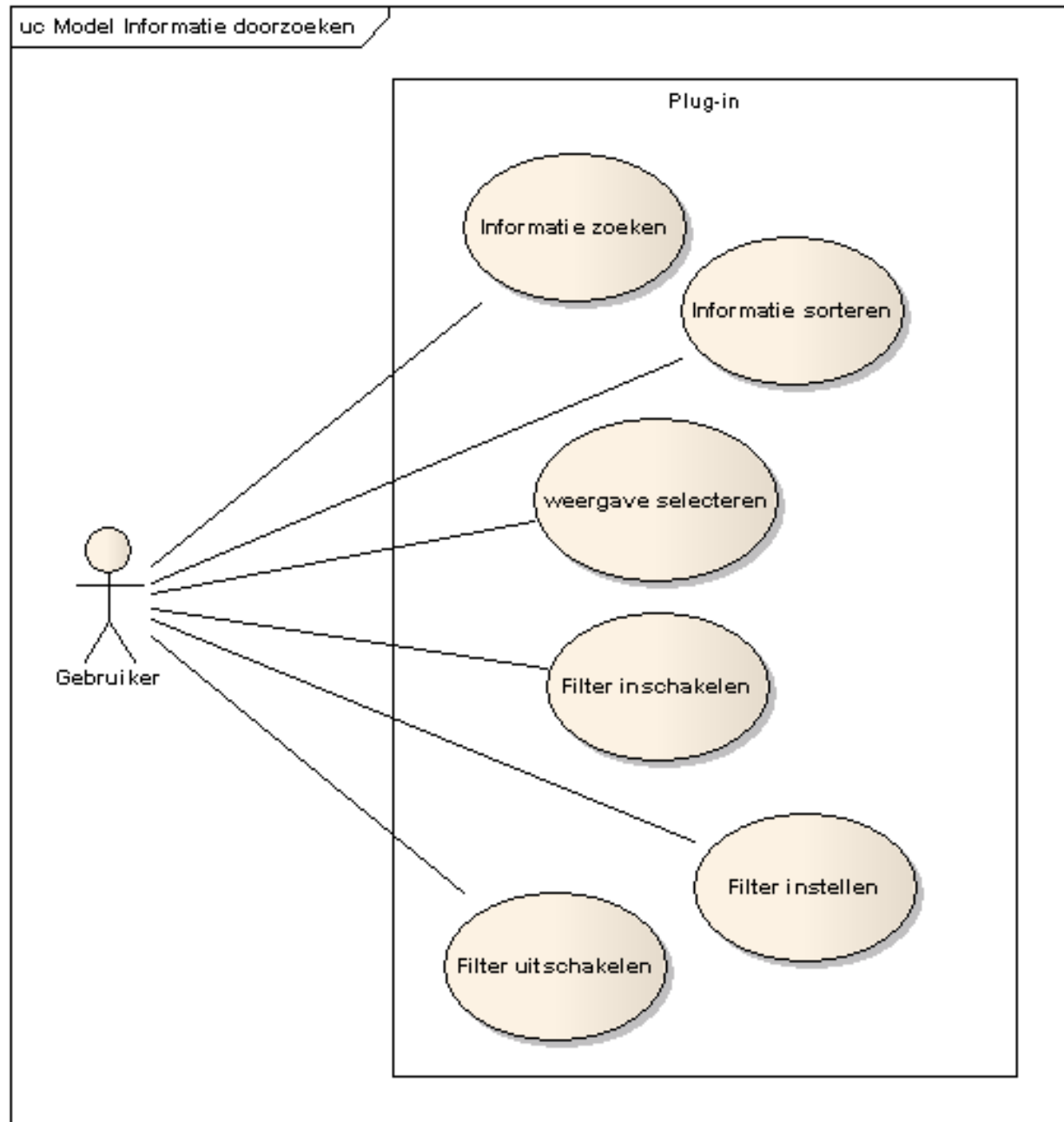
| | |
|---------------------|---|
| | Als er al een module geselecteerd was wordt deze overschreven |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Module verwijderen |
| Samenvatting | Een module wordt uit de plug-in verwijderd |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes |
| Preconditie | De plug-in bevat een module |
| Postconditie | De module is niet weergegeven in de plug-in |
| Omschrijving | De gebruiker kan met de rechtermuisknop op een module in de plug-in klikken. Daar staat dan een optie om de module te verwijderen. |
| Uitzondering | Geen |



Afbeelding 1.5: scenario "informatie inzien"

1.2.3 Informatie doorzoeken



Afbeelding 1.6: use-case-diagram "informatie doorzoeken"

| | |
|---------------------|--|
| Naam | Informatie zoeken |
| Samenvatting | Er wordt gezocht in de plug-in |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes |
| Preconditie | De plug-in bevat een module |

| | |
|---------------------|--|
| Postconditie | De resultaten van de zoekactie zijn zichtbaar in de plug-in |
| Omschrijving | Onder de titel van de plug-in staat een zoekbalk. Hierin kan de gebruiker een zoekwoord typen en op de zoekknop klikken. |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Weergave selecteren |
| Samenvatting | De weergave van de module wordt geselecteerd |
| Aanname | De plug-in is geïnstalleerd, zichtbaar in de sidebar van Lotus Notes en de plug-in bevat een module die meerdere views ondersteund |
| Preconditie | Geen |
| Postconditie | De plug-in heeft een door de gebruiker geselecteerde weergave |
| Omschrijving | De gebruiker klikt met de rechtermuisknop op de titel van de plug-in. Daar kan geklikt worden op weergave. Dan komt er een nieuw scherm. Daar kan de gebruiker een weergave selecteren, zodat de plug-in op die manier weergegeven wordt. |
| Uitzondering | Geen |

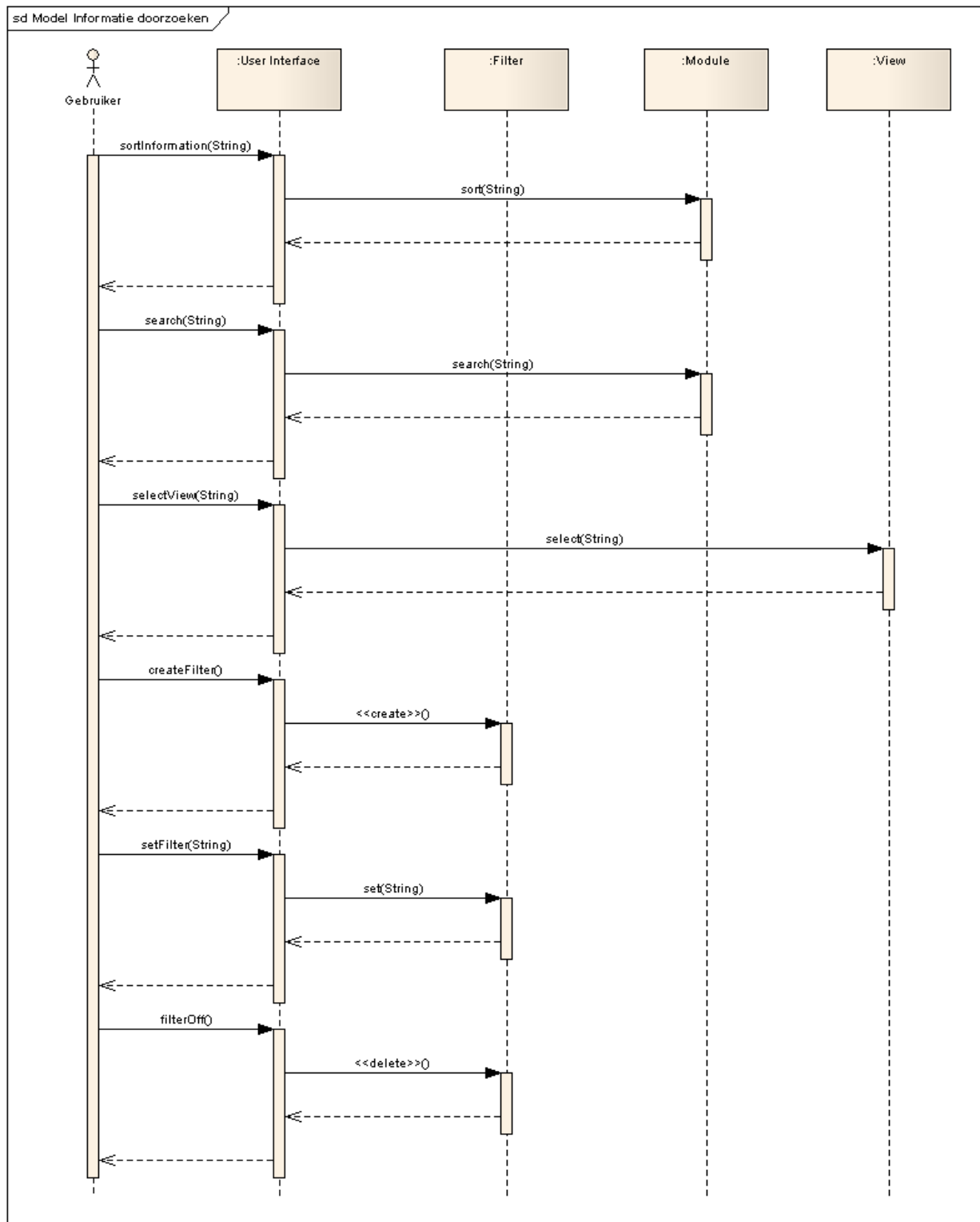
| | |
|---------------------|---|
| Naam | Informatie sorteren |
| Samenvatting | De informatie in de plug-in is gesorteerd |
| Aanname | De plug-in is geïnstalleerd, zichtbaar in de sidebar van Lotus Notes en de plug-in bevat een module |
| Preconditie | Geen |
| Postconditie | De resultaten van de sorteeractie zijn zichtbaar in de plug-in |
| Omschrijving | De gebruiker kan op de titel van een kolom klikken, dan wordt de informatie gesorteerd. |
| Uitzondering | Geen |

| | |
|---------------------|---|
| Naam | Filter inschakelen |
| Samenvatting | De filter van de plug-in wordt ingeschakeld |
| Aanname | De plug-in is geïnstalleerd, zichtbaar in de sidebar van Lotus Notes en de plug-in bevat een module |
| Preconditie | Geen |
| Postconditie | De plug-in laat alle informatie zien die door de filter gelaten wordt |
| Omschrijving | De gebruiker klikt met de rechtermuisknop op de titel van de plug-in. Daar is dan een optie om de filter in te schakelen. |
| Uitzondering | Geen |

| | |
|---------------------|---|
| Naam | Filter instellen |
| Samenvatting | De filter van de plug-in wordt ingesteld |
| Aanname | De plug-in is geïnstalleerd, zichtbaar in de sidebar van Lotus Notes en de plug-in bevat een module |
| Preconditie | Geen |
| Postconditie | De filter van de plug-in is ingesteld met voorwaarden die de gebruiker heeft ingevoerd |

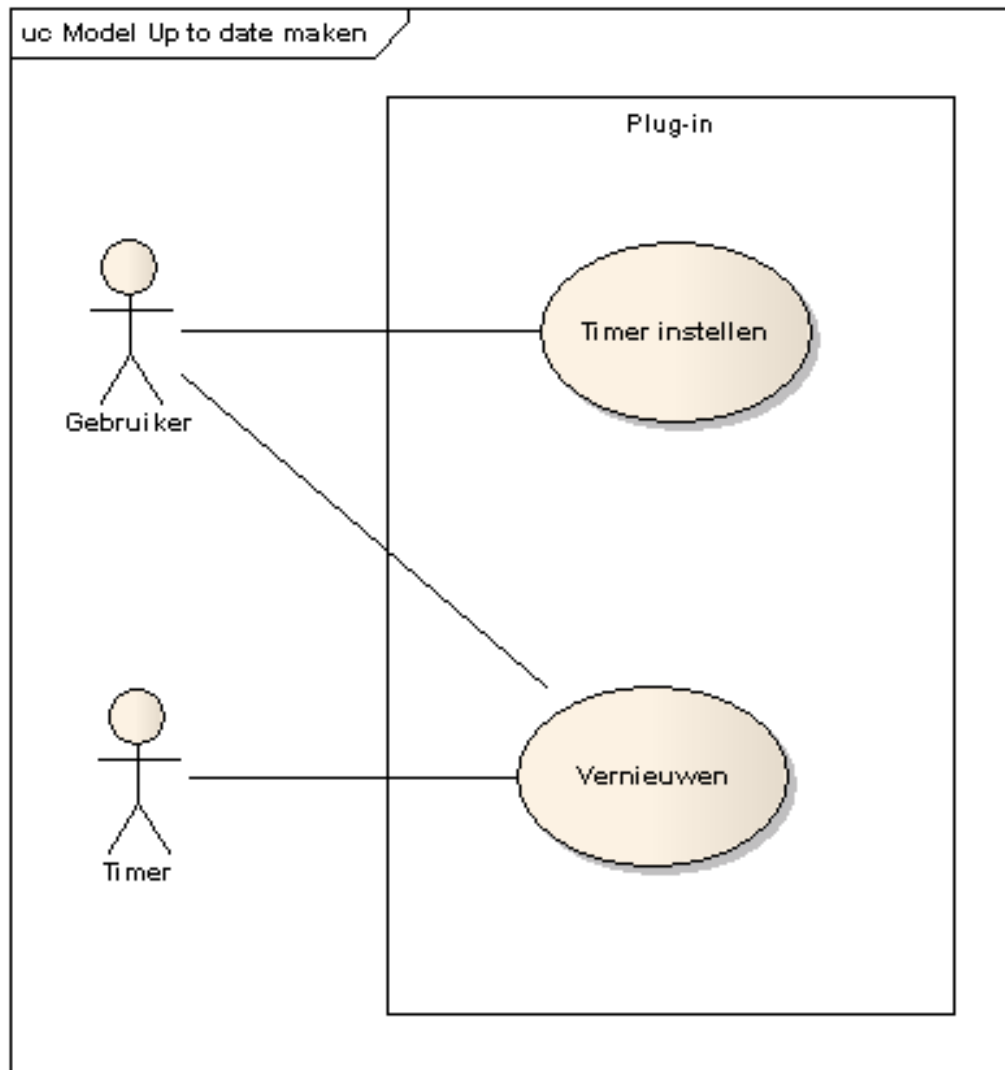
| | |
|---------------------|--|
| Omschrijving | De gebruiker klikt met de rechtermuisknop op de titel van de plug-in. Daar kan geklikt worden op filter instellen. Dan komt er een nieuw scherm. Daar kan de gebruiker voorwaarden stellen aan de informatie die weergegeven moet worden. De ingestelde waarden blijven bestaan, tot de gebruiker de instellingen wijzigt. |
| Uitzondering | Geen |

| | |
|---------------------|---|
| Naam | Filter uitschakelen |
| Samenvatting | De filter van de plug-in wordt uitgeschakeld |
| Aanname | De plug-in is geïnstalleerd, zichtbaar in de sidebar van Lotus Notes en de plug-in bevat een module |
| Preconditie | De filter van de plug-in is ingeschakeld |
| Postconditie | De filter van de plug-in is uitgeschakeld |
| Omschrijving | De gebruiker klikt met de rechtermuisknop op de titel van de plug-in. Daar kan geklikt worden op filter uitschakelen. |
| Uitzondering | Geen |



Afbeelding 1.7: scenario "informatie doorzoeken"

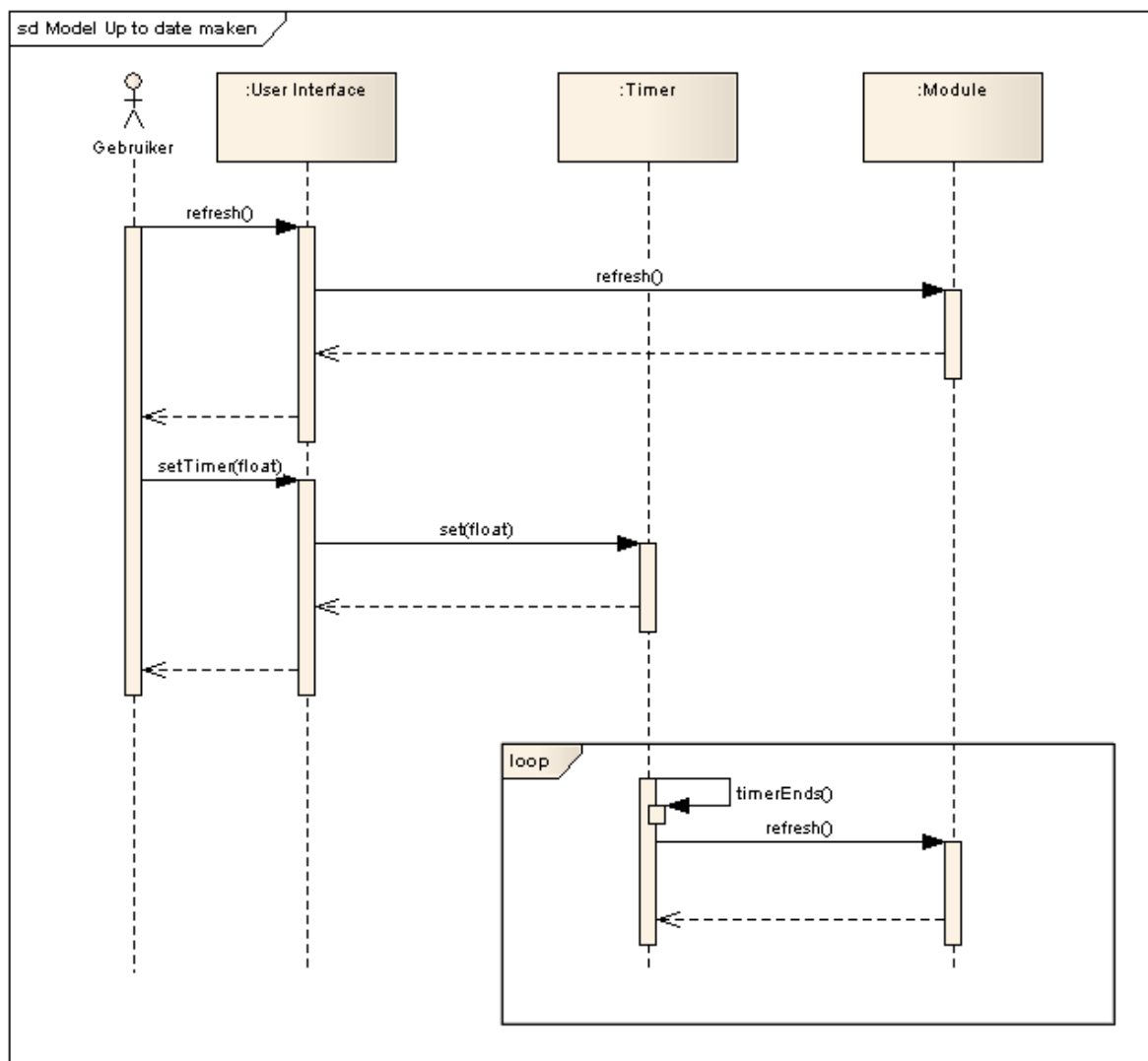
1.2.4 Up to date maken



Afbeelding 1.8: use-case-diagram "up to date maken"

| Naam | Timer instellen |
|---------------------|---|
| Samenvatting | De timer die regelt wanneer de plug-in vernieuwd wordt is ingesteld. |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes |
| Preconditie | Geen |
| Postconditie | De timer die regelt wanneer de plug-in vernieuwd wordt is ingesteld |
| Omschrijving | De gebruiker klikt met de rechtermuisknop op de titel van de plug-in. Daar staat dan een optie om timer van de plug-in in te stellen. |
| Uitzondering | Geen |

| | |
|---------------------|---|
| Naam | Vernieuwen |
| Samenvatting | De informatie in de plug-in wordt vernieuwd |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes |
| Preconditie | De plug-in bevat een module |
| Postconditie | De plug-in geeft de informatie weer die op dat moment in de database staat |
| Omschrijving | De gebruiker kan op een knop klikken in de header van de plug-in. De plug-in wordt dan vernieuwd De timer loopt periodiek af. Als de timer afloopt wordt de plug-in vernieuwd. |
| Uitzondering | Geen |



Afbeelding 1.5: scenario "up to date maken"

2 Eisen

Met de termen 'must have', 'should have', 'could have' of 'won't have' is aan te geven of er per moment nog een bepaalde speelruimte is of juist niet. Deze werkwijze is bekend onder de naam 'MoSCoW'. De termen hebben de volgende betekenis:

- Must have (M): moet op dat tijdstip gerealiseerd zijn
- Should have (S): zou op dat tijdstip gerealiseerd moeten zijn maar een tijdstip later is acceptabel
- Could have (C): mag (als tijd en geld over is) op het tijdstip gerealiseerd zijn maar moet niet.
- Won't have (W): zal bij het bereiken van het tijdstip niet gerealiseerd zijn.

In dit project wordt gebruik gemaakt van deze methode. De prioriteiten staan achter elke eis. Ook staat er op welk tijdstip de eis van toepassing is, increment één (I1) of increment twee (I2).

In bijlage 1 staat een tabel met de nummers van de eisen en daarbij bij wie ze vandaan komen en wanneer ze vastgelegd zijn.

| Eisen | | Prio | Tijd | Nr. | |
|-------------|-----------------|--|------|-----|----|
| Functioneel | | | | | |
| | Informatie | | | | |
| | | De gebruiker kan een module per plug-in selecteren. | M | I1 | 1 |
| | | De gebruiker kan informatie inzien over de module die geselecteerd is. | M | I1 | 2 |
| | | De gebruiker kan de plug-in verbergen. | S | I2 | 3 |
| | Filter | | | | |
| | | Een geselecteerde module heeft een filter | M | I2 | 32 |
| | | De gebruiker kan de filter inschakelen. | M | I2 | 4 |
| | | De gebruiker kan de filter uitschakelen. | M | I2 | 5 |
| | | De gebruiker kan de filter instellen. | M | I2 | 6 |
| | (de)installatie | | | | |
| | | De gebruiker kan de plug-in installeren. | M | I1 | 7 |
| | | De gebruiker kan de plug-in deïnstalleren. | M | I1 | 8 |
| | Sorteren | | | | |
| | | De gebruiker kan informatie sorteren. | S | I2 | 9 |
| | Zoeken | | | | |
| | | De gebruiker kan informatie zoeken | S | I2 | 10 |
| | Vernieuwen | | | | |
| | | De plug-in heeft een timer | M | I1 | 33 |
| | | De gebruiker kan de plug-in vernieuwen. | M | I1 | 11 |
| | | De plug-in wordt per interval vernieuwd. | M | I2 | 12 |
| | | De gebruiker kan interval van vernieuwen instellen. | M | I2 | 13 |
| | Gemak | | | | |
| | | De plug-in is in de Nederlandse taal. | M | I1 | 14 |
| | | De configuratie van de filter wordt onthouden | C | I2 | 29 |

| | | | | |
|--|--|---|----|----|
| | als Lotus Notes afgesloten wordt en als een andere module geselecteerd wordt | | | |
| | De instelling van de timer worden onthouden als Lotus Notes afgesloten wordt | S | I2 | 30 |
| | De instellingen van de weergave worden per module onthouden als Lotus Notes afgesloten wordt en als een andere module geselecteerd wordt | C | I2 | 31 |

Niet-functioneel

| | | | | |
|--|---|---|----|----|
| | Performance | | | |
| | De plug-in geeft informatie snel weer. <i>Snel = sneller dan huidige navigatie naar informatie</i> | M | I1 | 15 |
| | De plug-in is snel geïnstalleerd <i>Snel = korter dan 3 minuten</i> | M | I1 | 16 |
| | De plug-in is snel gedeïnstalleerd <i>Snel = korter dan 3 minuten</i> | M | I1 | 17 |
| | Weergave | | | |
| | De gebruiker kan een weergave selecteren. | S | I2 | 18 |
| | De gebruiker kan een weergave aanpassen. | S | I2 | 19 |
| | Uiterlijk | | | |
| | De user-interface van de plug-in moet aansluiten bij CSP. | M | I1 | 20 |
| | De user-interface van de plug-in moet aansluiten bij Lotus Notes. | M | I1 | 21 |
| | De plug-in past in de sidebar van Lotus Notes. | M | I1 | 22 |
| | Platform | | | |
| | De plug-in werkt op het platform IBM Lotus Notes en Domino 8.5. | M | I1 | 23 |
| | De plug-in werkt op de versie 8 van CSP. | M | I1 | 24 |
| | De plug-in werkt op windows 2003 en hoger, 32 en 64 bits versie | M | I1 | 25 |
| | Generiek | | | |
| | De plug-in werkt met alle modules in CSP die plug-in enabled zijn. | M | I2 | 26 |
| | De plug-in werkt in nog te maken modules van CSP die plug-in enabled zijn. | S | I2 | 27 |
| | Gemak | | | |
| | De gebruiker kan de te maken plug-in meerdere keren in de sidebar van Lotus Notes installeren | M | I1 | 28 |

3 Onthouden van instellingen

Voor de gebruikersvriendelijkheid is het goed dat tijdens het werken met de plug-in verschillende instellingen onthouden worden en verschillende instellingen juist niet. Als eerste wordt de filter onthouden. Een filter is gekoppeld aan een module, want elke module heeft weer andere informatie om weer te geven. De gebruiker kan de filter instellen. Als daarna een andere module geselecteerd wordt worden de ingestelde waarden van de filter van de verwijderde module onthouden. Als deze module later weer wordt geselecteerd dan staan de ingestelde gegevens nog in de instellingen van de filter. Het kan ook zijn dat Lotus Notes opnieuw opgestart wordt, dan staan de instellingen van de filter ook nog zoals ze waren op het moment dat Lotus Notes afgesloten werd.

De timer kan ook ingesteld worden, deze waarde wordt ook onthouden. Dus als Lotus Notes opnieuw wordt opgestart dan heeft de timer nog steeds de ingestelde waarden.

De gebruiker kan ook een weergave kiezen. Ook de weergave is gebonden aan een module. De geselecteerde weergave wordt onthouden, op dezelfde manier als de instellingen van de filter.

Het resizen, minimaliseren en maximaliseren wordt niet onthouden. De installatie, deinstallatie en het verbergen en tonen wordt, vanzelfsprekend, wel onthouden.

Bijlagen

1. Afkomst eisen

| Afkomst | Nr. |
|--|-----|
| Nico Scheppink (NIS), mondeling bij interview over eisen | 1 |
| NIS, schriftelijk bij de opdrachtoomschrijving | 2 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 3 |
| NIS, telefonisch bij de opdrachtoomschrijving | 4 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 5 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 6 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 7 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 8 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 9 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 10 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 11 |
| NIS, mondeling bij bespreken van plan van aanpak | 12 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 13 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 14 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 15 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 16 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 17 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 18 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 19 |
| NIS, mondeling bij bespreken van plan van aanpak | 20 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 21 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 22 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 23 |
| NIS, schriftelijk bij de opdrachtoomschrijving | 24 |
| NIS, mondeling bij interview over eisen (02-03-2010) | 25 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 26 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 27 |
| NIS, mondeling bij interview over eisen (02-03-2010) | 28 |
| NIS, mondeling bij interview over eisen (02-03-2010) | 29 |
| NIS, mondeling bij interview over eisen (02-03-2010) | 30 |
| NIS, mondeling bij interview over eisen (02-03-2010) | 31 |
| NIS, mondeling bij interview over eisen (08-03-2010) | 32 |
| NIS, mondeling bij interview over eisen (08-03-2010) | 33 |

6 Ontwerp increment 1

Ontwerp increment 1

Generieke sidebar plug-in

voor IBM Lotus Notes & Domino

AedifiComm

| | |
|---------------------|--------------------------------|
| Opdrachtgever: | F.J.C. Spruit |
| Bedrijfsmentor: | N.G. Scheppink |
| Begeleider: | D.R. Stikkolorum |
| Expert: | A.G.P. Pronk |
| Afstudeerder: | Chris den Arend |
| Email afstudeerder: | Chris.den.Arend@aedificomm.com |
| Plaats: | Barendrecht |
| Datum: | 07-04-2010 |
| Versie: | 0.4 |



Inhoudsopgave

| | |
|--|----|
| Inleiding..... | 3 |
| 1 Platform overzicht | 4 |
| 1.1 Lotus Notes | 4 |
| 1.2 IBM Lotus Expeditor | 4 |
| 1.3 Eclipse | 6 |
| 1.4 CSP | 7 |
| 2 Architectuur van plug-in | 8 |
| 2.1 Packagediagram | 8 |
| 2.2 Klassediagram..... | 9 |
| 2.2.1 Context Service | 10 |
| 2.2.2 Plug-in | 11 |
| 2.2.3 Eclipse | 12 |
| 3 Installatie | 14 |
| 4 Bronnen | 15 |
| 4.1 Boeken | 15 |
| 4.2 Websites..... | 15 |
| 4.3 Overige | 15 |
| Bijlagen..... | 16 |
| 1 Complete klassediagram..... | 16 |
| 2 Klassediagram van Context Service..... | 16 |
| 3 Stappenplan updatesite (1)..... | 16 |
| 4 Stappenplan updatesite (2)..... | 16 |
| 5 Notes plug-in development..... | 16 |
| 6 DTD van installatie XML | 16 |

Inleiding

Dit document beschrijft het ontwerp van increment één van het project Generieke sidebar plug-in. Het ontwerp is nodig voor de ontwikkelaar om uit te zoeken hoe het systeem gemaakt gaat worden. Daarnaast is het ontwerp nuttig voor partijen die eventueel in de toekomst verder willen met het resultaat van dit project. Aan de hand van de tabellen kan dan snel worden gezien hoe het systeem gemaakt is. De opdrachtgever heeft ervaring met het maken van ontwerpen, dus aan de hand van het ontwerp kan hij zien of het project de goede kant op gaat.

Als eerste wordt besproken met welke platformen gewerkt gaat worden. Dat is de basis om te beginnen met ontwikkelen. Daarna wordt besproken hoe de plug-in opgebouwd wordt aan de hand van een klassediagram.

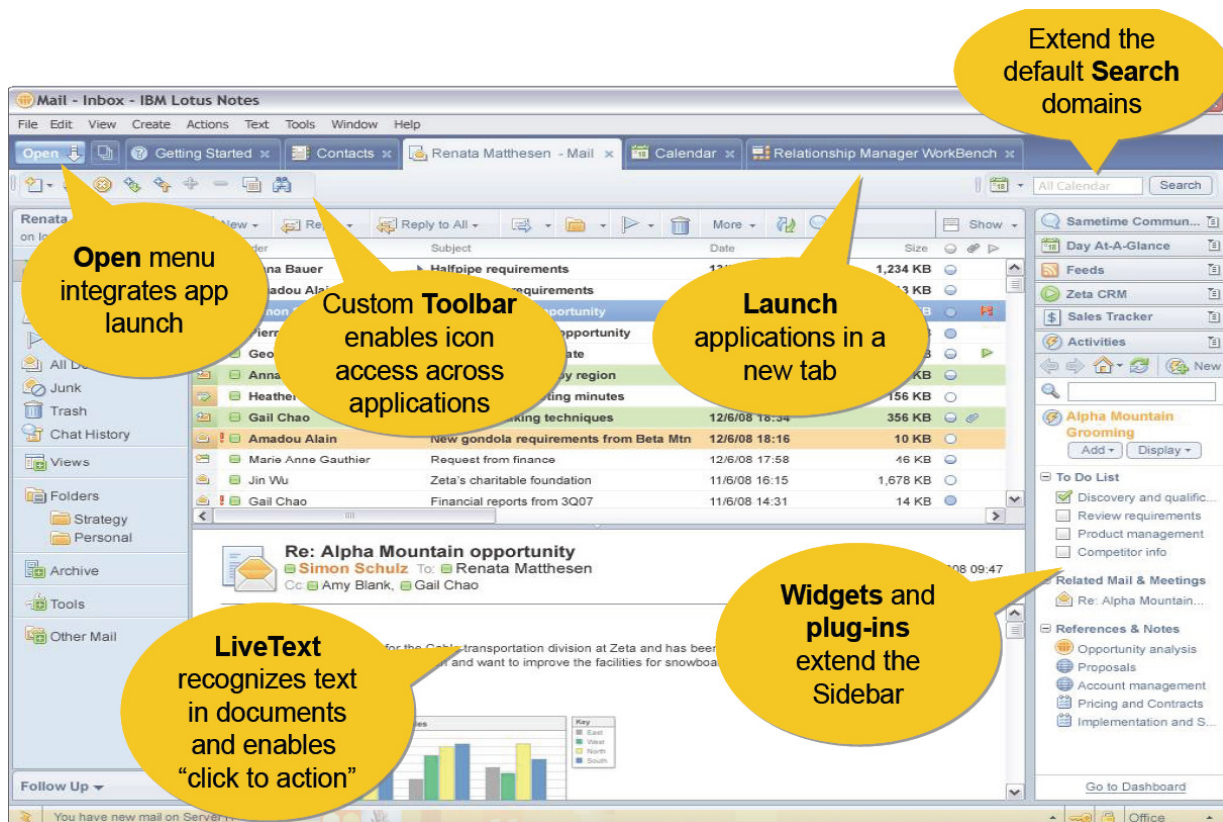
Tijdens het ontwerpen is gebruik gemaakt van de boeken [Warner06] en [Pender03].

1 Platform overzicht

In dit hoofdstuk wordt beschreven op welke platformen ontwikkeld wordt en op welke manier de platformen met elkaar samenwerken. Bij het schrijven van dit hoofdstuk is gebruik gemaakt van [Lotus1203].

1.1 Lotus Notes

Lotus Notes is een product van IBM. De software wordt vooral gebruikt voor email, agenda en documenten. In afbeelding 1.1 is een screenshot te zien van Lotus Notes versie 8.5.1. Aan de rechterkant van de afbeelding is de sidebar te zien. Deze sidebar is belangrijk voor het project, want daar komt de plug-in.



Afbeelding 1.1: screenshot Lotus Notes 8.5.1

In Lotus Notes staan een aantal widgets en plug-ins in de sidebar. Het doel daarvan is de meest voorkomende dingen die gebruikers doen en nodig hebben binnen drie klikken mogelijk te maken of weer te geven. Het streven van Lotus Notes is om hiermee 10% van de dingen die 90% van de mensen nodig hebben heel erg toegankelijk en makkelijk te maken. Sinds versie 8 is Lotus Notes gebaseerd op Lotus Expeditor.

1.2 IBM Lotus Expeditor

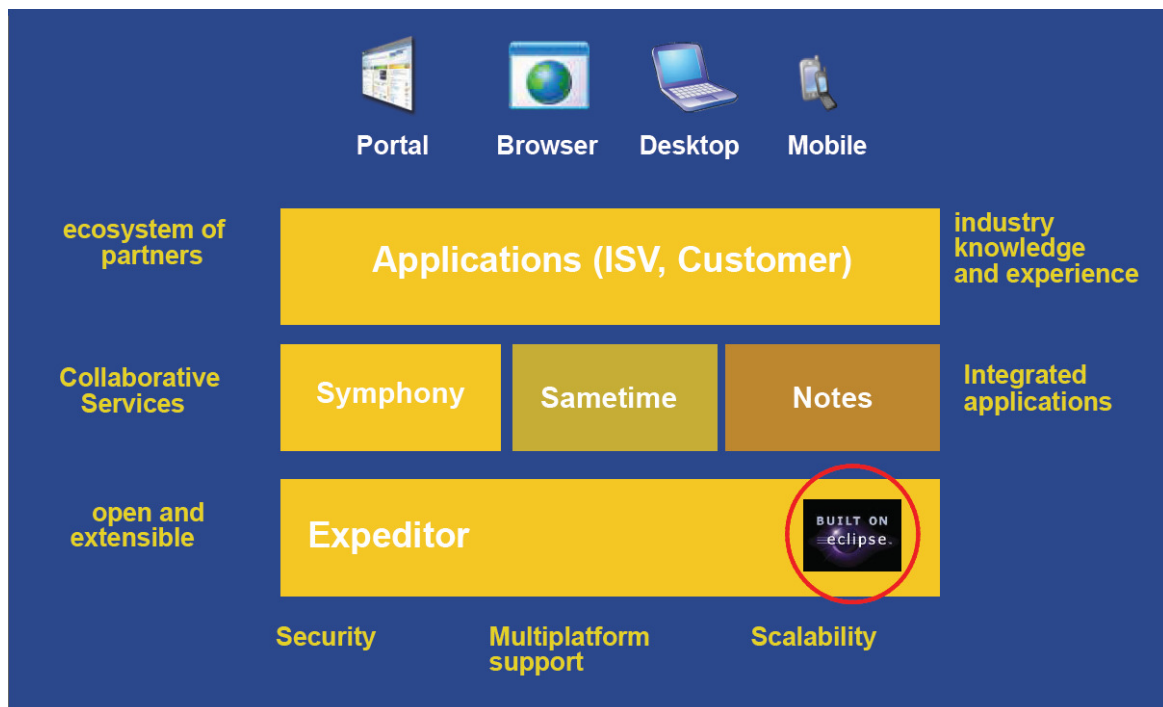
IBM Lotus Expeditor is een software framework van IBM Lotus Software en is een Service Oriented Architecture (SOA) platform. Het bevat verbindingscomponenten

voor veel toepassingen op laptops, desktops en mobiele systemen die via een server worden beheerd. De gebruikers kunnen lokaal toepassingen uitvoeren, ook als het systeem niet is verbonden met het netwerk. Als er wel verbinding is met het netwerk kan het systeem synchroniseren.

Om makkelijk toepassingen te ontwikkelen voor Lotus Expeditor kan de IBM Lotus Expeditor Toolkit gebruikt worden. De Toolkit kan geïntegreerd worden met Eclipse en maakt ontwikkelen, debuggen, testen en package mogelijk, voor IBM Lotus Expeditor, IBM Lotus Sametime 7.5.1 en IBM Lotus Notes vanaf versie 8. De Toolkit biedt de mogelijkheid om applicaties te ontwikkelen voor de volgende types:

- Eclipse Rich Client Platform (Lotus Notes)
- Eclipse embedded Rich Client Platform
- Web
- Embedded transaction
- Portlet
- Databases
- Messaging
- Web services

Dit project wordt gebruik gemaakt van de Toolkit, omdat deze kan samenwerken met de versies waarvoor de plug-in gemaakt wordt namelijk Lotus Notes versie 8 en hoger. Verder is het mogelijk om met de Expeditor Toolkit een plug-in te ontwikkelen en dat is precies wat in dit project gebeurt. Hoe dat kan staat in [1Domiclipse0803]. De plug-in wordt gemaakt voor een Eclipse Rich Client Platform, namelijk Lotus Notes. Voor het project wordt versie 6.2.1 van de Toolkit gebruikt. In afbeelding 1.2 is te zien op welke manier Lotus, Eclipse en de Expeditor Toolkit met elkaar samenwerken.



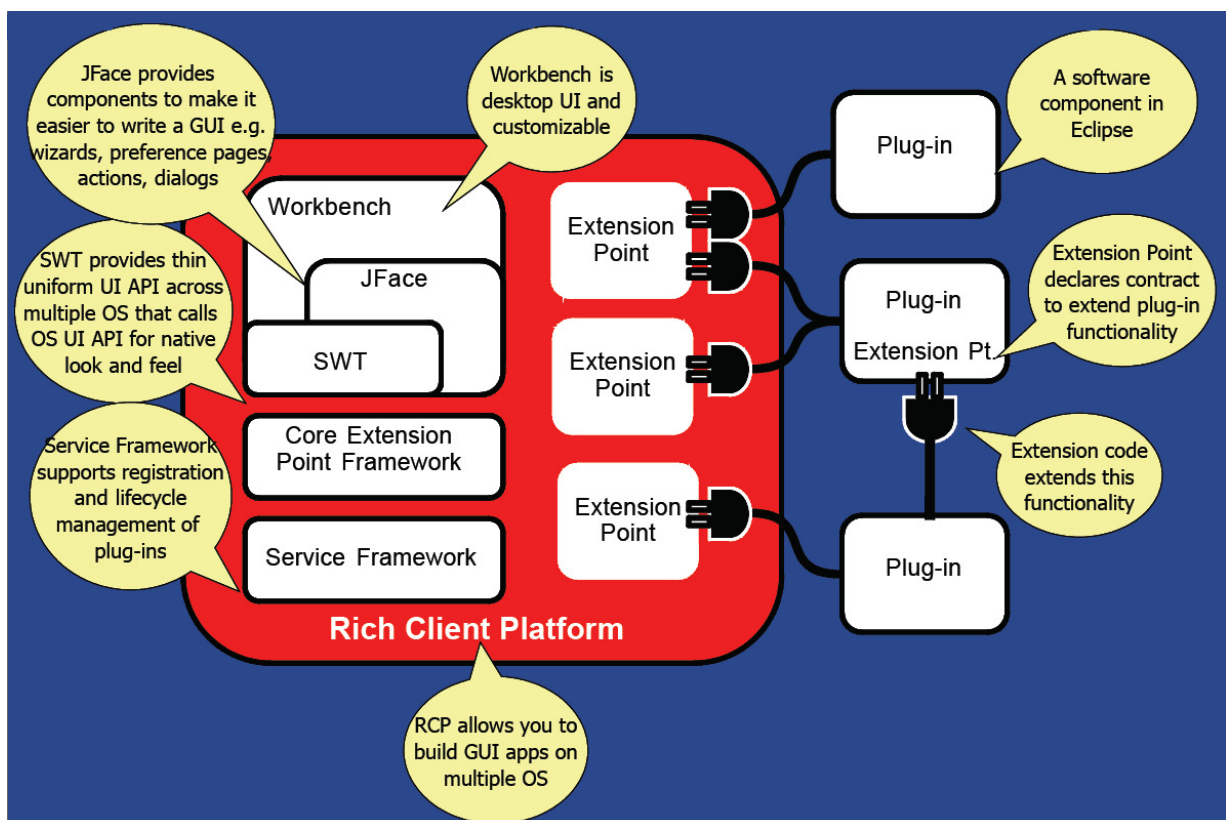
Afbeelding 1.2: Lotus Clients met Eclipse

1.3 Eclipse

Eclipse is een op Java gebaseerd open source ontwikkelplatform. Eclipse bestaat uit een standaard set van plug-ins, waaronder Java Development Tools(JDT). de gebruiker kan deze set zelf uitbreiden. Het feit dat Eclipse in Java geschreven is betekent niet dat andere talen uitgesloten zijn. Er zijn plug-ins voor Eclipse beschikbaar die het mogelijk maken om bijvoorbeeld in C of C++ te ontwikkelen. Tijdens dit project wordt wel in Java ontwikkeld om de volgende redenen:

- Lotus Notes is ook in Java geschreven, dus om daar goed op aan te sluiten is Java een geschikte programmeertaal
- Met de Expeditor kan een plug-in ontwikkeld worden, deze moet in Java geschreven worden
- AedifiComm wil graag dat de plug-in in Java geschreven wordt

In afbeelding 1.3 is de architectuur van Eclipse te zien.



Afbeelding 1.3: Eclipse platform architectuur

Voor dit project wordt Eclipse versie 3.4.2 gebruikt en dan het pakket "IDE for Java EE Developers". Deze keuze is gemaakt, omdat de Expeditor Toolkit goed op dat pakket past.

1.4 CSP

AedifiComm heeft het softwarepakket CSP ontwikkeld. CSP biedt ondersteuning voor werkprocessen, en wordt vooral gebruikt door accountants en belastingadviseurs. Het pakket is gebaseerd op Lotus Notes. Dat betekent dat de sidebar die in paragraaf 1.1 besproken is ook in CSP te vinden is, zoals te zien is in afbeelding 1.4. Daardoor is de plug-in te zien in CSP.

The screenshot shows the AedifiComm CSP interface. The left sidebar contains a 'Modules' menu with options: Relaties, Personeel, Post, Dossier, and Tools. Below this is a 'Verjaardagen' (Birthdays) section. The main area is divided into several panels. The top panel, 'Mijn Activiteiten' (My Activities), lists tasks with dates and descriptions, such as 'HHS - Algemeen' and 'AFSTUDEREN B.V. - Algemeen'. The 'Groepsagenda' (Group Agenda) for Friday, March 19, 2010, shows a schedule from 09:00 to 19:00 with events like 'Dossier review Kwakernaak (kantoor)' and 'vrijdagmiddagborrel (stankroeg)'. The 'Nieuwsberichten' (News) section contains several news items dated 18-06-2009. The 'Mijn Favorieten' (My Favorites) section lists contacts like 'AedifiComm', 'Apotheek Kwaakernaak', and 'Belastingdienst Nederland' with their respective phone numbers. The right sidebar includes 'Sametime Contacts', 'My Widgets', and a calendar for March 2010.

Afbeelding 1.4: CSP

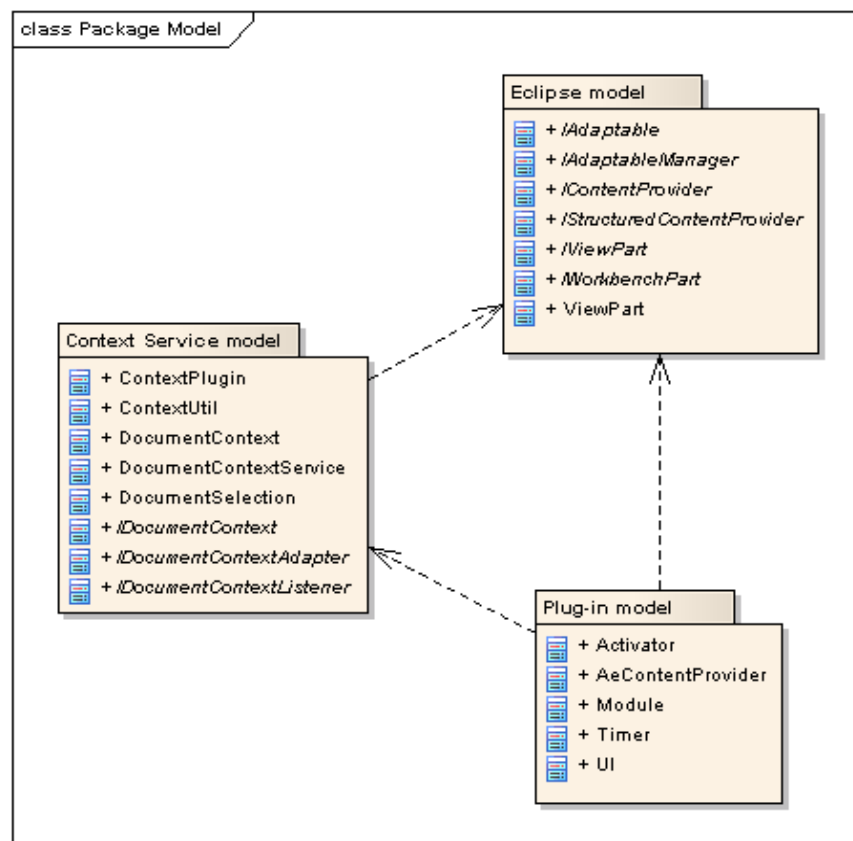
2 Architectuur van plug-in

In dit hoofdstuk wordt het ontwerp van de plug-in besproken aan de hand van een packagediagram en een klassediagram. Als eerste wordt het packagediagram weergegeven, daarna het klassediagram.

2.1 Packagediagram

Een packagediagram bestaat uit packages en een package is een deel van een klassediagram. Door het klassediagram op te delen in verschillende groepen klassen ontstaan packages. Met een packagediagram wordt een globale weergave van het systeem gegeven. Ook wordt duidelijk hoe de verhoudingen tussen packages zijn. Voor dit project is een packagediagram gebruikt om de verhoudingen tussen de plug-in en bestaande code die gebruikt wordt weer te geven.

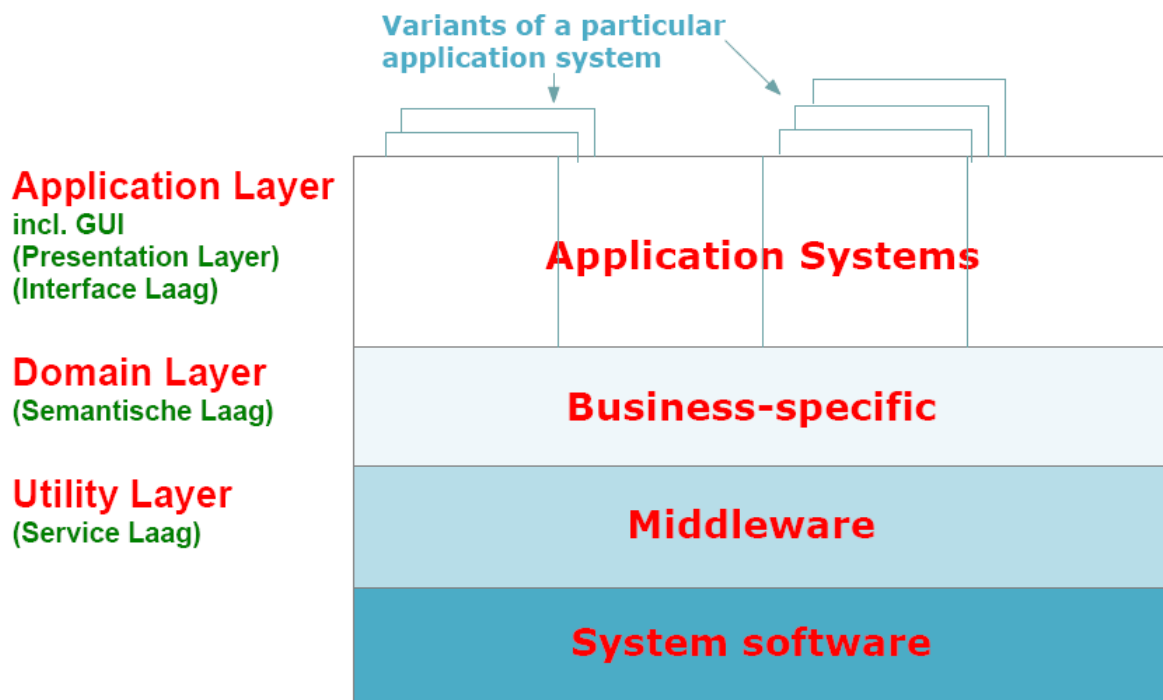
In hoofdstuk 1 is uitgelegd dat Lotus Notes gebaseerd is op Eclipse en CSP op Lotus Notes. Eclipse bestaat uit verschillende lagen en biedt plug-ins de mogelijkheid om te luisteren naar wat er gebeurt in deze lagen, informatie daarover is te vinden op [IBM1503]. Voor Eclipse is een plug-in te downloaden die het toegankelijker maakt om bij te houden wanneer de gebruiker in Lotus Notes van document wisselt en wat er in de documenten staat. Deze plug-in heet Context Service en wordt gebruikt voor de plug-in van dit project. In afbeelding 2.1 is het packagediagram te zien, de package Context Service model bevat de klassen van Context Service.



Afbeelding 2.1: package diagram

Naast het gebruik van context.service heeft Eclipse zelf een grote verzameling van klassen die gebruikt kunnen worden. Op [Eclipse1503] staat die verzameling, Eclipse noemt het Eclipse Platform API Specification. In het packagediagram is een package Eclipse model opgenomen. De bedoeling van die package is om de belangrijkste klassen die uit Eclipse gebruikt worden te laten zien. Daarnaast staat in het diagram de package Plug-in model, die package is de plug-in die gemaakt wordt tijdens dit project.

De package Plug-in model is afhankelijk van Context Service model en van verschillende klassen uit Eclipse. Context Service model is op zijn beurt weer afhankelijk van verschillende klassen uit Eclipse.



Afbeelding 2.2: lagenstructuur

In het diagram zijn daardoor verschillende lagen te herkennen, de Plug-in is de interfacelaag en het minst stabiel. De Context Service is abstracter en stabiel dan de Plug-in en valt onder de semantische laag. De meest abstracte package is Eclipse, deze bevat veel interfaces die geïmplementeerd worden in de twee andere packages van het model. Eclipse valt onder de semantische laag en de service laag, omdat het componenten biedt die ontwikkelaars in applicaties in kunnen vullen. In afbeelding 2.2 is de lagenstructuur te zien. Meer informatie over lagen is te vinden in [Bosch08].

2.2 Klassediagram

In deze paragraaf komen de klassediagrammen aan bod. Als eerste wordt het diagram van de Context Service besproken, daarna het diagram van de te maken

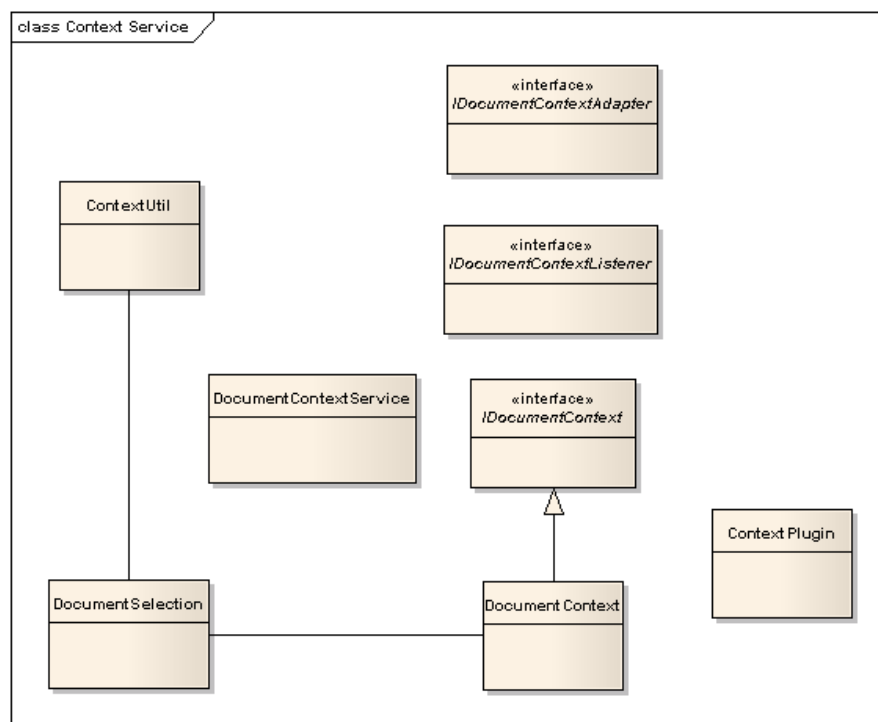
plug-in en klassen uit Eclipse die gebruikt worden door de plug-in of Context Service.

Een klassediagram beschrijft hoe een systeem is opgebouwd. Welke klassen het bevat en welke operaties en data een klasse heeft. De bedoeling is dat door het klassediagram al nagedacht wordt over de opbouw van het systeem voordat het systeem geïmplementeerd wordt. Daarnaast maakt een klassediagram het systeem duidelijk voor iemand die er later mee te maken krijgt.

2.2.1 Context Service

In bijlage 2 staat het complete klassediagram van alle packages en in bijlage 3 staat het volledige klassediagram van de package Context Service. In afbeelding 2.3 staan de klassen van Context Service en hun afhankelijkheden. De Context Service bevat de volgende klassen:

- DocumentContextService
- DocumentSelection
- ContextUtil
- DocumentContext
- ContextPlugin
- IDocumentContextAdapter
- IDocumentContextListener
- IDocumentContext



Afbeelding 2.3: klassediagram context service

DocumentContextService

Er bestaan in Eclipse twee functies om documenten te selecteren. In DocumentContextService worden beide functies aangeboden. De eerste is addSelectionListener(), deze functie reageert als een ander document wordt geselecteerd en als binnen een document iets geselecteerd wordt. De tweede functie is addListener, die reageert alleen als een ander document wordt geselecteerd. Er wordt een nieuw document geselecteerd als de URI van het geselecteerde document anders is dan de URI van het huidige document.

DocumentSelection

In de klasse worden verzamelingen van documenten bijgehouden. Elk document bevat informatie over zichzelf door middel van een IDocumentContext object.

ContextUtil

Deze klasse bevat een functie die het mogelijk maakt om een object te vertalen naar een adapter.

DocumentContext

Hier wordt gebruik gemaakt van de klasse ContextUtil. DocumentContext haalt de titel, URI, icoon en de eigenschappen van het geselecteerde object op. Als eerste wordt de IURIProvider geraadpleegd, als niet alles over het geselecteerde document te vinden is wordt de IWorkbenchAdapter gebruikt. Als er geen URI is dan wordt geprobeerd het object aan te passen aan de URI klasse. Als laatste worden eigenschappen uit IPropertySource gehaald. Als dat niet lukt wordt terug gegaan naar de huidige view.

ContextPlugin

Deze klasse creëert een instantie van de plug-in Context Service. Er zitten functies in die worden aangeroepen als de plug-in wordt gestart en als de plug-in wordt gestopt.

2.2.2 Plug-in

In afbeelding 2.4 is het klassediagram van de te maken plug-in te zien. De plug-in bevat de volgende klassen.

- UI
- Timer
- Module
- AeContentProvider
- Activator

UI

In de klasse UI wordt de plug-in opgebouwd. De form, sectie en tabel binnen de plug-in worden gedefinieerd.

Timer

De timer loopt eens in de zoveel tijd af, als dat gebeurt dan wordt de plug-in gerefreshed. De gebruiker kan de timer ook instellen, dus aangeven hoelang het duurt voordat de timer afloopt.

Module

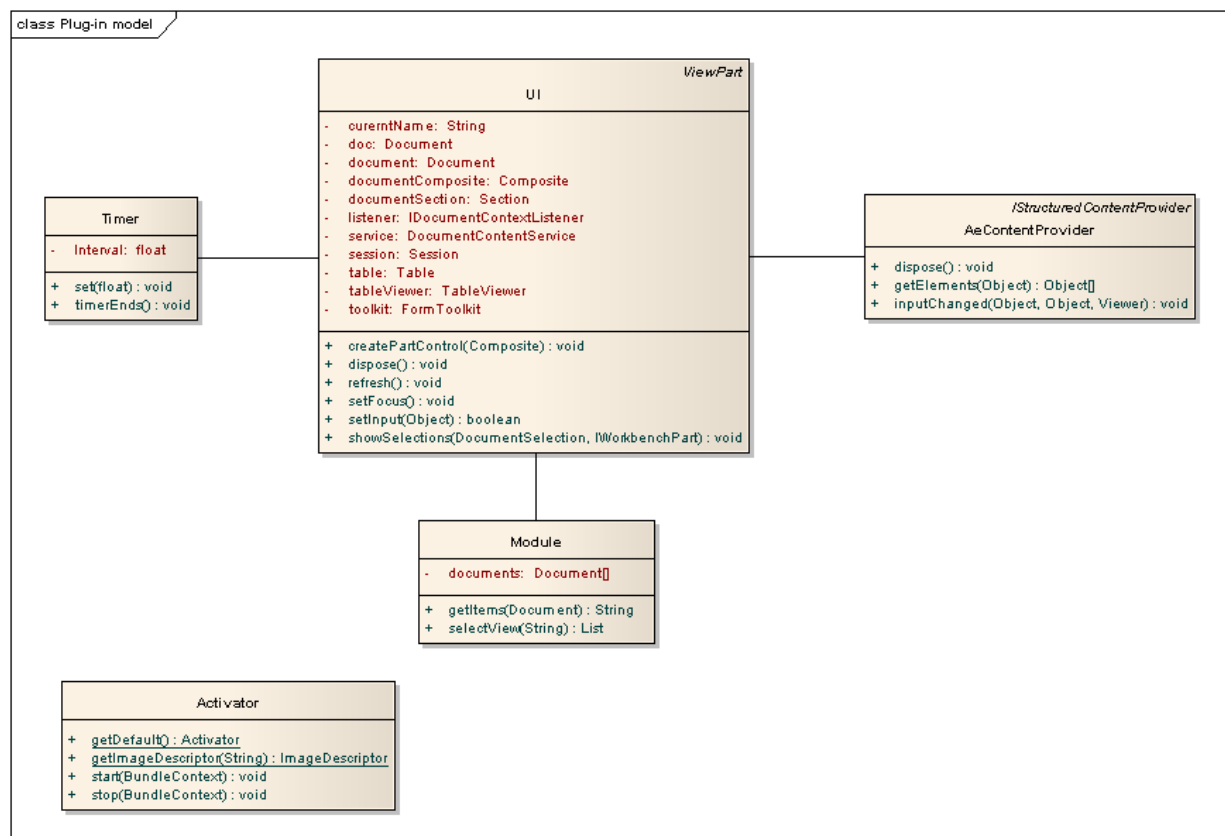
Hier wordt bepaald wat er in de UI te zien is. In CSP bestaan databases en documenten, deze kunnen op verschillende manieren worden weergegeven. Deze klasse kijkt op welke manier de informatie in CSP wordt weergegeven en gebruikt de informatie over die weergave om de items aan te geven die in de plug-in te zien moeten zijn.

AeContentProvider

De AeContentProvider geeft aan wat er in elke kolom van de tabel moet staan. De functie `getElements` krijgt het object dat weergegeven moet worden, daar haalt hij de informatie uit die weergegeven moet worden en zet deze in een array.

Activator

De klasse Activator is te vergelijken met de klasse `ContextPlugin` van de package `Context Service`. De klasse creëert een instantie van de plug-in en heeft functies die worden aangeroepen als de plug-in gestart of gestopt wordt.



Afbeelding 2.4: klassediagram plug-in

2.2.3 Eclipse

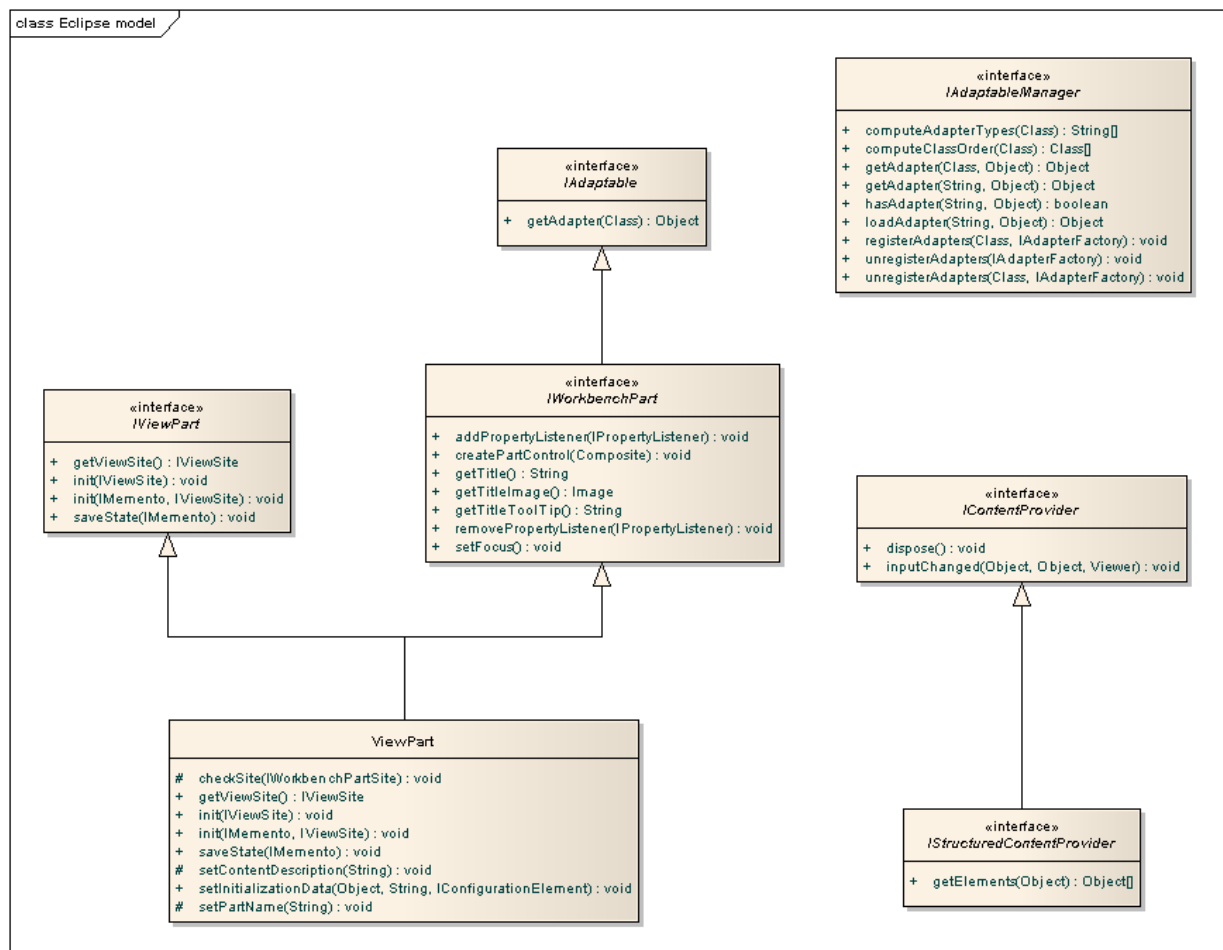
Het klassediagram dat bij de package `Eclipse` hoort is te zien in afbeelding 2.5. Het is niet nodig om alle klassen te bespreken, want de meeste klassen zijn interfaces.

Sommige functies uit deze interfaces worden geïmplementeerd in klassen die in Context Service of Plug-in zitten.

De klassen IAdaptable, IWorkbenchPart, IViewPart en Viewpart hebben te maken met de weergave van de Plug-in. De klasse UI uit Plug-in is een subklasse van ViewPart.

De klassen IContentProvider en IStructuredContentProvider zijn superinterfaces voor AeContentProvider uit Plug-in. Deze zorgen in dit geval voor de goede invulling van de tabel in de plug-in.

IAdaptableManager en IAdaptable worden gebruikt om het huidig geselecteerde document in CSP of Lotus Notes te verkrijgen.



Afbeelding 2.5: klassediagram eclipse

3 Installatie

De gemaakte plug-in moet geïnstalleerd worden in de sidebar van Lotus Notes. Dat kan op twee manieren, voor beide manieren is een updatesite in Lotus Notes nodig, waar de plug-in geplaatst wordt. In Eclipse moet van het project een "feature project" gemaakt worden en van dat "feature project" moet weer een "updatesite project" gemaakt worden. Hoe dat gerealiseerd wordt staat uitgelegd in [2Domiclipse0803] en [3Domiclipse0803].

Als de updatesite gemaakt is kan er een XML gemaakt worden die verwijst naar de updatesite. De gebruiker kan de XML in de sidebar(my widgets) slepen en dan wordt de plug-in geïnstalleerd. IBM heeft een Document Type Definition (DTD) gemaakt voor deze XML. In een DTD staat gedefinieerd welke elementen in de XML voorkomen, waar deze aan moeten voldoen en wat de kenmerken van het element zijn. De DTD met uitleg voor de XML van de installatie is te vinden in bijlage 7 en op [IBM1803]. De elementen die aangepast ingevuld moeten worden om naar de goede updatesite te wijzen zijn:

- De titel van de plug-in
- De URL van de updatesite
- Het id van de feature
- De naam van de plug-in

Een beschrijving over hoe de XML ingevuld moet worden staat op [IBM1203].

Als de plug-in op deze manier wordt geïnstalleerd kan de gebruiker de plug-in verbergen, tonen, resizen en verwijderen.

4 Bronnen

4.1 Boeken

[Pender03] Pender, T., UML Bible, 1^e druk, 2003.

[Warner06] Warner, J. En Kleppe, A., Praktisch UML, 3^e druk, 2006.

4.2 Websites

[1Domiclipse0803] Domiclipse,
<http://www.domiclipse.com/domiclipse/web.nsf/d6plinks/FRUX-7LV42L> (8 maart 2010)

[2Domiclipse0803] Domiclipse ,
<http://www.domiclipse.com/domiclipse/web.nsf/d6plinks/FRUX-7LV45T> (8 maart 2010)

In bijlage 4 is een pdf file van deze pagina opgenomen

[3Domiclipse0803] Domiclipse ,
<http://www.domiclipse.com/domiclipse/web.nsf/d6plinks/FRUX-7LV3TE> (8 maart 2010)

In bijlage 5 is een pdf file van deze pagina opgenomen

[Eclipse1503] Eclipse,
http://help.eclipse.org/help32/index.jsp?topic=/org.eclipse.jet.doc/references/xpath/PLUGINS_ROOT/org.eclipse.platform.doc.isv/reference/api/overview-summary.html(15 maart 2010)

[IBM1803] IBM,
<http://publib.boulder.ibm.com/infocenter/ledoc/v6r11/topic/com.ibm.rcp.tools.doc.admin/manifestDTD.html> (18 maart 2010)

[IBM1203] IBM,
https://infocenters.lotus.com/domino/index.jsp?topic=/com.ibm.help.domino.admin.doc/DOC/H_CREATE_EXTENSION_XML_STEPS.html (12 maart 2010)

[IBM1503] IBM, <http://www.ibm.com/developerworks/lotus/library/notes8-context/> (15 maart 2010)

[Lotus1203] Lotus, <http://www-10.lotus.com/ldd/lewiki.nsf/0/DDBB4F11B7C87436852575EC00749CD7/%24FILE/DeveloperTalk%20Notes%208.5%20Plugin%20Development.pdf> (12 maart 2010)

In bijlage 6 is een pdf file van deze pagina opgenomen

4.3 Overige

[Bosch08] Van den Bosch, H.C., Sheets OGO 100212, 2008. pp. 79-103

Bijlagen

1 Diagrammen

Deze bijlage wordt digitaal gestuurd, met dit document.

2 Complete klassediagram

Deze bijlage wordt digitaal gestuurd, met dit document.

3 Klassediagram van Context Service

Deze bijlage wordt digitaal gestuurd, met dit document.

4 Stappenplan updatesite (1)

Deze bijlage wordt digitaal gestuurd, met dit document.

5 Stappenplan updatesite (2)

Deze bijlage wordt digitaal gestuurd, met dit document.

6 Notes plug-in development

Deze bijlage wordt digitaal gestuurd, met dit document.

7 DTD van installatie XML

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- The root tag of the CA XML formatted file. -->
<!ELEMENT ibm-portal-composite (domain-object)+>

<!-- When specified with a name attribute of "com.ibm.rcp.installmanifest",
      this denotes that this is an provisioning manifest object. -->
<!ELEMENT domain-object (object-data)+>
<!ATTLIST domain-object name CDATA #FIXED "com.ibm.rcp.installmanifest">

<!-- Wrapper tag for the generic object data. -->
<!ELEMENT object-data (install)>

<!--This tag is required in the hierarchy -->
<!ELEMENT install (installfeature)+>

<!-- The list of features will be equivalent to InstallShield features
      which in the future can be presented as a list from which user can
      choose to selectively install certain features.
```

This tag will have the following attributes:

id

The id of the install feature.

name

The name of the install feature. The value of this attribute can be contributed to an offering installer's UI when the ReadDeploymentManifest bean is used in the offering installer.

restartPersonality (optional)

Used to indicate the personality with which the platform should be restarted once the provisioning of all the features has succeeded.

required (optional)

Denotes whether this is a required installfeature or can be optionally installed. Default value is false.

mergeaction (optional)

Selected values are "add" and "remove"

During upgrade scenario's, a pre-existing provisioning manifest will be merged with a new manifest, according to the following rules.

The mergeaction attribute is processed only from the new provisioning manifest, and is ignored in the old manifest.

add

forces an installfeature specified in the new provisioning manifest file to be retained in the merged result.

remove

forces an installfeature specified in the new provisioning manifest file to be removed from the merged result.

If the mergeaction attribute is not present, then the features within the installfeature are processed against the install feature of the identical id

-->

<!ELEMENT installfeature (requirements)>

<!ATTLIST installfeature

id CDATA #REQUIRED

name CDATA #IMPLIED

restartPersonality CDATA #IMPLIED

required CDATA #IMPLIED

mergeaction (add|remove) #IMPLIED>

<!--The requirements tag denotes a requirements block which will

contribute a list of required features to be used by the provisioning system. -->

<!ELEMENT requirements (feature)+>

<!-- This tag denotes a feature that is required by the platform for a given deployment.

This tag will have the following attributes:

id

The id of the feature to be provisioned

version

The version of the feature to be provisioned

match(optional)

Used to indicate the desired match rule.

According to the manifest definition in the Eclipse help, the match rules are as follows:

perfect

Dependent plug-in version must match exactly the specified version. If "patch" is "true", "perfect" is assumed and other values cannot be set.

equivalent

Dependent plug-in version must be at least at the version specified, or at a higher service level (major and minor version levels must equal the specified version).

compatible

Dependent plug-in version must be at least at the version specified, or at a higher service level or minor level (major version level must equal the specified version).

greaterOrEqual

Dependent plug-in version must be at least at the version specified, or at a higher service, minor or major level.

Match attributes are specified in both the provisioning manifest (user.xml or multiuser.xml) and in the feature itself.

Match attributes are specified in both the provisioning manifest

(install.xml) and in the feature itself.

The match attributes specified inside the feature apply to the feature requirements and plugin requirements for the feature. These are processed by the update manager to determine whether all requirements for the feature have been satisfied.

The match attributes specified in the provisioning manifest are used as a predetermination for installing the feature.

If not specified, the setting of the Eclipse preference `org.eclipse.update.core/org.eclipse.update.core.updateVersions` is used.

`url` (optional)

The URL to the update site where the feature can be found. An update site is any valid file URL which contains a site manifest file, a features folder, and a plugins folder.

Example: `url=http://www.myupdatehost.com/updateSite`

Example: `url=file:/C:/Program Files/application/updateSite`

"`${installer.root}`" is an optional token that is replaced by the installer when the manifest is processed prior to initial provisioning. The parameter is replaced with the directory from which the installer has launched, and can therefore only be used in conjunction with a relative update site path.

Example: "`${installer.root}`" = "`file:/D:/ProductCD/install/`"
`url="${installer.root}/../updates"` translates to
`file:/D:/ProductCD/../updates/`

If a URL is not specified, then it is expected that the `com.ibm.rcp.provisioning/updateSiteList` preference has been updated with a list of update sites from which features will be installed. During the installation process, the installer automatically provides an update site that is used if the feature does not otherwise have a url specified. For more information, see:

[6.1.1 Configuring feature search order](#)

6.1.2 [Configuring feature search order in 6.1.2](#)

`size`(optional)

The installed size (in kilobytes) of the feature. It is used by the install application to calculate the total size of the installed product. The size attribute is currently used only by the install application.

`download-size`(optional)

The download size (in kilobytes) of the feature. It is used by the install application to calculate the amount

of temporary space required during the install processing. The download-size attribute is currently used only by the install application.

shared(optional)

If set to true, the feature will be installed into the shared site (if no other feature version or affinity is satisfied). If false, the feature will be installed into the user site (if no other feature version or affinity is satisfied). The default setting will depend on the install configuration. If a multiuser install configuration is being used, the default is true. If a single user install configuration is being used, the default is false.

remove(optional)

Remove this feature. This attribute is deprecated. Specify action="uninstall" instead.

action(optional)

Selected values are as follows:

install

Install and configure the feature (default)

installonly

Install, but do not configure the feature

enable

Enable a feature that is currently installed

disable

Disable a feature that is currently installed

uninstall

Disable, then uninstall a feature

ignore

Perform no actions on this feature

mergeaction(optional)

Selected values are "add" and "remove"

During upgrade scenarios, a pre-existing provisioning manifest will be merged with a new manifest, according to the following rules.

The mergeaction attribute is processed only from the new provisioning manifest, and is ignored in the old manifest.

add

forces a feature specified in the new provisioning manifest file to be retained in the merged result.

remove

forces a feature specified in the new install manifest file to be removed from the merged result.

If the feature is present in the old manifest, and the feature is present in the new manifest

mergeaction attribute not specified

The feature is retained with the attributes defined in the new manifest

mergeaction = "add"

new feature is retained completely

mergeaction = "remove"

feature is removed

If the installfeature is present in the old manifest, and does not exist in the new manifest, it will be retained completely

If the installfeature is not present in the old manifest, it will only exist in the result if mergeaction="add" is specified

-->

<!ELEMENT feature EMPTY>

<!ATTLIST feature

id CDATA #REQUIRED

url CDATA #IMPLIED

version CDATA #REQUIRED

match (perfect|compatible|equivalent|greaterOrEqual) #IMPLIED

size CDATA #IMPLIED

download-size CDATA #IMPLIED

shared CDATA #IMPLIED

mergeaction (add|remove) #IMPLIED

action (install|installonly|enable|disable|uninstall|ignore)

"install"

remove CDATA "true" >

7 Testplan increment 1

Testplan

Generieke sidebar plug-in

voor IBM Lotus Notes & Domino

AedifiComm

| | |
|---------------------|---------------------------------|
| Opdrachtgever: | F.J.C. Spruit |
| Bedrijfsmentor: | N.G. Scheppink |
| Begeleider: | D.R. Stikkolorum |
| Expert: | A.G.P. Pronk |
| Afstudeerder: | Chris den Arend |
| Email afstudeerder: | Chris.den.Arend@aedificommm.com |
| Plaats: | Barendrecht |
| Datum: | 09-04-2010 |
| Versie: | 1.0 |



Inleiding

Tijdens het ontwikkelen van software is het belangrijk dat de software getest wordt, want dat komt de kwaliteit ten goede. In dit document staan verschillende cases om het prototype uit increment één te testen. De cases zijn gemaakt aan de hand van de eisen die in het document 'Increment 1' staan.

Na de testen wordt een testrapport gemaakt en dat rapport wordt meegenomen naar de analysefase van increment twee. Daar wordt het rapport gebruikt om te bepalen welke invloed de testen hebben op de eisen. Het testen wordt uitgevoerd op systemen waar Lotus Notes en Domino al geïnstalleerd is. Het installatiebestand van de plug-in staat op de server waart Lotus Notes gebruikt van maakt.

Windows XP (32 bit)

Case 1: Installatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker sleept de installatiefile (XML) in de sidebar van Lotus Notes, in het "my widgets" gedeelte.

Uitkomst:

De gebruiker kan aangeven of de plug-in geïnstalleerd moet worden. Als voor de installatie gekozen wordt begint deze en is de plug-in binnen 3 minuten geïnstalleerd. Nadat Lotus Notes opnieuw is opgestart is de plug-in zichtbaar in de sidebar en staat hij in "my widgets".

Test resultaat:

.....
.....
.....

Case 2: Informatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert in CSP de module "persoonlijke activiteiten".

Uitkomst:

De plug-in geeft informatie weer over "persoonlijke activiteiten" sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 3: Vernieuwen

Geslaagd: ja/nee

Beschrijving:

De gebruiker kan klikken op de knop "vernieuwen".

Uitkomst:

De plug-in wordt vernieuwd.

Test resultaat:

.....
.....
.....

Case 4: Gemak (1)

Geslaagd: ja/nee

Beschrijving:

De plug-in is in de Nederlandse taal.

Test resultaat:

.....
.....
.....

Case 5: Gemak (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker installeert de plug-in meerdere malen op de manier zoals beschreven in case 2 van dit testplan.

Uitkomst:

Na elke installatie is de plug-in opnieuw geïnstalleerd, naast plug-in(s) die al geïnstalleerd waren

Test resultaat:

.....
.....
.....

Case 6: Uiterlijk

Geslaagd: ja/nee

Beschrijving:

De user-interface sluit aan bij Lotus Notes en CSP.

Test resultaat:

.....
.....
.....

Case 7: Deinstallatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker klikt met de rechtermuisknop op de plug-in in "my widgets" en kiest remove.

Uitkomst:

De plug-in wordt binnen 3 minuten verwijderd, nadat Lotus Notes opnieuw is opgestart is de plug-in gedeïnstalleerd.

Test resultaat:

.....
.....
.....

Windows Vista (32 bit)

Case 1: Installatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker sleept de installatiefile (XML) in de sidebar van Lotus Notes, in het "my widgets" gedeelte.

Uitkomst:

De gebruiker kan aangeven of de plug-in geïnstalleerd moet worden. Als voor de installatie gekozen wordt begint deze en is de plug-in binnen 3 minuten geïnstalleerd. Nadat Lotus Notes opnieuw is opgestart is de plug-in zichtbaar in de sidebar en staat hij in "my widgets".

Test resultaat:

.....
.....
.....

Case 2: Informatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert in CSP de module "persoonlijke activiteiten".

Uitkomst:

De plug-in geeft informatie weer over "persoonlijke activiteiten" sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 3: Vernieuwen

Geslaagd: ja/nee

Beschrijving:

De gebruiker kan klikken op de knop "vernieuwen".

Uitkomst:

De plug-in wordt vernieuwd.

Test resultaat:

.....
.....
.....

Case 4: Gemak (1)

Geslaagd: ja/nee

Beschrijving:

De plug-in is in de Nederlandse taal.

Test resultaat:

.....
.....
.....

Case 5: Gemak (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker installeert de plug-in meerdere malen op de manier zoals beschreven in case 2 van dit testplan.

Uitkomst:

Na elke installatie is de plug-in opnieuw geïnstalleerd, naast plug-in(s) die al geïnstalleerd waren

Test resultaat:

.....
.....
.....

Case 6: Uiterlijk

Geslaagd: ja/nee

Beschrijving:

De user-interface sluit aan bij Lotus Notes en CSP.

Test resultaat:

.....
.....
.....

Case 7: Deinstallatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker klikt met de rechtermuisknop op de plug-in in "my widgets" en kiest remove.

Uitkomst:

De plug-in wordt binnen 3 minuten verwijderd, nadat Lotus Notes opnieuw is opgestart is de plug-in gedeïnstalleerd.

Test resultaat:

.....
.....
.....

Windows 7 (64 bit)

Case 1: Installatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker sleept de installatiefile (XML) in de sidebar van Lotus Notes, in het "my widgets" gedeelte.

Uitkomst:

De gebruiker kan aangeven of de plug-in geïnstalleerd moet worden. Als voor de installatie gekozen wordt begint deze en is de plug-in binnen 3 minuten geïnstalleerd. Nadat Lotus Notes opnieuw is opgestart is de plug-in zichtbaar in de sidebar en staat hij in "my widgets".

Test resultaat:

.....
.....
.....

Case 2: Informatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert in CSP de module "persoonlijke activiteiten".

Uitkomst:

De plug-in geeft informatie weer over "persoonlijke activiteiten" sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 3: Vernieuwen

Geslaagd: ja/nee

Beschrijving:

De gebruiker kan klikken op de knop "vernieuwen".

Uitkomst:

De plug-in wordt vernieuwd.

Test resultaat:

.....
.....
.....

Case 4: Gemak (1)

Geslaagd: ja/nee

Beschrijving:

De plug-in is in de Nederlandse taal.

Test resultaat:

.....
.....
.....

Case 5: Gemak (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker installeert de plug-in meerdere malen op de manier zoals beschreven in case 2 van dit testplan.

Uitkomst:

Na elke installatie is de plug-in opnieuw geïnstalleerd, naast plug-in(s) die al geïnstalleerd waren

Test resultaat:

.....
.....
.....

Case 6: Uiterlijk

Geslaagd: ja/nee

Beschrijving:

De user-interface sluit aan bij Lotus Notes en CSP.

Test resultaat:

.....
.....
.....

Case 7: Deinstallatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker klikt met de rechtermuisknop op de plug-in in "my widgets" en kiest remove.

Uitkomst:

De plug-in wordt binnen 3 minuten verwijderd, nadat Lotus Notes opnieuw is opgestart is de plug-in gedeïnstalleerd.

Test resultaat:

.....
.....
.....

8 Testrapport increment 1

Testrapport

Generieke sidebar plug-in

voor IBM Lotus Notes & Domino



| | |
|---------------------|---------------------------------|
| Opdrachtgever: | F.J.C. Spruit |
| Bedrijfsmentor: | N.G. Scheppink |
| Begeleider: | D.R. Stikkolorum |
| Expert: | A.G.P. Pronk |
| Afstudeerder: | Chris den Arend |
| Email afstudeerder: | Chris.den.Arend@aedificommm.com |
| Plaats: | Barendrecht |
| Datum: | 13-04-2010 |
| Versie: | 1.0 |



Inleiding

Tijdens het ontwikkelen van software is het belangrijk dat de software getest wordt, want dat komt de kwaliteit ten goede. In dit document staan verschillende cases om het prototype uit increment één te testen. De cases zijn gemaakt aan de hand van de eisen die in het document 'Increment 1' staan.

Na de testen wordt een testrapport gemaakt en dat rapport wordt meegenomen naar de analysefase van increment twee. Daar wordt het rapport gebruikt om te bepalen welke invloed de testen hebben op de eisen. Het testen wordt uitgevoerd op systemen waar Lotus Notes en Domino al geïnstalleerd is. Het installatiebestand van de plug-in staat op de server waart Lotus Notes gebruikt van maakt.

Windows XP (32 bit)

Case 1: Installatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker sleept de installatiefile (XML) in de sidebar van Lotus Notes, in het "my widgets" gedeelte.

Uitkomst:

De gebruiker kan aangeven of de plug-in geïnstalleerd moet worden. Als voor de installatie gekozen wordt begint deze en is de plug-in binnen 3 minuten geïnstalleerd. Nadat Lotus Notes opnieuw is opgestart is de plug-in zichtbaar in de sidebar en staat hij in "my widgets".

Test resultaat:

De plug-in is geïnstalleerd in 2 minuten en 4 seconden.

Case 2: Informatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert in CSP de module "persoonlijke activiteiten".

Uitkomst:

De plug-in geeft informatie weer over "persoonlijke activiteiten" sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

De plug-in geeft geen informatie weer over "persoonlijke activiteiten", er worden kolommen weergegeven in de pug-in die overeenkomen met de kolommen in "persoonlijke activiteiten".

Case 3: Vernieuwen

Geslaagd: ja/nee

Beschrijving:

De gebruiker kan klikt op de knop "vernieuwen".

Uitkomst:

De plug-in wordt vernieuwd.

Test resultaat:

De plug-in bevat geen knop vernieuwen.

Case 4: Gemak (1)

Geslaagd: **ja**/nee

Beschrijving:

De plug-in is in de Nederlandse taal.

Test resultaat:

Dat klopt.

Case 5: Gemak (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker installeert de plug-in meerdere malen op de manier zoals beschreven in case 2 van dit testplan.

Uitkomst:

Na elke installatie is de plug-in opnieuw geïnstalleerd, naast plug-in(s) die al geïnstalleerd waren

Test resultaat:

De Lotus Notes software geeft de volgende melding "The widget "Plug-in naam" already exists. Do you want to reinstall/upgrade the selected widget?"

Case 6: Uiterlijk

Geslaagd: **ja**/nee

Beschrijving:

De user-interface sluit aan bij Lotus Notes en CSP.

Test resultaat:

Dat klopt.

Case 7: Deinstallatie

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker klikt met de rechtermuisknop op de plug-in in "my widgets" en kiest remove.

Uitkomst:

De plug-in wordt binnen 3 minuten verwijderd, nadat Lotus Notes opnieuw is opgestart is de plug-in gedeïnstalleerd.

Test resultaat:

De plug-in is verwijderd in 33 seconden.

Windows Vista (32 bit)

Case 1: Installatie

Geslaagd: **ja**/~~nee~~

Beschrijving:

De gebruiker sleept de installatiefile (XML) in de sidebar van Lotus Notes, in het "my widgets" gedeelte.

Uitkomst:

De gebruiker kan aangeven of de plug-in geïnstalleerd moet worden. Als voor de installatie gekozen wordt begint deze en is de plug-in binnen 3 minuten geïnstalleerd. Nadat Lotus Notes opnieuw is opgestart is de plug-in zichtbaar in de sidebar en staat hij in "my widgets".

Test resultaat:

De plug-in is geïnstalleerd in 2 minuten en 38 seconden.

Case 2: Informatie

Geslaagd: **ja**/~~nee~~

Beschrijving:

De gebruiker selecteert in CSP de module "persoonlijke activiteiten".

Uitkomst:

De plug-in geeft informatie weer over "persoonlijke activiteiten" sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

De plug-in geeft geen informatie weer over "persoonlijke activiteiten", er worden kolommen weergegeven in de pug-in die overeenkomen met de kolommen in "persoonlijke activiteiten".

Case 3: Vernieuwen

Geslaagd: **ja**/~~nee~~

Beschrijving:

De gebruiker kan klikt op de knop "vernieuwen".

Uitkomst:

De plug-in wordt vernieuwd.

Test resultaat:

De plug-in bevat geen knop vernieuwen.

Case 4: Gemak (1)

Geslaagd: **ja**/nee

Beschrijving:

De plug-in is in de Nederlandse taal.

Test resultaat:

Dat klopt.

Case 5: Gemak (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker installeert de plug-in meerdere malen op de manier zoals beschreven in case 2 van dit testplan.

Uitkomst:

Na elke installatie is de plug-in opnieuw geïnstalleerd, naast plug-in(s) die al geïnstalleerd waren

Test resultaat:

De Lotus Notes software geeft de volgende melding "The widget "Plug-in naam" already exists. Do you want to reinstall/upgrade the selected widget?"

Case 6: Uiterlijk

Geslaagd: **ja**/nee

Beschrijving:

De user-interface sluit aan bij Lotus Notes en CSP.

Test resultaat:

Dat klopt.

Case 7: Deinstallatie

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker klikt met de rechtermuisknop op de plug-in in "my widgets" en kiest remove.

Uitkomst:

De plug-in wordt binnen 3 minuten verwijderd, nadat Lotus Notes opnieuw is opgestart is de plug-in gedeïnstalleerd.

Test resultaat:

De plug-in is verwijderd in 32 seconden.

Windows 7 (64 bit)

Case 1: Installatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker sleept de installatiefile (XML) in de sidebar van Lotus Notes, in het "my widgets" gedeelte.

Uitkomst:

De gebruiker kan aangeven of de plug-in geïnstalleerd moet worden. Als voor de installatie gekozen wordt begint deze en is de plug-in binnen 3 minuten geïnstalleerd. Nadat Lotus Notes opnieuw is opgestart is de plug-in zichtbaar in de sidebar en staat hij in "my widgets".

Test resultaat:

De plug-in is geïnstalleerd in 35 seconden.

Case 2: Informatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert in CSP de module "persoonlijke activiteiten".

Uitkomst:

De plug-in geeft informatie weer over "persoonlijke activiteiten" sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

De plug-in geeft geen informatie weer over "persoonlijke activiteiten", er worden kolommen weergegeven in de pug-in die overeenkomen met de kolommen in "persoonlijke activiteiten".

Case 3: Vernieuwen

Geslaagd: ja/nee

Beschrijving:

De gebruiker kan klikt op de knop "vernieuwen".

Uitkomst:

De plug-in wordt vernieuwd.

Test resultaat:

De plug-in bevat geen knop vernieuwen.

Case 4: Gemak (1)

Geslaagd: **ja**/nee

Beschrijving:

De plug-in is in de Nederlandse taal.

Test resultaat:

Dat klopt.

Case 5: Gemak (2)

Geslaagd: ja/**nee**

Beschrijving:

De gebruiker installeert de plug-in meerdere malen op de manier zoals beschreven in case 2 van dit testplan.

Uitkomst:

Na elke installatie is de plug-in opnieuw geïnstalleerd, naast plug-in(s) die al geïnstalleerd waren

Test resultaat:

De Lotus Notes software geeft de volgende melding "The widget "Plug-in naam" already exists. Do you want to reinstall/upgrade the selected widget?"

Case 6: Uiterlijk

Geslaagd: **ja**/nee

Beschrijving:

De user-interface sluit aan bij Lotus Notes en CSP.

Test resultaat:

Dat klopt.

Case 7: Deinstallatie

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker klikt met de rechtermuisknop op de plug-in in "my widgets" en kiest remove.

Uitkomst:

De plug-in wordt binnen 3 minuten verwijderd, nadat Lotus Notes opnieuw is opgestart is de plug-in gedeïnstalleerd.

Test resultaat:

De plug-in is verwijderd in 22 seconden.

9 Incrementplan increment 2

Increment 2

Generieke sidebar plug-in
voor IBM Lotus Notes & Domino

AedifiComm

| | |
|---------------------|--------------------------------|
| Opdrachtgever: | F.J.C. Spruit |
| Bedrijfsmentor: | N.G. Scheppink |
| Begeleider: | D.R. Stikkolorum |
| Expert: | A.G.P. Pronk |
| Afstudeerder: | Chris den Arend |
| Email afstudeerder: | Chris.den.Arend@aedificomm.com |
| Plaats: | Barendrecht |
| Datum: | 15-04-2010 |
| Versie: | 0.1 |



Inhoudsopgave

| | |
|----------------------------|---|
| Inleiding..... | 3 |
| 1 Resultaat..... | 4 |
| 1.1 De plug-in..... | 4 |
| 1.2 Documenten..... | 4 |
| 2 Eisen van prototype..... | 5 |
| 3 Planning..... | 7 |
| 4 Werkzaamheden | 8 |

Inleiding

In dit document wordt besproken wat de planning is voor increment twee van het project: ontwikkelen van een configureerbare en generieke plugin voor de lotus notes sidebar.

Er wordt ook verteld welke werkzaamheden worden uitgevoerd in increment twee. Dit document biedt uitbreidingen op de gedeelten over increment twee in het plan van aanpak en het programma van eisen.

Het document wordt aan het einde van de analysefase opgeleverd. Daar is voor gekozen, omdat de eisen dan bekend zijn, waardoor de werkzaamheden gedetailleerder in de planning van het increment gezet kunnen worden.

Het doel van dit verslag is uiteenzetten van de werkzaamheden in increment een. Daarbij hoort een planning en afbakening van de werkzaamheden. Als eerste wordt het beoogde resultaat van het increment behaald. Daarna de eisen, de planning en de werkzaamheden.

1 Resultaat

In dit hoofdstuk wordt besproken wat de geplande resultaten zijn van dit increment. Eerst komt de plug-in aan bod en vervolgens de documenten die worden opgeleverd.

1.1 De plug-in

Increment twee levert onder andere een plug-in op. De eisen waar de plug-in na increment twee aan moet voldoen staan in hoofdstuk drie van dit document. De volledige lijst van eisen staat in het programma van eisen.

Het doel van de plug-in staat in het plan van aanpak. In het tweede deel van dit project wordt veel aandacht besteed aan het generiek maken van de plug-in. Na increment één is de plug-in voor een deel generiek. De kolommen van de huidige geselecteerde module zijn te zien in de plug-in. Na increment twee moet de plug-in alleen de relevante kolommen van de huidige geselecteerde module weergeven. Daarnaast moet ook de inhoud van de module weergegeven worden in de plug-in. De gebruiker moet de module die op dat moment geselecteerd is ook vast kunnen zetten in de plug-in.

Verder bevat de plug-in functies om door informatie in de vastgezette module te zoeken of deze informatie te filteren of sorteren.

1.2 Documenten

Naast de plug-in worden in het tweede increment ook documenten opgeleverd.

Het programma van eisen dat in iteratie 1 gemaakt is wordt aangepast en aangevuld. Dat gebeurt met behulp van het prototype en de opgedane kennis over de gebruikte platformen.

Tijdens de ontwerpfase wordt een nieuw ontwerp gemaakt. Tijdens iteratie 1 werd duidelijk dat het van toegevoegde waarde is dat ook gedeelten van Eclipse en een gebruikte plug-in worden gemodelleerd. Dat komt ook terug in het model van iteratie twee.

Aan het einde van increment twee wordt de plug-in getest. De uitkomsten van de testen komen in een testrapport, deze bevindingen kunnen eventueel na het project worden gebruikt om de plug-in te optimaliseren.

Tijdens het afstuderen wordt een eindverslag gemaakt. Dat verslag wordt tijdens iteratie twee gemaakt. Alle gemaakte verslagen staan als bijlage in dat verslag, daarom is het eindverslag ook meteen het afstudeerdossier.

2 Eisen van prototype

In dit hoofdstuk staan de eisen die van toepassing zijn op increment twee. De eisen komen uit het programma van eisen, maar wel gefilterd op increment twee.

| Eisen | | Prio | Tijd | Nr. |
|-------------------------|---|------|------|-----|
| Functioneel | | | | |
| | Informatie | | | |
| | De plug-in laat informatie zien over de huidige geselecteerde view. | M | I2 | 34 |
| | De gebruiker kan de huidige geselecteerde view vastzetten in de plug-in. | M | I2 | 35 |
| | De gebruiker kan informatie inzien over de module die vastgezet is. | M | I2 | 2 |
| | De gebruiker kan de module die de plug-in vast weergeeft losmaken. | M | I2 | 43 |
| | De plug-in geeft kolommen met de eigenschap "hidden" niet weer. | M | I2 | 47 |
| | De plug-in geeft kolommen met de eigenschap "icon" niet weer. | M | I2 | 49 |
| | Filter | | | |
| | Een geselecteerde module heeft een filter. | M | I2 | 32 |
| | De gebruiker kan de filter inschakelen. | M | I2 | 4 |
| | De gebruiker kan de filter uitschakelen. | M | I2 | 5 |
| | De gebruiker kan de filter instellen. | M | I2 | 6 |
| | Sorteren | | | |
| | De gebruiker kan informatie sorteren. | M | I2 | 9 |
| | Zoeken | | | |
| | De gebruiker kan in de plug-in zoeken. | S | I2 | 10 |
| | Vernieuwen | | | |
| | De plug-in heeft een timer. | M | I2 | 33 |
| | De gebruiker kan de plug-in vernieuwen. | M | I2 | 11 |
| | De plug-in wordt per interval vernieuwd. | M | I2 | 12 |
| | De gebruiker kan interval van vernieuwen instellen. | S | I2 | 13 |
| | Gemak | | | |
| | De configuratie van de filter wordt onthouden als Lotus Notes afgesloten wordt. | S | I2 | 29 |
| | De plug-in onthoudt welke module vaststaat als Lotus Notes afgesloten wordt. | M | I2 | 46 |
| | De instelling van de timer wordt onthouden als Lotus Notes afgesloten wordt. | C | I2 | 30 |
| | De instellingen van de weergave worden per module onthouden als Lotus notes afgesloten wordt. | C | I2 | 31 |
| Niet-functioneel | | | | |
| | Performance | | | |

| | | | | | |
|--|----------|--|---|----|----|
| | | De plug-in geeft informatie snel weer. <i>Snel = sneller dan huidige navigatie naar informatie.</i> | M | I2 | 15 |
| | | | | I2 | |
| | | De plug-in is snel geïnstalleerd. <i>Snel = korter dan 3 minuten.</i> | M | I2 | 16 |
| | | | | I2 | |
| | | De plug-in is snel gedeïnstalleerd. <i>Snel = korter dan 3 minuten.</i> | M | I2 | 17 |
| | | | | I2 | |
| | Weergave | | | | |
| | | De gebruiker kan kolommen uit de weergave verbergen. | M | I2 | 51 |
| | | De gebruiker kan kolommen uit de weergave tonen. | M | I2 | 52 |
| | Generiek | | | | |
| | | De plug-in werkt met alle modules in CSP die plug-in enabled zijn. | M | I2 | 26 |
| | | De plug-in werkt in nog te maken modules van CSP die plug-in enabled zijn. | S | I2 | 27 |
| | Gemak | | | | |
| | | De gebruiker kan de te maken plug-in meederere keren in de sidebar van Lotus Notes installeren. | S | I2 | 28 |
| | | De code van de plug-in is voorzien van engels commentaar. | M | I2 | 45 |
| | | Verschillende gebruikers kunnen op dezelfde werkplek gebruik maken van hun persoonlijke plug-in | M | I2 | 50 |

3 Planning

In dit hoofdstuk staan korte werkzaamheden met daarbij de deadline. Het plan van aanpak bevat een bijlage met een planning. De planning in dit hoofdstuk komt, op een paar wijzigingen in de versienummers, overeen met de planning uit het plan van aanpak.

| Taak | Deadline |
|--|-----------------|
| PvE versie 2.0 | 14-apr |
| Feedback PvE opdrachtgever | 15-apr |
| PvA versie 2.1 | 16-apr |
| Plannen tussentijds assessment | 16-apr |
| Functioneel ontwerp versie 0.1 | 23-apr |
| Feedback op afstudeerdossier opdrachtgever, bedrijfsmentor en begeleider | 24-apr |
| Feedback functioneel ontwerp bedrijfsmentor | 27-apr |
| Functioneel ontwerp versie 1.0 | 30-apr |
| Implementeren plug-in 1.0 | 14-mei |
| Afstudeerdossier versie 0.2 | 14-mei |
| Tussentijds assessment | 10 mei – 21 mei |
| Testplan 1.0 | 18-mei |
| Feedback op afstudeerdossier opdrachtgever, bedrijfsmentor en begeleider | 18-mei |
| Testen uitvoeren | 19-mei |
| Uitkomsten testen documenteren | 21-mei |
| Afstudeerdossier versie 1.0 | 21-mei |
| Eventueel feedback op afstudeerdossier opdrachtgever, bedrijfsmentor en begeleider | 25-mei |
| Afstudeerdossier versie 2.0 | 27-mei |

4 Werkzaamheden

Het laatste hoofdstuk gaat over de werkzaamheden. Hierin wordt per fase beschreven wat de werkzaamheden zijn tijdens increment een.

Analyse

De analysefase van increment twee wordt gebruikt om het programma van eisen aan te passen. Door het prototype is het beeld van het eindproduct na increment één beter dan daarvoor. Daarom is het goed om te bepalen of er eisen zijn die aangepast moeten worden of dat er eisen ontbreken.

Ontwerp

Na de analysefase begint de ontwerpfase. Tijdens deze fase wordt het ontwerp uit increment één uitgebreid. Dit wordt gedaan met de modelleertaal UML. Aan het einde van deze fase wordt het functioneel ontwerp opgeleverd. Na het ontwerp wordt begonnen met de realisatie.

Realisatie

Tijdens de realisatie wordt gewerkt in een demonstratieopstelling van CSP. Als de realisatie af is wordt er getest.

Testen

Voordat getest wordt moet eerst een testplan gemaakt worden. De uitkomsten worden gedocumenteerd en eventueel na het project gebruikt om de plug-in aan te passen. De oplevering van increment twee is een eerste versie van de plug-in.

10 Pogramma van eisen versie 2.2

Programma van Eisen

Generieke sidebar plug-in

voor IBM Lotus Notes & Domino

AedifiComm

| | |
|---------------------|--------------------------------|
| Opdrachtgever: | F.J.C. Spruit |
| Bedrijfsmentor: | N.G. Scheppink |
| Begeleider: | D.R. Stikkolorum |
| Expert: | A.G.P. Pronk |
| Afstudeerder: | Chris den Arend |
| Email afstudeerder: | Chris.den.Arend@aedificomm.com |
| Plaats: | Barendrecht |
| Datum: | 27-04-2010 |
| Versie: | 2.2 |



Inhoudsopgave

| | |
|---|----|
| Inleiding..... | 3 |
| 1 Use-case-diagrammen..... | 4 |
| 1.1 Globaal use-case-diagram | 4 |
| 1.2 Gedetailleerde use-case-diagrammen..... | 6 |
| 1.2.1 Plug-in aanpassen | 6 |
| 1.2.2 Informatie inzien..... | 9 |
| 1.2.3 Informatie doorzoeken | 11 |
| 1.2.4 Up to date maken | 14 |
| 2 Eisen | 16 |
| 3 Onthouden van instellingen..... | 19 |
| 4 Aangepaste en toegevoegde eisen | 20 |
| 4.1 Informatie weergeven..... | 20 |
| 4.2 Kolommen tonen en verbergen..... | 20 |
| 4.3 Meerdere installaties van plug-in | 20 |
| Bijlagen..... | 21 |
| 1. Afkomst eisen | 21 |

Inleiding

Aedificomm heeft het softwarepakket CSP ontwikkeld. CSP bestaat uit diverse modules en het is op dit moment niet mogelijk om de meest relevante informatie van die modules altijd zichtbaar te maken voor de eindgebruiker. Elke module bevat informatie en voor elke gebruiker heeft het begrip relevante informatie een andere invulling.

In CSP wordt de relevante informatie niet op elk moment weergegeven. Om informatie te vinden moet eerst genavigeerd worden naar de module waar de gezochte informatie in staat. Vervolgens moet binnen die module een weergave geopend worden waarin de juiste informatie te raadplegen is. Al deze acties kosten veel tijd en klikmomenten en dat is niet gebruikersvriendelijk. Daarom wordt er een plug-in gemaakt voor de sidebar van Lotus Notes. Deze plug-in geeft altijd de gewenste informatie weer.

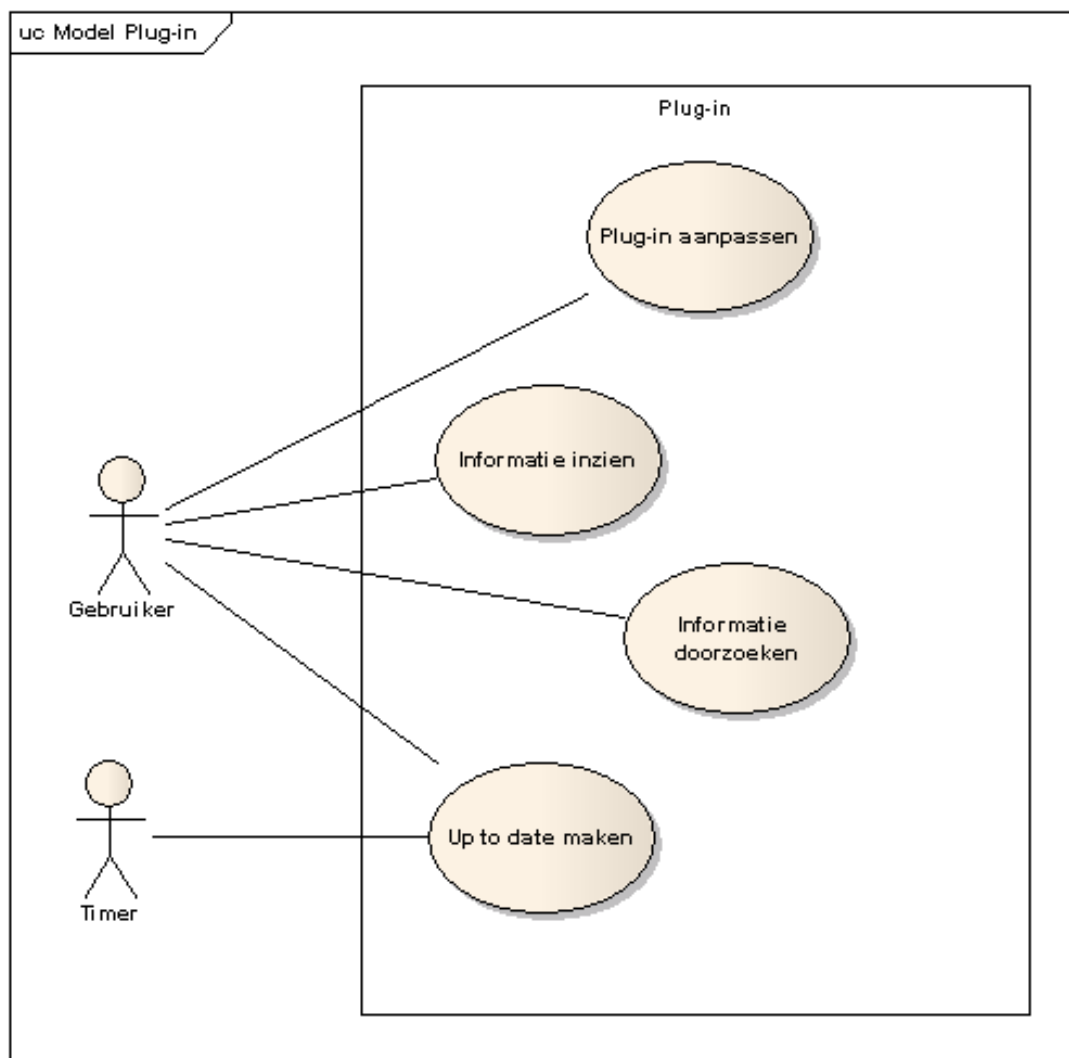
In dit document staat aan welke eisen de te maken plug-in moet voldoen. Als eerste wordt stilgestaan bij de use-case-diagrammen. Er is een globaal diagram en er zijn meerdere specifieke diagrammen. Er is voor gekozen om geen scenario's te maken, omdat die niet van toegevoegde waarde zijn voor dit document. De scenario's die zich aan de kant van de gebruiker afspelen zijn minimaal en niet tijdsgebonden. Daarom verduidelijkt een diagram over een scenario dit document niet. Na de diagrammen komen de eisen aan bod. Als laatste worden de eisen over het onthouden van instellingen uitgelegd.

1 Use-case-diagrammen

In dit hoofdstuk staan de use-case-diagrammen. Als eerste wordt een globaal use-case-diagram weergegeven waar in één oogopslag is te zien wat de gebruiker met het systeem kan. Daarna is er per use-case een gedetailleerd model gemaakt. De use-case-diagrammen zijn bedoeld om samen met de gebruikers en opdrachtgever vast te leggen hoe de gebruiker met de plug-in om moet gaan. Bij elk diagram staan ook beschrijvingen per use-case. Er wordt in dit hoofdstuk niet stilgestaan bij de implementatie van de plug-in.

1.1 Globaal use-case-diagram

In dit use-case-diagram, zie afbeelding 1.1, staan de globale use-cases die betrekking hebben op de gehele plug-in.



Afbeelding 1.1: globaal use-case-diagram van de plug-in

| | |
|---------------------|---|
| Naam | Plug-in aanpassen |
| Samenvatting | De instellingen van de plug-in worden aangepast. |
| Aanname | De plug-in is beschikbaar. |
| Preconditie | Geen |
| Postconditie | De plug-in is aangepast. |
| Omschrijving | De gebruiker gebruikt een functie van de plug-in of de sidebar van Lotus Notes om de plug-in aan te passen. |
| Uitzondering | Geen |

| | |
|---------------------|---|
| Naam | Informatie inzien |
| Samenvatting | De plug-in laat informatie zien. |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes. |
| Preconditie | Geen |
| Postconditie | De plug-in laat informatie zien. |
| Omschrijving | De gebruiker raadpleegt de plug-in. |
| Uitzondering | Geen |

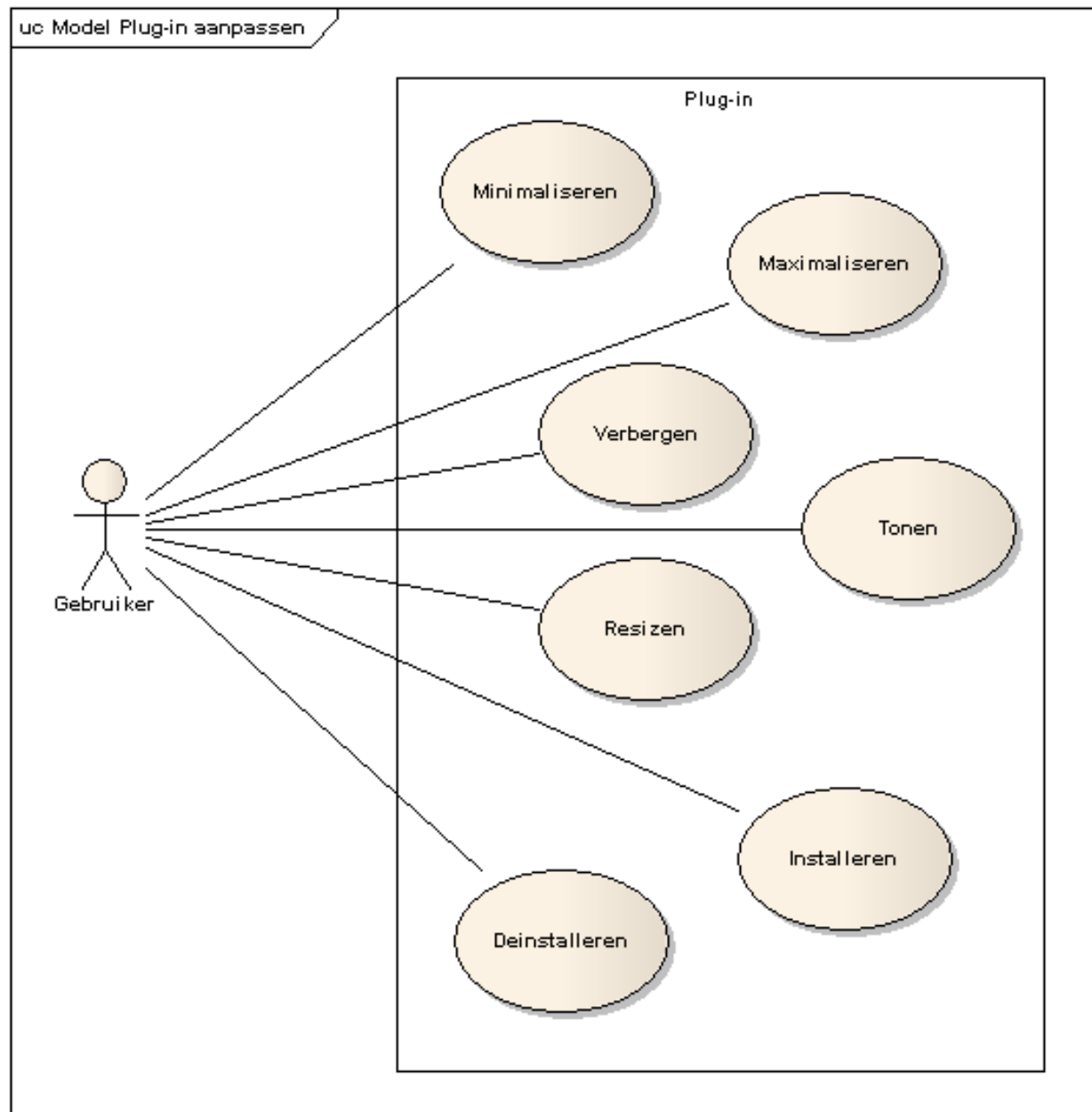
| | |
|---------------------|---|
| Naam | Informatie doorzoeken |
| Samenvatting | In de plug-in wordt naar informatie gezocht. |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes. |
| Preconditie | Er is een module vastgezet in de plug-in. |
| Postconditie | De plug-in laat gezochte informatie zien. |
| Omschrijving | De gebruiker geeft criteria op waaraan de informatie moet voldoen. |
| Uitzondering | Geen |

| | |
|---------------------|---|
| Naam | Up to date maken |
| Samenvatting | De plug-in haalt informatie op die op dat moment in de database staat. |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes. |
| Preconditie | Er is een module geselecteerd. |
| Postconditie | De plug-in laat actuele informatie zien. |
| Omschrijving | De gebruiker geeft aan dat de actuele informatie opgehaald moet worden, of er loopt een timer af. |
| Uitzondering | Geen |

1.2 Gedetailleerde use-case-diagrammen

In dit gedeelte van het document staat per use-case van het globale diagram een use-case-model en de daarbij horende beschrijving.

1.2.1 Plug-in aanpassen



Afbeelding 1.2: use-case-diagram "plug-in aanpassen"

| | |
|---------------------|--|
| Naam | Minimaliseren |
| Samenvatting | De plug-in wordt geminimaliseerd in de sidebar. |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes en er staan naast deze plug-in één of meer andere plug-ins in de sidebar. |
| Preconditie | De plug-in is niet geminimaliseerd. |
| Postconditie | De plug-in staat geminimaliseerd in de sidebar. |
| Omschrijving | De gebruiker klikt met de linkermuisknop op de titel van de plug-in. |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Maximaliseren |
| Samenvatting | De plug-in word gemaximaliseerd in de sidebar. |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes en er staan naast deze plug-in één of meer andere plug-ins in de sidebar. |
| Preconditie | De plug-in is niet gemaximaliseerd. |
| Postconditie | De plug-in staat gemaximaliseerd in de sidebar. |
| Omschrijving | De gebruiker klikt met de linkermuisknop op de titel van de plug-in. |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Verbergen |
| Samenvatting | De plug-in verdwijnt uit de sidebar van Lotus Notes, maar is nog wel geïnstalleerd. |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes. |
| Preconditie | Geen |
| Postconditie | De plug-in is niet zichtbaar in de sidebar van Lotus Notes. |
| Omschrijving | De gebruiker klikt met de rechtermuisknop op de titel van de plugin en kiest in het menu dat openklapt: Hide panel: "plug-in naam". Of de gebruiker navigeert in het menu naar view->right sidebar panels en vinkt daar de plug-in uit. |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Tonen |
| Samenvatting | De plug-in wordt zichtbaar in de sidebar van Lotus Notes. |
| Aanname | De plug-in is geïnstalleerd in de sidebar van Lotus Notes en is verborgen. |
| Preconditie | Geen |
| Postconditie | De plug-in is zichtbaar in de sidebar van Lotus Notes. |
| Omschrijving | De gebruiker navigeert in het menu naar view->right sidebar |

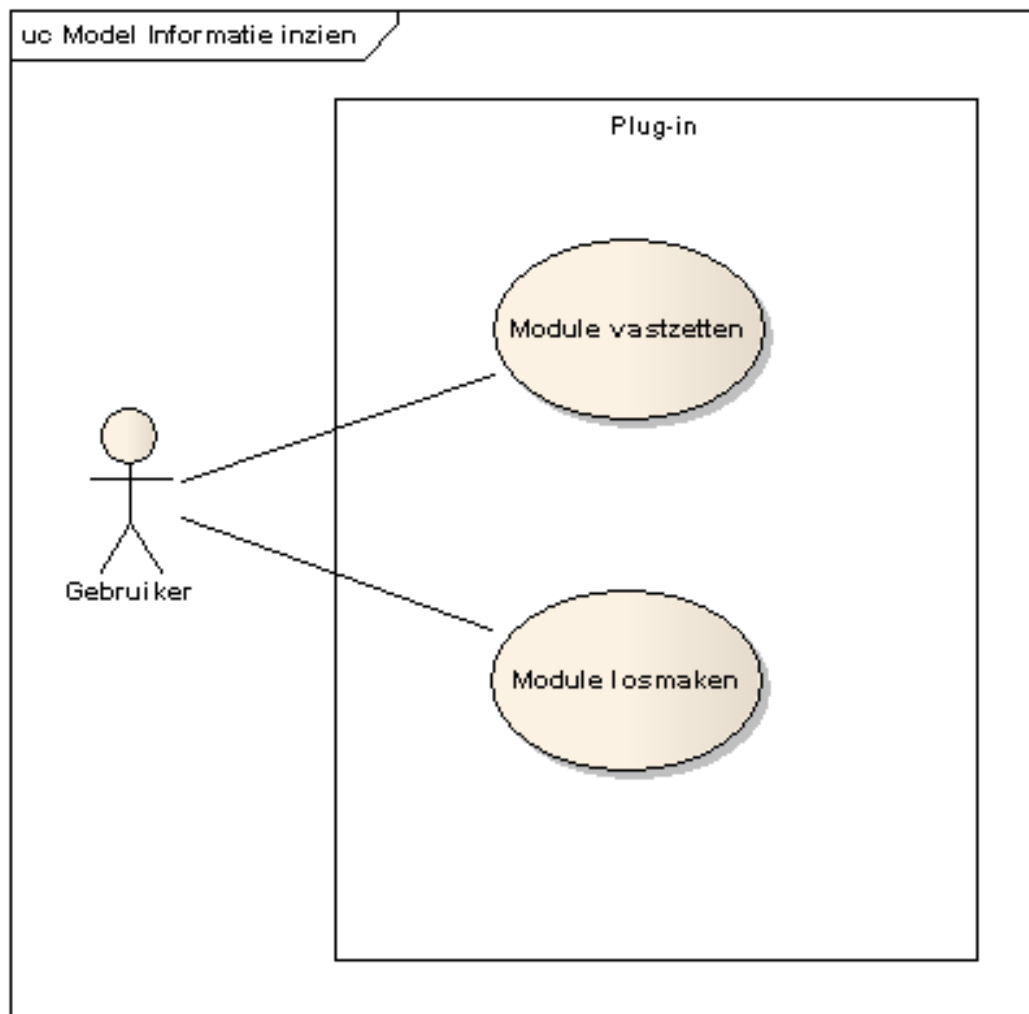
| | |
|---------------------|--------------------------------------|
| | panels en vinkt daar de plug-in aan. |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Resizen |
| Samenvatting | De afmetingen van de plug-in worden aangepast. |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes. |
| Preconditie | Er staat naast deze plug-in één of meer andere plug-ins in de sidebar. |
| Postconditie | De afmetingen van de plug-in zijn aangepast. |
| Omschrijving | De gebruiker sleept de onderkant van de plug-in omhoog of omlaag. Of de gebruiker sleept de linker zijkant van de sidebar naar links of naar rechts. |
| Uitzondering | Geen |

| | |
|---------------------|---|
| Naam | Installeren |
| Samenvatting | De plug-in wordt in de sidebar van Lotus Notes geïnstalleerd. |
| Aanname | De plug-in is beschikbaar. |
| Preconditie | Geen |
| Postconditie | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes. |
| Omschrijving | De gebruiker sleept het icoon van de plug-in in de sidebar van Lotus Notes. Of de gebruiker navigeert in het menu naar file->application->install, dan opent een nieuw venster daar kiest de gebruiker voor "search for new features to install", waarna de plug-in toegevoegd kan worden. |
| Uitzondering | Geen |

| | |
|---------------------|---|
| Naam | Deïnstalleren |
| Samenvatting | De plug-in wordt verwijderd uit Lotus Notes. |
| Aanname | De plug-in is geïnstalleerd. |
| Preconditie | Geen |
| Postconditie | De verwijderde plug-in is niet meer aanwezig in Lotus Notes. |
| Omschrijving | De gebruiker gaat naar de plug-in "My Widgets", daar doet de gebruiker rechtermuisknop op de te verwijderen plug-in en klikt dan op remove. |
| Uitzondering | Geen |

1.2.2 Informatie inzien

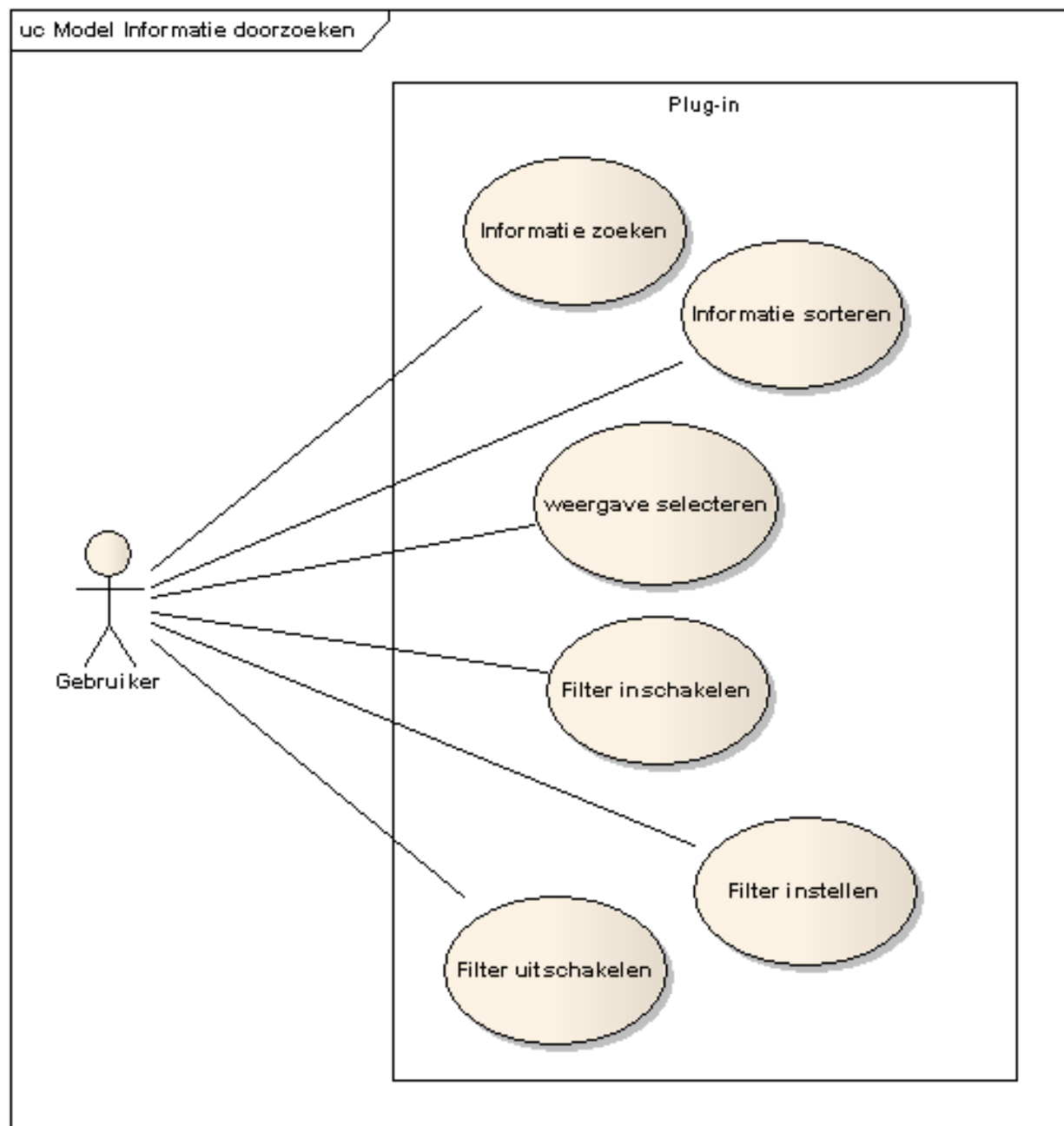


Afbeelding 1.4: use-case-diagram "informatie inzien"

| | |
|---------------------|--|
| Naam | Module vastzetten |
| Samenvatting | Er wordt een module in de plug-in vastgezet. |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes. |
| Preconditie | De plug-in bevat nog geen module die is vastgezet. |
| Postconditie | Informatie over de geselecteerde module wordt weergegeven in de plug-in. |
| Omschrijving | De gebruiker kan een module openen, de plug-in laat informatie zien van de huidig geselecteerde module. In de plug-in is een knop om de huidig geselecteerde module vast te zetten in de plug-in, de gebruiker klikt daar op. |
| Uitzondering | Geen |

| | |
|---------------------|---|
| Naam | Module losmaken |
| Samenvatting | Een module wordt uit de plug-in verwijderd. |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes. |
| Preconditie | De plug-in bevat een module die is vastgezet. |
| Postconditie | De module is niet weergegeven in de plug-in. |
| Omschrijving | De gebruiker kan op de knop in de module klikken die ervoor zorgt dat de module die vaststaat in de plug-in losgemaakt wordt. |
| Uitzondering | Geen |

1.2.3 Informatie doorzoeken



Afbeelding 1.6: use-case-diagram "informatie doorzoeken"

| | |
|---------------------|---|
| Naam | Informatie zoeken |
| Samenvatting | Er wordt gezocht in de plug-in. |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes. |
| Preconditie | De plug-in bevat een module die is vastgezet. |
| Postconditie | De resultaten van de zoekactie zijn zichtbaar in de plug-in. |
| Omschrijving | Onder de titel van de plug-in staat een zoekbalk. Hierin kan de |

| | |
|---------------------|--|
| | gebruiker een zoekwoord typen en op de zoekknop klikken. |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Weergave selecteren |
| Samenvatting | De weergave van de module wordt geselecteerd. |
| Aanname | De plug-in is geïnstalleerd, zichtbaar in de sidebar van Lotus Notes en de plug-in bevat een vastgezette module die meerdere views ondersteund. |
| Preconditie | Geen |
| Postconditie | De plug-in heeft een door de gebruiker geselecteerde weergave. |
| Omschrijving | De gebruiker klikt met de rechtermuisknop op de titel van de plug-in. Daar kan geklikt worden op weergave. Dan ziet de gebruiker een drop-down menu. Daar kan de gebruiker een weergave selecteren, zodat de plug-in op die manier weergegeven wordt. |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Informatie sorteren |
| Samenvatting | De informatie in de plug-in is gesorteerd. |
| Aanname | De plug-in is geïnstalleerd, zichtbaar in de sidebar van Lotus Notes en de plug-in bevat een module. |
| Preconditie | Geen |
| Postconditie | De resultaten van de sorteeractie zijn zichtbaar in de plug-in. |
| Omschrijving | De gebruiker kan op de titel van een kolom klikken, dan wordt de informatie gesorteerd. |
| Uitzondering | Geen |

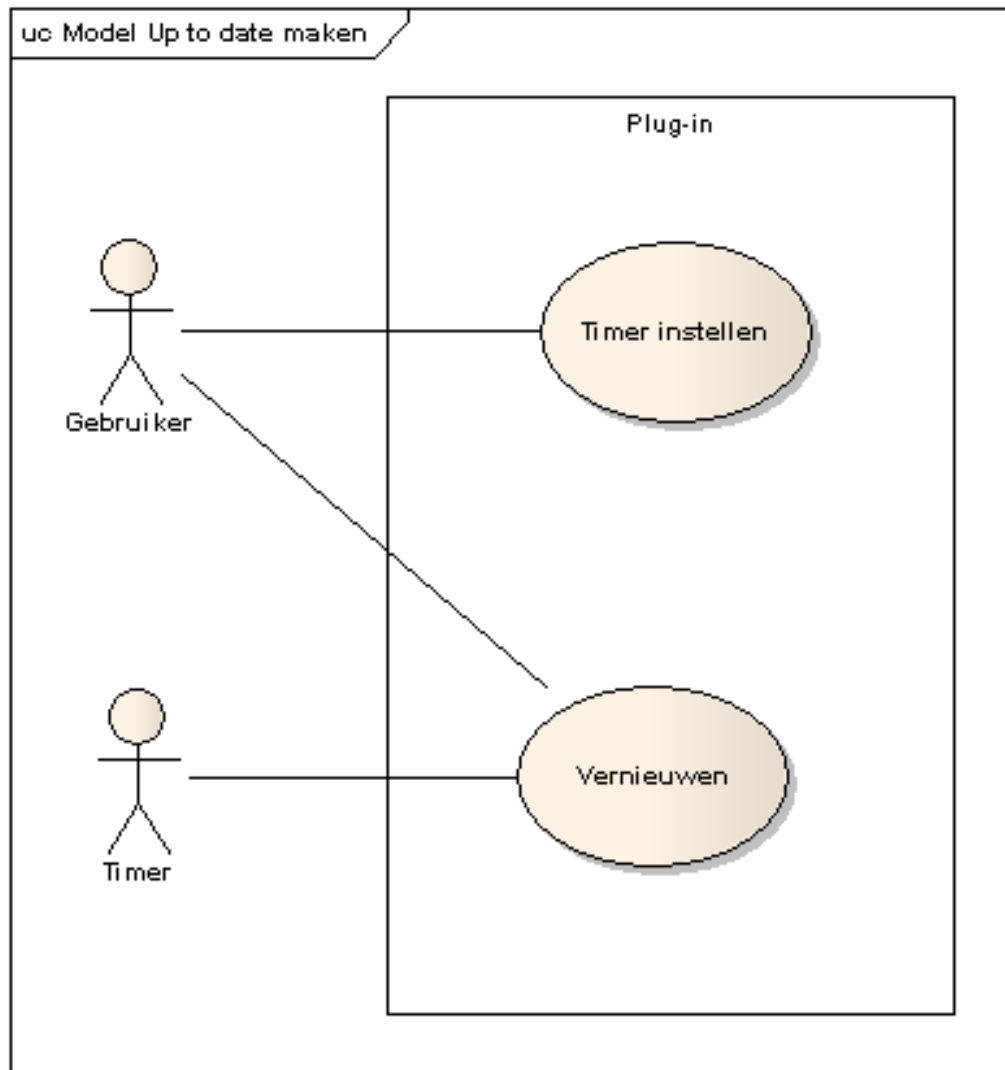
| | |
|---------------------|---|
| Naam | Filter inschakelen |
| Samenvatting | De filter van de plug-in wordt ingeschakeld. |
| Aanname | De plug-in is geïnstalleerd, zichtbaar in de sidebar van Lotus Notes en de plug-in bevat een vastgezette module. |
| Preconditie | Geen |
| Postconditie | De plug-in laat alle informatie zien die door de filter gelaten wordt. |
| Omschrijving | De gebruiker klikt met de rechtermuisknop op de titel van de plug-in. Daar is dan een optie om de filter in te schakelen. |
| Uitzondering | Geen |

| | |
|---------------------|--|
| Naam | Filter instellen |
| Samenvatting | De filter van de plug-in wordt ingesteld. |
| Aanname | De plug-in is geïnstalleerd, zichtbaar in de sidebar van Lotus Notes en de plug-in bevat een module. |
| Preconditie | Geen |
| Postconditie | De filter van de plug-in is ingesteld met voorwaarden die de gebruiker heeft ingevoerd. |
| Omschrijving | Onder de titel van de plug-in staat een filter. Hierin kan de |

| | |
|---------------------|--|
| | gebruiker een woord typen waarop gefilterd moet worden en daarna op de filterknop klikken. |
| Uitzondering | Geen |

| | |
|---------------------|---|
| Naam | Filter uitschakelen |
| Samenvatting | De filter van de plug-in wordt uitgeschakeld. |
| Aanname | De plug-in is geïnstalleerd, zichtbaar in de sidebar van Lotus Notes en de plug-in bevat een module. |
| Preconditie | De filter van de plug-in is ingeschakeld. |
| Postconditie | De filter van de plug-in is uitgeschakeld. |
| Omschrijving | De gebruiker klikt met de rechtermuisknop op de titel van de plug-in. Daar kan geklikt worden op filter uitschakelen. |
| Uitzondering | Geen |

1.2.4 Up to date maken



Afbeelding 1.8: use-case-diagram "up to date maken"

| Naam | Timer instellen |
|---------------------|---|
| Samenvatting | De timer die regelt wanneer de plug-in vernieuwd wordt is ingesteld. |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes. |
| Preconditie | Geen |
| Postconditie | De timer die regelt wanneer de plug-in vernieuwd wordt is ingesteld. |
| Omschrijving | De gebruiker klikt met de rechtermuisknop op de titel van de plug-in. Daar staat dan een optie om timer van de plug-in in te stellen. |
| Uitzondering | Geen |

| | |
|---------------------|---|
| Naam | Vernieuwen |
| Samenvatting | De informatie in de plug-in wordt vernieuwd |
| Aanname | De plug-in is geïnstalleerd en zichtbaar in de sidebar van Lotus Notes |
| Preconditie | De plug-in bevat een module |
| Postconditie | De plug-in geeft de informatie weer die op dat moment in de database staat |
| Omschrijving | De gebruiker kan op een knop klikken in de header van de plug-in. De plug-in wordt dan vernieuwd De timer loopt periodiek af. Als de timer afloopt wordt de plug-in vernieuwd. |
| Uitzondering | Geen |

2 Eisen

In dit hoofdstuk worden de eisen benoemd. Dat gebeurt volgens de MoSCoW methode. Met de termen 'must have', 'should have', 'could have' of 'won't have' is aan te geven of er per moment nog een bepaalde speelruimte is of juist niet. De termen hebben de volgende betekenis:

- Must have (M): moet op dat tijdstip gerealiseerd zijn
- Should have (S): zou op dat tijdstip gerealiseerd moeten zijn maar een tijdstip later is acceptabel
- Could have (C): mag (als tijd en geld over is) op het tijdstip gerealiseerd zijn maar moet niet.
- Won't have (W): zal bij het bereiken van het tijdstip niet gerealiseerd zijn.

In dit project wordt gebruik gemaakt van deze methode. De prioriteiten staan achter elke eis. Ook staat er op welk tijdstip de eis van toepassing is, increment één (I1) of increment twee (I2).

In bijlage 1 staat een tabel met de nummers van de eisen en daarbij bij wie ze vandaan komen en wanneer ze vastgelegd zijn.

| Eisen | | | Prio | Tijd | Nr. |
|--------------------|-----------------|--|------|------|-----|
| Functioneel | | | | | |
| | Informatie | | | | |
| | | De plug-in laat informatie zien over de huidige geselecteerde view. | M | I2 | 34 |
| | | De gebruiker kan de huidige geselecteerde view vastzetten in de plug-in. | M | I2 | 35 |
| | | De gebruiker kan informatie inzien over de module die vastgezet is. | M | I2 | 2 |
| | | De gebruiker kan de module die de plug-in vast weergeeft losmaken. | M | I2 | 43 |
| | | De plug-in geeft kolommen met de eigenschap "hidden" niet weer. | M | I2 | 47 |
| | | De plug-in geeft kolommen met de eigenschap "icon" niet weer. | M | I2 | 49 |
| | Aanpassen | | | | |
| | | De gebruiker kan de plug-in verbergen. | S | I1 | 3 |
| | | De gebruiker kan de plug-in tonen. | M | I1 | 41 |
| | | De gebruiker kan de plug-in minimaliseren. | M | I1 | 39 |
| | | De gebruiker kan de plug-in maximaliseren. | M | I1 | 40 |
| | | De gebruiker kan de plug-in resizen. | M | I1 | 42 |
| | Filter | | | | |
| | | Een geselecteerde module heeft een filter. | M | I2 | 32 |
| | | De gebruiker kan de filter inschakelen. | M | I2 | 4 |
| | | De gebruiker kan de filter uitschakelen. | M | I2 | 5 |
| | | De gebruiker kan de filter instellen. | M | I2 | 6 |
| | (de)installatie | | | | |
| | | De gebruiker kan de plug-in installeren. | M | I1 | 7 |

| | | | | | |
|--|------------|---|---|----|----|
| | | De gebruiker kan de plug-in deïnstalleren. | M | I1 | 8 |
| | Sorteren | | | | |
| | | De gebruiker kan informatie sorteren. | M | I2 | 9 |
| | Zoeken | | | | |
| | | De gebruiker kan in de plug-in zoeken. | S | I2 | 10 |
| | Vernieuwen | | | | |
| | | De plug-in heeft een timer. | M | I2 | 33 |
| | | De gebruiker kan de plug-in vernieuwen. | M | I2 | 11 |
| | | De plug-in wordt per interval vernieuwd. | M | I2 | 12 |
| | | De gebruiker kan interval van vernieuwen instellen. | S | I2 | 13 |
| | Gemak | | | | |
| | | De plug-in is in de Nederlandse taal. | M | I1 | 14 |
| | | De configuratie van de filter wordt onthouden als Lotus Notes afgesloten wordt. | S | I2 | 29 |
| | | De plug-in onthoudt welke module vaststaat als Lotus Notes afgesloten wordt. | M | I2 | 46 |
| | | De instelling van de timer wordt onthouden als Lotus Notes afgesloten wordt. | C | I2 | 30 |
| | | De instellingen van de weergave worden per module onthouden als Lotus notes afgesloten wordt. | C | I2 | 31 |

Niet-functioneel

| | | | | | |
|--|-------------|---|---|----|----|
| | Performance | | | | |
| | | De plug-in geeft informatie snel weer. | M | I2 | 15 |
| | | <i>Snel = sneller dan huidige navigatie naar informatie.</i> | | I2 | |
| | | De plug-in is snel geïnstalleerd. | M | I2 | 16 |
| | | <i>Snel = korter dan 3 minuten.</i> | | I2 | |
| | | De plug-in is snel gedeïnstalleerd. | M | I2 | 17 |
| | | <i>Snel = korter dan 3 minuten.</i> | | I2 | |
| | Weergave | | | | |
| | | De gebruiker kan kolommen uit de weergave verbergen. | M | I2 | 51 |
| | | De gebruiker kan kolommen uit de weergave tonen. | M | I2 | 52 |
| | Uiterlijk | | | | |
| | | De user-interface van de plug-in moet aansluiten bij CSP. | M | I1 | 20 |
| | | De user-interface van de plug-in moet aansluiten bij Lotus Notes. | M | I1 | 21 |
| | | De plug-in past in de sidebar van Lotus Notes. | M | I1 | 22 |
| | | Na installatie is de plug-in de bovenste plug-in in de sidebar. | M | I1 | 44 |
| | Platform | | | | |
| | | De plug-in werk op het platform IBM Lotus | M | I1 | 23 |

| | | | | | |
|--|----------|---|---|----|----|
| | | Notes en Domino 8.5. | | | |
| | | De plug-in werkt op versie 8 van CSP. | M | I1 | 24 |
| | | De plug-in werkt op Windows XP. | M | I1 | 36 |
| | | De plug-in werkt op Windows Vista. | M | I1 | 37 |
| | | De plug-in werkt op Windows 7. | M | I1 | 38 |
| | Generiek | | | | |
| | | De plug-in werkt met alle modules in CSP die plug-in enabled zijn. | M | I2 | 26 |
| | | De plug-in werkt in nog te maken modules van CSP die plug-in enabled zijn. | S | I2 | 27 |
| | Gemak | | | | |
| | | De gebruiker kan de te maken plug-in meederere keren in de sidebar van Lotus Notes installeren. | S | I2 | 28 |
| | | De code van de plug-in is voorzien van engels commentaar. | M | I2 | 45 |
| | | Verschillende gebruikers kunnen op dezelfde werkplek gebruik maken van hun persoonlijke plug-in | M | I2 | 50 |

3 Onthouden van instellingen

Voor de gebruikersvriendelijkheid is het goed dat tijdens het werken met de plug-in verschillende instellingen onthouden worden en verschillende instellingen juist niet. De gebruiker kan een module vastzetten in de plug-in. Als dat gebeurt is kan de informatie in de plug-in gefilterd worden. Op die manier kan de gebruiker relevante informatie weer laten geven in de plug-in. Het vastzetten en de instellingen van de filter worden onthouden, zolang de filter vaststaat in de plug-in. Als een module losgemaakt wordt van de plug-in verdwijnen ook alle instellingen in de module. Het resizen, minimaliseren en maximaliseren wordt niet onthouden. De installatie, deinstallatie en het verbergen en tonen wordt, vanzelfsprekend, wel onthouden.

4 Aangepaste en toegevoegde eisen

In dit hoofdstuk wordt besproken wat de veranderingen en toevoegingen aan de eisen zijn in increment twee.

4.1 Informatie weergeven

In increment één is vastgelegd dat de gebruiker een gegevensbron uit CSP in de plug-in kan slepen waardoor de informatie uit die bron weergegeven wordt in de plug-in. Het prototype van increment één laat bij elke selectie van een gegevensbron de informatie daaruit zien in de plug-in. Dat bracht de opdrachtgever op het idee om dat zo te houden. Daarnaast moet de gebruiker de module die op dat moment geselecteerd is vast kunnen zetten in de plug-in.

Op deze manier ziet de gebruiker hoe de informatie uit de geselecteerde gegevensbron in de plug-in weergegeven wordt. Als de gebruiker eerst een module moet slepen in de plug-in kost dat meer tijd. Daarom is die eise veranderd.

4.2 Kolommen tonen en verbergen

Een eis in increment één was dat de gebruiker de weergave van de plug-in kan veranderen. Die eis maakt niet in een keer duidelijk wat er precies kan worden veranderd. Daarom is die eis in increment twee verduidelijkt naar "De gebruiker kan kolommen uit de weergave tonen en verbergen". Naast de notatie van de eis is ook de prioriteit veranderd. Door het prototype werd duidelijk dat veel gegevensbronnen meer kolommen hebben dan goed is weer te geven in de plug-in. Een van de eisen is dat de plug-in moet passen in de sidebar, dat is niet altijd het geval. Daarom moet de gebruiker kolommen kunnen tonen en verbergen.

4.3 Meerdere installaties van plug-in

Tijdens increment één bleek dat het lastig is om een plug-in te ontwikkelen die meerdere malen te installeren is in dezelfde sidebar, terwijl dat wel een eis is. De ontwikkelaar en de opdrachtgever hebben toen besloten de prioriteit van deze eis te veranderen van M naar S. Op die manier wordt duidelijk dat eerst andere functies van de plug-in ontwikkeld moeten worden en als er genoeg tijd is wordt de eis van het meerdere malen installeren van de plug-in behandeld.

Bijlagen

1. Afkomst eisen

| Afkomst | Nr. |
|--|-----|
| NIS, schriftelijk bij de opdrachtomschrijving | 2 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 3 |
| NIS, telefonisch bij de opdrachtomschrijving | 4 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 5 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 6 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 7 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 8 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 9 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 10 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 11 |
| NIS, mondeling bij bespreken van plan van aanpak | 12 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 13 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 14 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 15 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 16 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 17 |
| NIS, mondeling bij bespreken van plan van aanpak | 20 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 21 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 22 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 23 |
| NIS, schriftelijk bij de opdrachtomschrijving | 24 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 26 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 27 |
| NIS, mondeling bij interview over eisen (02-03-2010) | 28 |
| NIS, mondeling bij interview over eisen (02-03-2010) | 29 |
| NIS, mondeling bij interview over eisen (02-03-2010) | 30 |
| NIS, mondeling bij interview over eisen (02-03-2010) | 31 |
| NIS, mondeling bij interview over eisen (08-03-2010) | 32 |
| NIS, mondeling bij interview over eisen (08-03-2010) | 33 |
| NIS, mondeling (12-04-2010) | 34 |
| NIS, mondeling (12-04-2010) | 35 |
| NIS, mondeling (12-04-2010) | 36 |
| NIS, mondeling (12-04-2010) | 37 |
| NIS, mondeling (12-04-2010) | 38 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 39 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 40 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 41 |
| NIS, mondeling bij interview over eisen (18-02-2010) | 42 |
| NIS, mondeling (12-04-2010) | 43 |
| NIS, mondeling bij interview over eisen (15-04-2010) | 44 |

| | |
|--|----|
| NIS, mondeling bij interview over eisen (15-04-2010) | 45 |
| NIS, mondeling bij interview over eisen (15-04-2010) | 46 |
| NIS, mondeling bij interview over eisen (19-04-2010) | 47 |
| NIS, mondeling bij interview over eisen (19-04-2010) | 49 |
| NIS, mondeling bij interview over eisen (19-04-2010) | 50 |
| NIS, mondeling bij interview over eisen (23-04-2010) | 51 |
| NIS, mondeling bij interview over eisen (23-04-2010) | 52 |

11 Ontwerp increment 2

Ontwerp increment 2

Generieke sidebar plug-in
voor IBM Lotus Notes & Domino

AedifiComm

| | |
|---------------------|---------------------------------|
| Opdrachtgever: | F.J.C. Spruit |
| Bedrijfsmentor: | N.G. Scheppink |
| Begeleider: | D.R. Stikkolorum |
| Expert: | A.G.P. Pronk |
| Afstudeerder: | Chris den Arend |
| Email afstudeerder: | Chris.den.Arend@aedificommm.com |
| Plaats: | Barendrecht |
| Datum: | 10-05-2010 |
| Versie: | 0.1 |



Inhoudsopgave

| | |
|-------------------------------------|----|
| Inleiding..... | 3 |
| 1 Packagediagram | 4 |
| 2 Klassediagram..... | 6 |
| 2.1 Plugin..... | 6 |
| 2.1.1 Weergave informatie | 6 |
| 2.1.2 Acties..... | 7 |
| 2.1.3 Instellingen onthouden..... | 7 |
| 2.2 Context Service | 7 |
| 2.3 Lotus Notes | 8 |
| 2.4 Eclipse | 8 |
| 3 Sequentiediagram | 9 |
| 3.1 Globaal diagram | 9 |
| 3.2 Gedetailleerde diagrammen | 10 |
| 3.2.1 Filter..... | 10 |
| 3.2.2 Vastzetten van pagina | 11 |
| 3.2.3 Drop down menu | 12 |
| 4 Bronnen | 13 |
| 4.1 Boeken | 13 |
| 4.2 Websites..... | 13 |
| Bijlagen..... | 14 |
| 1 Diagrammen | 14 |

Inleiding

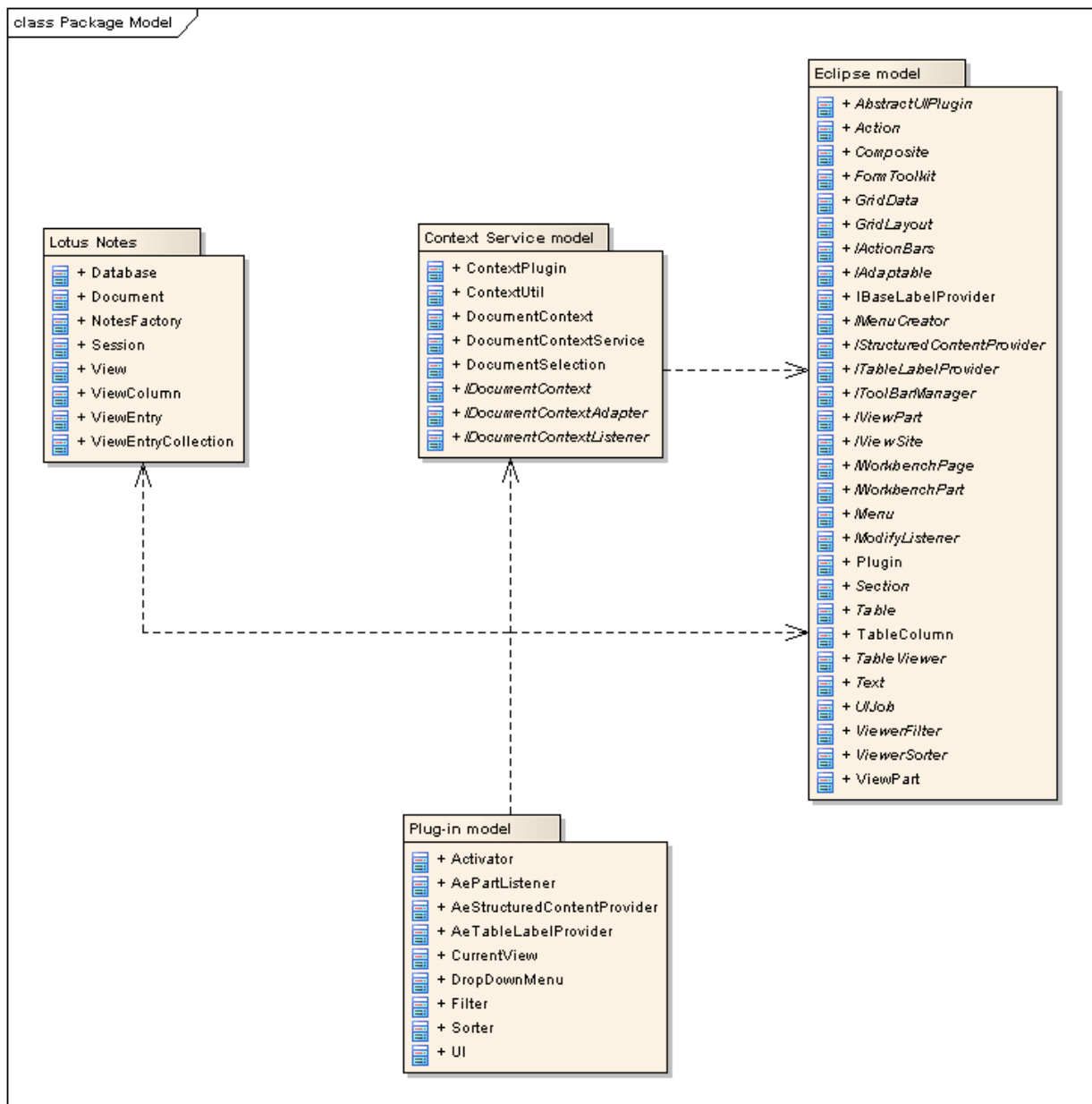
Dit document sluit aan op het ontwerp van increment één. In het ontwerp van increment één is vooral gekeken naar de platformen waarop de plug-in gaat werken en hoe de plug-in van die platformen gebruik kan maken. Het ontwerp van increment twee, dat in dit document beschreven staat, gaat dieper in op de plug-in. Tijdens het ontwerp zijn klassediagrammen en sequentiediagrammen gemaakt. Deze komen in dit verslag aan bod.

In bijlage 1 staat een .EAP bestand met alle diagrammen uit het ontwerp. Het bestand is te openen met Enterprise Architect. Sommige diagrammen zijn te groot om leesbaar op papier te zetten, daarom deze bijlage. De diagrammen bevatten klassen van de plug-in en klassen uit andere platformen waar de plug-in een relatie mee heeft.

Tijdens het ontwerpen is gebruik gemaakt van [Warner06].

1 Packagediagram

In dit hoofdstuk wordt het packagediagram besproken. Een packagediagram geeft op een abstractere manier dan een klassediagram weer hoe een systeem in elkaar zit. Tijdens het ontwerp breidde het klassediagram uit en werd te groot om overzichtelijk in één diagram te laten zien. In afbeelding 1.1 staat het packagediagram voor de plug-in.



Afbeelding 1.1: packagediagram

In het ontwerprapport van increment één staat uiteengelegd welke lager te vinden zijn in het packagediagram. Het ontwerp van increment één bevatte geen package Lotus Notes, dit diagram wel. Deze package valt onder de semantische laag en is minder

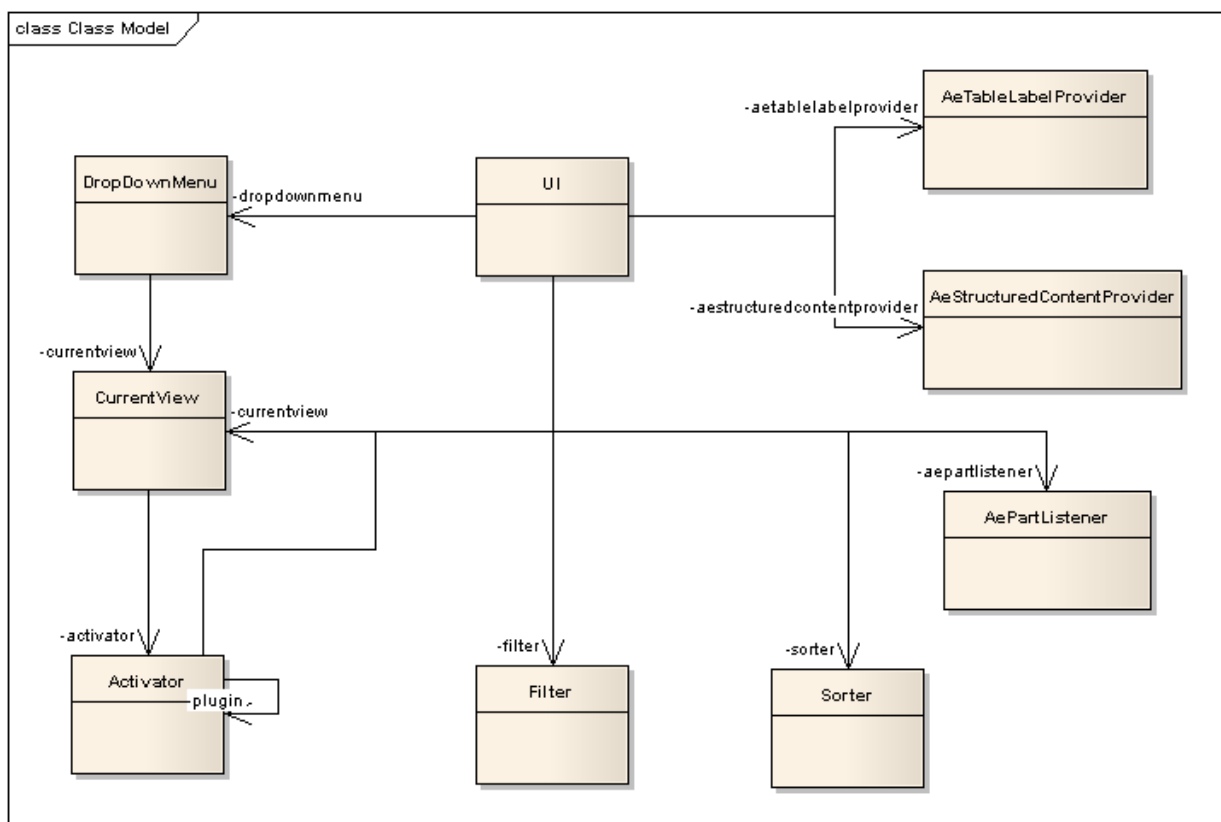
abstract dan de package Eclipse. Dat kan gezegd worden, omdat Lotus Notes gebaseerd is op Eclipse. De package Plug-in wordt geïmplementeerd, van de andere packages wordt gebruik gemaakt door de Plug-in. Voor de duidelijkheid zijn deze in dit model opgenomen.

2 Klassediagram

In dit hoofdstuk staan de klassediagrammen. Een klassediagram laat zien hoe een systeem is opgebouwd. Het diagram wordt gemaakt voordat geïmplementeerd wordt, zo denkt de ontwikkelaar na over het systeem voordat het systeem werkelijk gebouwd wordt.

2.1 Plugin

In afbeelding 2.1 is het klassediagram van de plug-in te zien zonder attributen en operaties. Het volledige diagram is te vinden in bijlage 1.



Afbeelding 2.1: klassediagram van plug-in

2.1.1 Weergave informatie

In het diagram is te zien dat de klasse **UI** het middelpunt is van de plug-in. Deze klasse wordt aangemaakt bij het starten van de plug-in. Daardoor wordt de user interface van de plug-in opgebouwd. Voor die opbouw zijn veel klassen uit de package Eclipse nodig, daar wordt op verder gegaan in paragraaf 2.4.

De klasse **UI** heeft ook een listener uit de package Context Service. Als de plug-in is opgebouwd zorgt die listener er voor dat een functie wordt aangeroepen op het moment dat de gebruiker een andere pagina selecteerd in CSP. De plug-in haalt dan de view, die in de package Lotus Notes staat, op en geeft deze weer in de plug-in. Op deze manier laat de plug-in informatie zien over de huidige geselecteerde view.

2.1.2 Acties

Tijdens het opbouwen van de plug-in wordt ook een toolbar gemaakt met acties daarin. De acties in de plug-in zijn:

- Huidige geselecteerde view vastzetten
- Informatie filteren (klasse Filter)
- Kolommen tonen of verbergen (klasse DropDownMenu)

De gebruiker kan op een van deze acties klikken, waardoor deze wordt uitgevoerd. Op die manier kan de huidige geselecteerde view worden vastgezet in de plug-in. Als de gebruiker dan een andere pagina selecteerd blijft de view die is vastgezet zichtbaar. Als een pagina is vastgezet kan de gebruiker de overige acties gebruiken, zolang een weergave niet is vastgezet zijn die acties niet beschikbaar. Een andere actie is de filter, deze kan de gebruiker in en uitschakelen en een waarde opgeven waarop gefiltert moet worden. De laatste actie in de toolbar is een drop down menu met daarin acties. Als het menu wordt opengeklapt staan daar de kolommen van de vastgezette view. De gebruiker kan in het menu op een kolom klikken waardoor deze wordt getoond of verborgen.

Het vastpinnen van de huidige geselecteerde view is geen aparte klasse, omdat bij die actie alleen een instelling van de plug-in wordt veranderd.

2.1.3 Instellingen onthouden

In het programma van eisen is te lezen dat de plug-in verschillende instellingen moet kunnen onthouden. Op die manier hoeft de gebruiker de plug-in niet elke keer opnieuw in te stellen als CSP wordt opgestart. De plug-in moet verschillende instellingen kunnen onthouden:

- De vastgezette view
- Zichtbare en onzichtbare kolommen
- Waarde van de filter

De plug-in bevat functies die het mogelijk maken om de instellingen te onthouden, in paragraaf 2.4 wordt daar verder op ingegaan.

Een andere eis is dat verschillende gebruikers op dezelfde computer gebruik kunnen maken van hun persoonlijke plug-in. Dat is te realiseren door de plug-in en zijn instellingen op te slaan in de persoonlijke map van Lotus Notes, waar CSP op is gebaseerd. Bij het installeren van de plug-in wordt deze default in de persoonlijke map gezet, dus daar hoeft in het ontwerp geen rekening mee worden gehouden.

2.2 Context Service

Deze package door IBM beschikbaar is gesteld. Hierin wordt bijgehouden welk document geselecteerd is in Lotus Notes. Daarnaast worden gegevens over het geselecteerde document opgeslagen in de klasse DocumentContext. In die klasse

staat de URI van het geselecteerde document. Die URI wordt in de plug-in gebruikt om de view te achterhalen die de gebruiker op dat moment open heeft staan.

2.3 Lotus Notes

De plug-in gebruikt verschillende klassen uit Lotus Notes. In paragraaf 2.2 is te lezen dat Context Service zorgt dat het huidige geselecteerde document en de URI daarvan beschikbaar zijn. De plug-in haalt dan de parent database van het document op. Uit de database kan een view gehaald worden. Daarvoor moeten bekend zijn welke view op dat moment op het scherm van de gebruiker weergegeven wordt. Dat is te achterhalen door het id van de view uit de URI te halen. Nu bekend is welke view weergegeven moet worden in de plug-in wordt elke regel uit deze view opgehaald en weergegeven met behulp van de klassen ViewEntryCollection en ViewEntry.

2.4 Eclipse

Zoals in paragraaf 2.1 staat worden veel klassen uit Eclipse gebruikt om de plug-in op te bouwen, namelijk:

- IViewPart
- IWorkbenchPart
- Viewpart
- FormToolkit
- Section
- Composite
- GridData
- Table
- TableViewer
- TableColumn

De UI maakt objecten van deze klassen aan waardoor de plug-in wordt opgebouwd in de sidebar van Lotus Notes.

De toolbar wordt gemaakt met de klasse IActionBars en de acties met de klasse Action.

Verder zijn twee klassen belangrijk voor het vullen van de plug-in, namelijk IStructuredContentProvider en ITableLabelProvider. De contentprovider zet alle informatie per cel klaar voor de labelprovider, die zorgt er voor dat die informatie in de goede cel van de tabel wordt gezet. Meer informatie over klassen in eclipse is te vinden op [Eclipse2005].

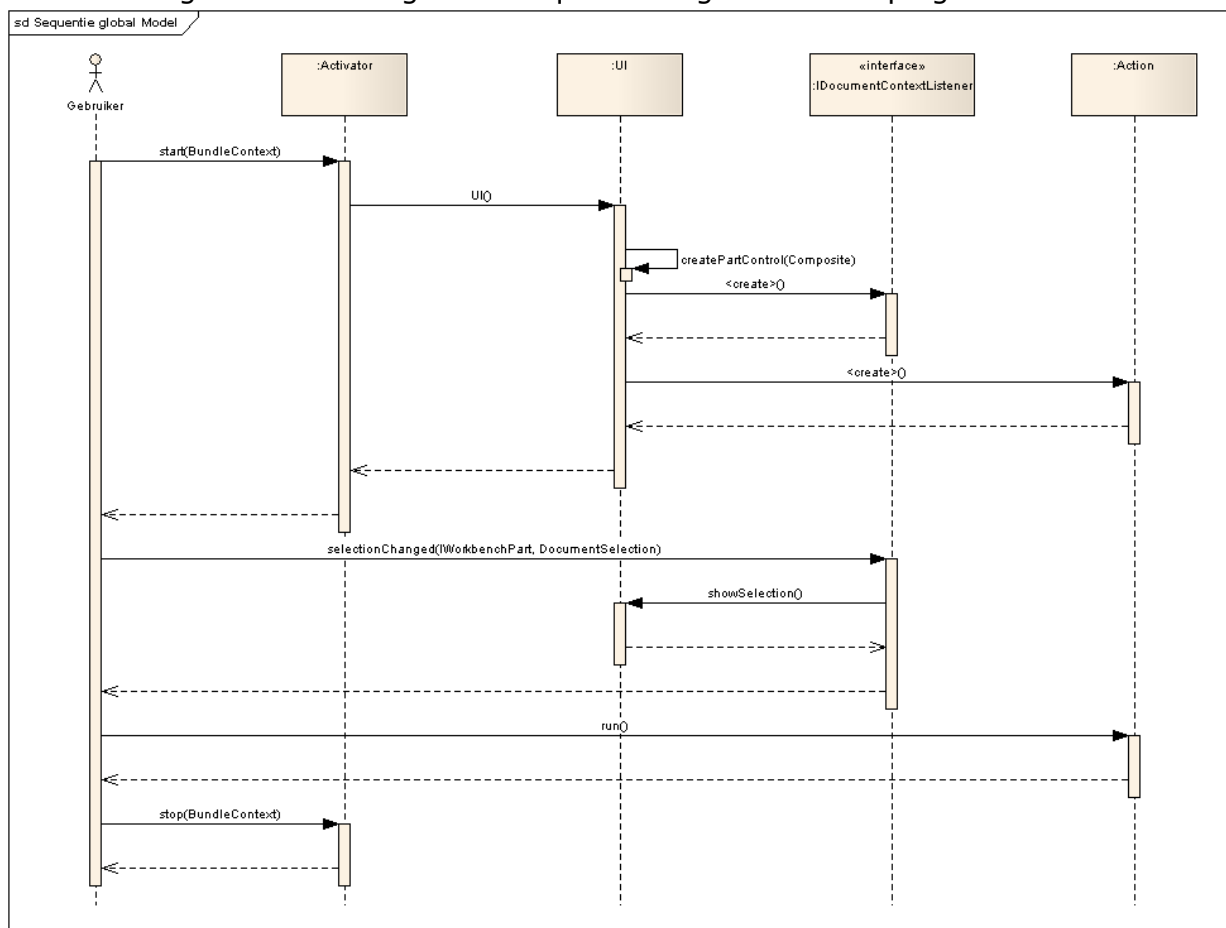
3 Sequentiediagram

In dit hoofdstuk staan de sequentiediagrammen. Deze diagrammen geven de interactie weer tussen de gebruiker en het systeem en tussen klassen in het systeem. Daarbij ligt de nadruk op het tijdsaspect. Niet alle diagrammen staan weergegeven in dit hoofdstuk, want sommige hebben daarvoor een te grote omvang. Die diagrammen staan wel in bijlage 1.

Dit hoofdstuk behandelt eerst het globale sequentiediagram en daarna de gedetailleerdere diagrammen.

3.1 Globaal diagram

In afbeelding 3.1 staat het globale sequentiediagram van de plug-in.



Afbeelding 3.1: globaal sequentiediagram van plug-in

In dit diagram komen de volgende functies aan bod:

5. Starten en opbouwen van de plug-in

6. Veranderen van de inhoud van de plug-in
7. Uitvoeren van acties
8. Sluiten van de plug-in

3.2 Gedetailleerde diagrammen

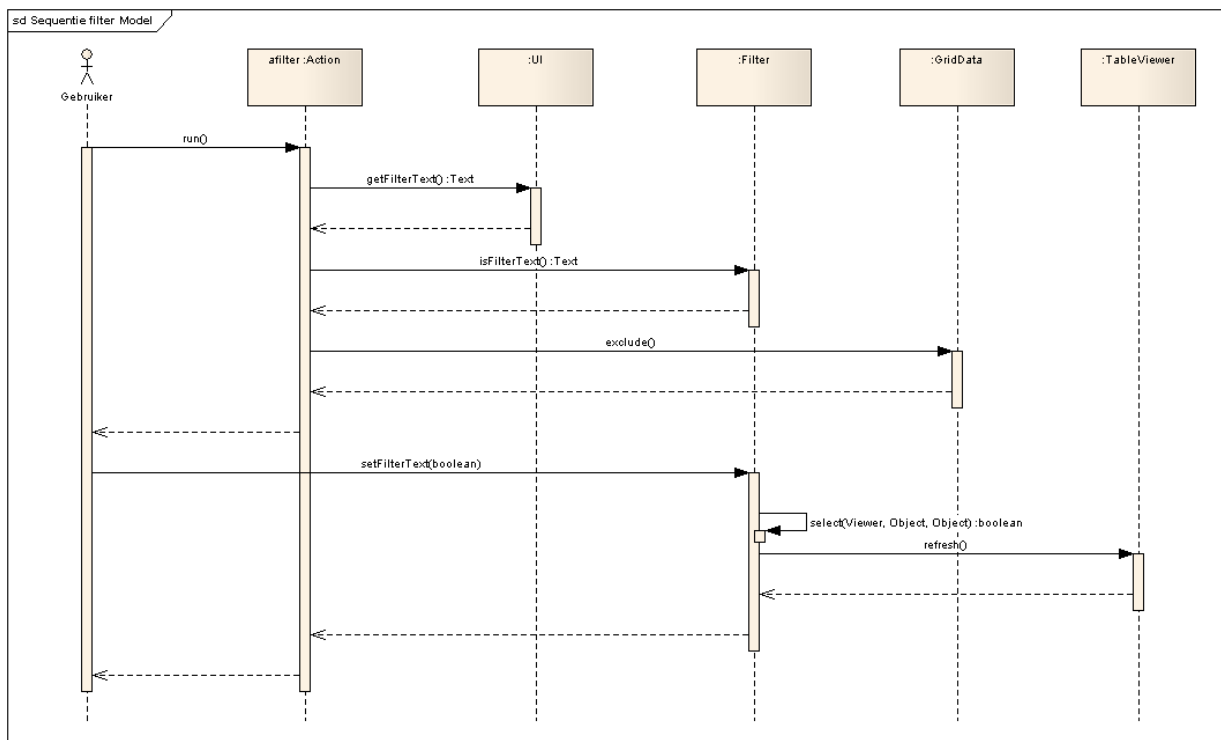
De gedetailleerde diagrammen gaan dieper in op gedeelten die in het globale diagram staan. Er zijn zes gedetailleerde diagrammen:

- Opbouw van de plug-in
- Opbouw van acties
- Verandering van geselecteerde document
- Actie filter
- Actie vastzetten van pagina
- Actie drop down menu

Voor de diagrammen van het opbouwen van de plug-in, het opbouwen van de acties en de verandering van geselecteerde document is een aparte uitleg niet van toegevoegde waarde. In die diagrammen worden vooral object aangemaakt.

3.2.1 Filter

In afbeelding 3.2 staat het diagram van de actie filter.

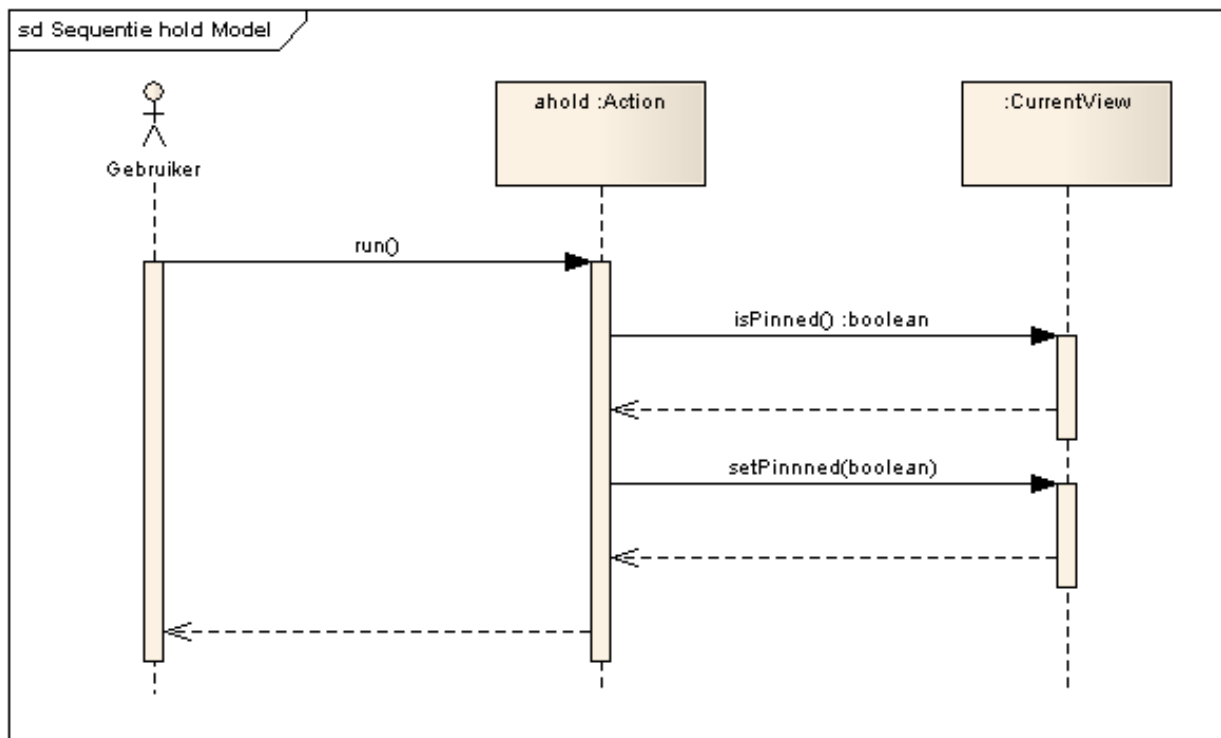


Afbeelding 3.2: actie filter

Als de gebruiker in de toolbar klikt op de filter dan verschijnt onder de toolbar een inputveld. Daar kan de gebruiker een voorwaarde typen waar de weer te geven informatie aan moet voldoen. Het inputveld heeft een listener die een functie aanroept als de gebruiker een waarde in het veld typt. Na elk teken wordt de informatie in de plug-in vernieuwd met de voorwaarde die de gebruiker heeft gegeven.

3.2.2 Vastzetten van pagina

In afbeelding 3.3 staat het diagram voor het vastzetten van de pagina.

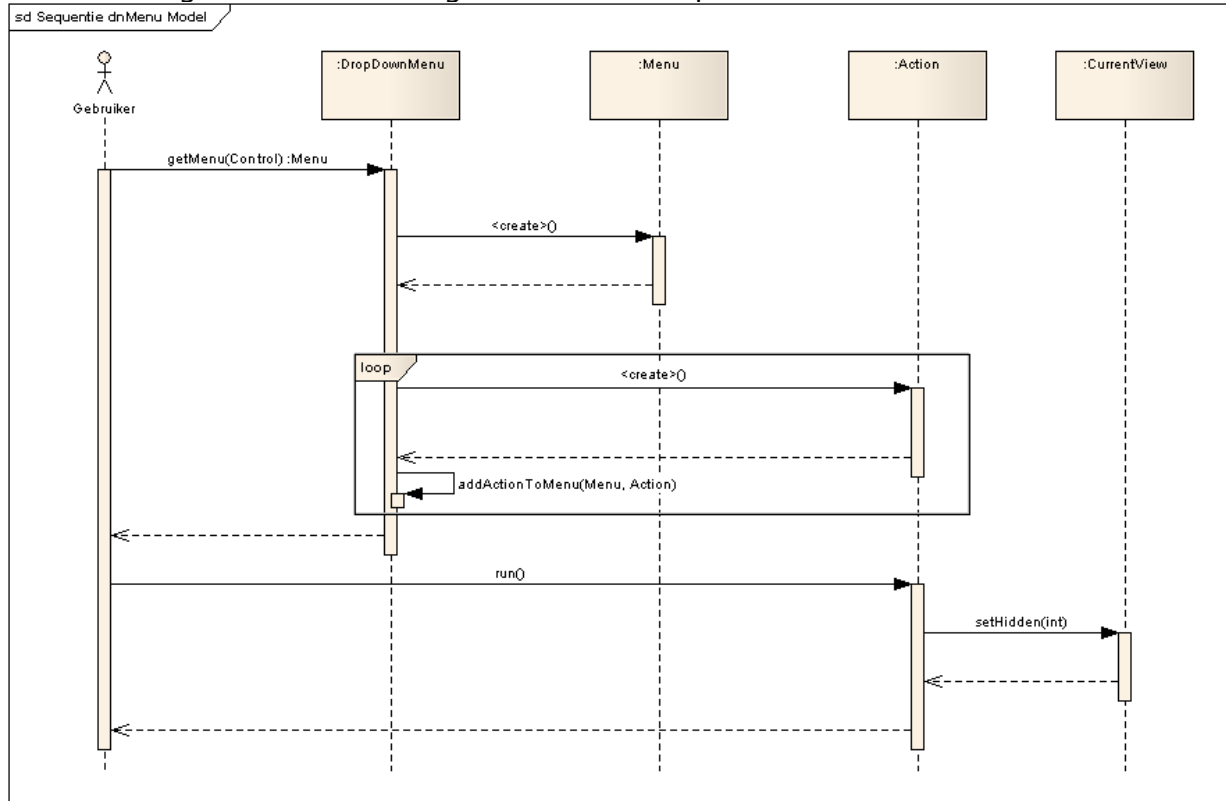


Afbeelding 3.3: actie vastzetten van de pagina

Als de gebruiker de pagina die op dat moment in de filter te zien is vastzet, dan wordt een boolean op true gezet. Deze actie is niet erg spectaculair, maar de gevolgen zijn groot. Als een pagina vaststaat in de plug-in kan de gebruiker de inhoud van de plug-in filteren en kolommen tonen en verbergen.

3.2.3 Drop down menu

In afbeelding 3.3 staat het diagram van het drop down menu.



Afbeelding 3.4: drop down menu

Het drop down menu is een menu van acties. Als de gebruiker op het menu klikt dan wordt het menu opgebouwd. Elke kolom is een menu item en elk menu item is een actie. Als de gebruiker op een menu item klikt wordt de kolom die daarbij hoort getoont als deze verborgen is en verborgen als deze getoont wordt.

4 Bronnen

4.1 Boeken

[Warner06] Warner, J. En Kleppe, A., Praktisch UML, 3^e druk, 2006.

4.2 Websites

[Eclipse2005] Eclipse,
http://help.eclipse.org/help32/index.jsp?topic=/org.eclipse.jet.doc/references/xpath/PLUGINS_ROOT/org.eclipse.platform.doc.isv/reference/api/overview-summary.html(20 mei 2010)

Bijlagen

1 Diagrammen

Deze bijlage wordt digitaal gestuurd, met dit document.

12 Testplan increment 2

Testplan

Generieke sidebar plug-in

voor IBM Lotus Notes & Domino



| | |
|---------------------|---------------------------------|
| Opdrachtgever: | F.J.C. Spruit |
| Bedrijfsmentor: | N.G. Scheppink |
| Begeleider: | D.R. Stikkolorum |
| Expert: | A.G.P. Pronk |
| Afstudeerder: | Chris den Arend |
| Email afstudeerder: | Chris.den.Arend@aedificommm.com |
| Plaats: | Barendrecht |
| Datum: | 09-04-2010 |
| Versie: | 1.0 |



Inleiding

Tijdens het ontwikkelen van software is het belangrijk dat de software getest wordt, want dat komt de kwaliteit ten goede. In dit document staan verschillende cases om het prototype uit increment twee te testen. De cases zijn gemaakt aan de hand van de eisen die in het document 'Increment 2' staan.

Na de testen wordt een testrapport gemaakt en dat rapport wordt gebruikt voor een eventuele volgende versie van de plug-in.

Het testen wordt uitgevoerd op systemen waar Lotus Notes en Domino al geïnstalleerd is. Het installatiebestand van de plug-in staat op de server waar Lotus Notes gebruikt van maakt.

De eisen die getest worden zijn eisen met prioriteit 'Must have'. Een uitzondering is het sorteren, want dat is niet geïmplementeerd. Het onthouden van de zichtbaarheid van kolommen en de waarde van de filter als Lotus Notes afgesloten wordt komt wel terug in het testplan, want dat is geïmplementeerd.

Windows XP (32 bit)

Case 1: Installatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker sleept de installatiefile (XML) in de sidebar van Lotus Notes, in het "my widgets" gedeelte.

Uitkomst:

De gebruiker kan aangeven of de plug-in geïnstalleerd moet worden. Als voor de installatie gekozen wordt begint deze en is de plug-in binnen 3 minuten geïnstalleerd. Nadat Lotus Notes opnieuw is opgestart is de plug-in zichtbaar in de sidebar en staat hij in "my widgets".

Test resultaat:

.....
.....
.....

Case 2: Informatie (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Relaties.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 3: Informatie (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Personeel.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 4: Informatie (3)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Post.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 5: Informatie (4)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Context Dossier.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 6: Filter (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Test resultaat:

.....
.....
.....

Case 7: Filter (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Test resultaat:

.....
.....
.....

Case 8: Filter (3)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Test resultaat:

.....
.....
.....

Case 9: Filter (4)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Test resultaat:

.....
.....
.....

Case 10: Filter (5)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Test resultaat:

.....
.....
.....

Case 11: Filter (6)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Test resultaat:

.....
.....
.....

Case 12: Filter (7)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Test resultaat:

.....
.....
.....

Case 13: Filter (8)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Test resultaat:

.....
.....
.....

Case 14: Vernieuwen (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker klikt op de knop "vernieuwen".

Uitkomst:

De plug-in wordt vernieuwd.

Test resultaat:

.....
.....
.....

Case 15: Vernieuwen (2)

Geslaagd: ja/nee

Beschrijving:

De timer loopt na 5 minuten af.

Uitkomst:

De plug-in wordt vernieuwd.

Test resultaat:

.....
.....
.....

Case 16: Weergave (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen.

Uitkomst:

De kolommen die de gebruiker verbergt zijn niet meer zichtbaar in de plug-in.

Test resultaat:

.....
.....
.....

Case 17: Weergave (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en toont één of meerdere kolommen.

Uitkomst:

De kolommen die de gebruiker toont zichtbaar in de plug-in.

Test resultaat:

.....
.....
.....

Case 18: Gemak (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties wordt weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 19: Gemak (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 20: Gemak (3)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 21: Gemak (4)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 22: Gemak (5)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 23: Gemak (6)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 24: Gemak (7)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 25: Gemak (8)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 26: Gemak (9)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Test resultaat:

.....
.....
.....

Case 27: Gemak (10)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Test resultaat:

.....
.....
.....

Case 28: Gemak (11)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Test resultaat:

.....

.....
.....

Case 29: Gemak (12)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Test resultaat:

.....
.....
.....

Case 30: Deinstallatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker klikt met de rechtermuisknop op de plug-in in "my widgets" en kiest remove.

Uitkomst:

De plug-in wordt binnen 3 minuten verwijderd, nadat Lotus Notes opnieuw is opgestart is de plug-in gedeïnstalleerd.

Test resultaat:

.....
.....
.....

Windows Vista (32 bit)

Case 1: Installatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker sleept de installatiefile (XML) in de sidebar van Lotus Notes, in het "my widgets" gedeelte.

Uitkomst:

De gebruiker kan aangeven of de plug-in geïnstalleerd moet worden. Als voor de installatie gekozen wordt begint deze en is de plug-in binnen 3 minuten geïnstalleerd. Nadat Lotus Notes opnieuw is opgestart is de plug-in zichtbaar in de sidebar en staat hij in "my widgets".

Test resultaat:

.....
.....
.....

Case 2: Informatie (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Relaties.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 3: Informatie (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Personeel.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 4: Informatie (3)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Post.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 5: Informatie (4)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Context Dossier.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 6: Filter (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Daarna klikt de gebruiker op de button filter.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Test resultaat:

.....
.....
.....

Case 7: Filter (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Daarna klikt de gebruiker op de button filter.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Test resultaat:

.....
.....
.....

Case 8: Filter (3)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Daarna klikt de gebruiker op de button filter.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Test resultaat:

.....
.....
.....

Case 9: Filter (4)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Daarna klikt de gebruiker op de button filter.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Test resultaat:

.....
.....
.....

Case 10: Filter (5)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Test resultaat:

.....
.....
.....

Case 11: Filter (6)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Test resultaat:

.....
.....
.....

Case 12: Filter (7)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Test resultaat:

.....
.....
.....

Case 13: Filter (8)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Test resultaat:

.....
.....
.....

Case 14: Vernieuwen (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker klikt op de knop "vernieuwen".

Uitkomst:

De plug-in wordt vernieuwd.

Test resultaat:

.....
.....
.....

Case 15: Vernieuwen (2)

Geslaagd: ja/nee

Beschrijving:

De timer loopt na 5 minuten af.

Uitkomst:

De plug-in wordt vernieuwd.

Test resultaat:

.....
.....
.....

Case 16: Weergave (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen.

Uitkomst:

De kolommen die de gebruiker verbergt zijn niet meer zichtbaar in de plug-in.

Test resultaat:

.....
.....
.....

Case 17: Weergave (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en toont één of meerdere kolommen.

Uitkomst:

De kolommen die de gebruiker toont zichtbaar in de plug-in.

Test resultaat:

.....
.....
.....

Case 18: Gemak (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties wordt weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 19: Gemak (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 20: Gemak (3)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 21: Gemak (4)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 22: Gemak (5)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 23: Gemak (6)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 24: Gemak (7)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 25: Gemak (8)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 26: Gemak (9)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Test resultaat:

.....
.....
.....

Case 27: Gemak (10)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Test resultaat:

.....
.....
.....

Case 28: Gemak (11)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Test resultaat:

.....

.....
.....

Case 29: Gemak (12)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Test resultaat:

.....
.....
.....

Case 30: Deinstallatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker klikt met de rechtermuisknop op de plug-in in "my widgets" en kiest remove.

Uitkomst:

De plug-in wordt binnen 3 minuten verwijderd, nadat Lotus Notes opnieuw is opgestart is de plug-in gedeïnstalleerd.

Test resultaat:

.....
.....
.....

Windows 7 (64 bit)

Case 1: Installatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker sleept de installatiefile (XML) in de sidebar van Lotus Notes, in het "my widgets" gedeelte.

Uitkomst:

De gebruiker kan aangeven of de plug-in geïnstalleerd moet worden. Als voor de installatie gekozen wordt begint deze en is de plug-in binnen 3 minuten geïnstalleerd. Nadat Lotus Notes opnieuw is opgestart is de plug-in zichtbaar in de sidebar en staat hij in "my widgets".

Test resultaat:

.....
.....
.....

Case 2: Informatie (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Relaties.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 3: Informatie (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Personeel.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 4: Informatie (3)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Post.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 5: Informatie (4)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Context Dossier.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Test resultaat:

.....
.....
.....

Case 6: Filter (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Daarna klikt de gebruiker op de button filter.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Test resultaat:

.....
.....
.....

Case 7: Filter (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Daarna klikt de gebruiker op de button filter.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Test resultaat:

.....
.....
.....

Case 8: Filter (3)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Daarna klikt de gebruiker op de button filter.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Test resultaat:

.....
.....
.....

Case 9: Filter (4)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Daarna klikt de gebruiker op de button filter.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Test resultaat:

.....
.....
.....

Case 10: Filter (5)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Test resultaat:

.....
.....
.....

Case 11: Filter (6)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Test resultaat:

.....
.....
.....

Case 12: Filter (7)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Test resultaat:

.....
.....
.....

Case 13: Filter (8)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Test resultaat:

.....
.....
.....

Case 14: Vernieuwen (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker klikt op de knop "vernieuwen".

Uitkomst:

De plug-in wordt vernieuwd.

Test resultaat:

.....
.....
.....

Case 15: Vernieuwen (2)

Geslaagd: ja/nee

Beschrijving:

De timer loopt na 5 minuten af.

Uitkomst:

De plug-in wordt vernieuwd.

Test resultaat:

.....
.....
.....

Case 16: Weergave (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen.

Uitkomst:

De kolommen die de gebruiker verbergt zijn niet meer zichtbaar in de plug-in.

Test resultaat:

.....
.....
.....

Case 17: Weergave (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en toont één of meerdere kolommen.

Uitkomst:

De kolommen die de gebruiker toont zichtbaar in de plug-in.

Test resultaat:

.....
.....
.....

Case 18: Gemak (1)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties wordt weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 19: Gemak (2)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 20: Gemak (3)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 21: Gemak (4)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 22: Gemak (5)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 23: Gemak (6)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 24: Gemak (7)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 25: Gemak (8)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Test resultaat:

.....
.....
.....

Case 26: Gemak (9)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Test resultaat:

.....
.....
.....

Case 27: Gemak (10)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Test resultaat:

.....
.....
.....

Case 28: Gemak (11)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Test resultaat:

.....

.....
.....

Case 29: Gemak (12)

Geslaagd: ja/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Test resultaat:

.....
.....
.....

Case 30: Deinstallatie

Geslaagd: ja/nee

Beschrijving:

De gebruiker klikt met de rechtermuisknop op de plug-in in "my widgets" en kiest remove.

Uitkomst:

De plug-in wordt binnen 3 minuten verwijderd, nadat Lotus Notes opnieuw is opgestart is de plug-in gedeïnstalleerd.

Test resultaat:

.....
.....
.....

13 Testrapport increment 2

Testrapport

Generieke sidebar plug-in

voor IBM Lotus Notes & Domino

AedifiComm

| | |
|---------------------|--------------------------------|
| Opdrachtgever: | F.J.C. Spruit |
| Bedrijfsmentor: | N.G. Scheppink |
| Begeleider: | D.R. Stikkolorum |
| Expert: | A.G.P. Pronk |
| Afstudeerder: | Chris den Arend |
| Email afstudeerder: | Chris.den.Arend@aedificomm.com |
| Plaats: | Barendrecht |
| Datum: | 26-05-2010 |
| Versie: | 1.0 |



Inleiding

Tijdens het ontwikkelen van software is het belangrijk dat de software getest wordt, want dat komt de kwaliteit ten goede. In dit document staan verschillende cases om het prototype uit increment twee te testen. De cases zijn gemaakt aan de hand van de eisen die in het document 'Increment 2' staan.

Na de testen wordt een testrapport gemaakt en dat rapport wordt gebruikt voor een eventuele volgende versie van de plug-in.

Het testen wordt uitgevoerd op systemen waar Lotus Notes en Domino al geïnstalleerd is. Het installatiebestand van de plug-in staat op de server waar Lotus Notes gebruikt van maakt.

De eisen die getest worden zijn eisen met prioriteit 'Must have'. Een uitzondering is het sorteren, want dat is niet geïmplementeerd. Het onthouden van de zichtbaarheid van kolommen en de waarde van de filter als Lotus Notes afgesloten wordt komt wel terug in het testplan, want dat is geïmplementeerd.

Windows XP (32 bit)

Case 1: Installatie

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker sleept de installatiefile (XML) in de sidebar van Lotus Notes, in het "my widgets" gedeelte.

Uitkomst:

De gebruiker kan aangeven of de plug-in geïnstalleerd moet worden. Als voor de installatie gekozen wordt begint deze en is de plug-in binnen 3 minuten geïnstalleerd. Nadat Lotus Notes opnieuw is opgestart is de plug-in zichtbaar in de sidebar en staat hij in "my widgets".

Test resultaat:

De plug-in is geïnstalleerd in 56 seconden.

Case 2: Informatie (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Relaties.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Case 3: Informatie (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Personeel.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Case 4: Informatie (3)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Post.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Case 5: Informatie (4)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Context Dossier.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Case 6: Filter (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Case 7: Filter (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Case 8: Filter (3)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Case 9: Filter (4)

Geslaagd: **ja**/~~nee~~

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Case 10: Filter (5)

Geslaagd: **ja**/~~nee~~

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Case 11: Filter (6)

Geslaagd: **ja**/~~nee~~

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Case 12: Filter (7)

Geslaagd: **ja**/~~nee~~

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Case 13: Filter (8)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Case 14: Vernieuwen (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker klikt op de knop "vernieuwen".

Uitkomst:

De plug-in wordt vernieuwd.

Case 15: Vernieuwen (2)

Geslaagd: **ja**/nee

Beschrijving:

De timer loopt na 5 minuten af.

Uitkomst:

De plug-in wordt vernieuwd.

Case 16: Weergave (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen.

Uitkomst:

De kolommen die de gebruiker verbergt zijn niet meer zichtbaar in de plug-in.

Case 17: Weergave (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en toont één of meerdere kolommen.

Uitkomst:

De kolommen die de gebruiker toont zichtbaar in de plug-in.

Case 18: Gemak (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties wordt weergegeven in de plug-in.

Case 19: Gemak (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt weergegeven in de plug-in.

Case 20: Gemak (3)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt weergegeven in de plug-in.

Case 21: Gemak (4)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt weergegeven in de plug-in.

Case 22: Gemak (5)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Case 23: Gemak (6)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Case 24: Gemak (7)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Case 25: Gemak (8)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Case 26: Gemak (9)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Case 27: Gemak (10)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Case 28: Gemak (11)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Case 29: Gemak (12)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Case 30: Deinstallatie

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker klikt met de rechtermuisknop op de plug-in in "my widgets" en kiest remove.

Uitkomst:

De plug-in wordt binnen 3 minuten verwijderd, nadat Lotus Notes opnieuw is opgestart is de plug-in gedeïnstalleerd.

Test resultaat:

De plug-in is gedeïnstalleerd in 35 seconden.

Windows Vista (32 bit)

Case 1: Installatie

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker sleept de installatiefile (XML) in de sidebar van Lotus Notes, in het "my widgets" gedeelte.

Uitkomst:

De gebruiker kan aangeven of de plug-in geïnstalleerd moet worden. Als voor de installatie gekozen wordt begint deze en is de plug-in binnen 3 minuten geïnstalleerd. Nadat Lotus Notes opnieuw is opgestart is de plug-in zichtbaar in de sidebar en staat hij in "my widgets".

Test resultaat:

De plug-in is geïnstalleerd in 1 minuut en 3 seconden.

Case 2: Informatie (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Relaties.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Case 3: Informatie (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Personeel.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Case 4: Informatie (3)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Post.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Case 5: Informatie (4)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Context Dossier.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Case 6: Filter (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Case 7: Filter (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Case 8: Filter (3)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Case 9: Filter (4)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Case 10: Filter (5)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Case 11: Filter (6)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Case 12: Filter (7)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Case 13: Filter (8)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Case 14: Vernieuwen (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker klikt op de knop "vernieuwen".

Uitkomst:

De plug-in wordt vernieuwd.

Case 15: Vernieuwen (2)

Geslaagd: **ja**/nee

Beschrijving:

De timer loopt na 5 minuten af.

Uitkomst:

De plug-in wordt vernieuwd.

Case 16: Weergave (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen.

Uitkomst:

De kolommen die de gebruiker verbergt zijn niet meer zichtbaar in de plug-in.

Case 17: Weergave (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en toont één of meerdere kolommen.

Uitkomst:

De kolommen die de gebruiker toont zichtbaar in de plug-in.

Case 18: Gemak (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties wordt weergegeven in de plug-in.

Case 19: Gemak (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt weergegeven in de plug-in.

Case 20: Gemak (3)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt weergegeven in de plug-in.

Case 21: Gemak (4)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt weergegeven in de plug-in.

Case 22: Gemak (5)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Case 23: Gemak (6)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Case 24: Gemak (7)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Case 25: Gemak (8)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Case 26: Gemak (9)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Case 27: Gemak (10)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Case 28: Gemak (11)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Case 29: Gemak (12)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Case 30: Deinstallatie

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker klikt met de rechtermuisknop op de plug-in in "my widgets" en kiest remove.

Uitkomst:

De plug-in wordt binnen 3 minuten verwijderd, nadat Lotus Notes opnieuw is opgestart is de plug-in gedeïnstalleerd.

Test resultaat:

De plug-in is gedeïnstalleerd in 42 seconden.

Windows 7 (64 bit)

Case 1: Installatie

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker sleept de installatiefile (XML) in de sidebar van Lotus Notes, in het "my widgets" gedeelte.

Uitkomst:

De gebruiker kan aangeven of de plug-in geïnstalleerd moet worden. Als voor de installatie gekozen wordt begint deze en is de plug-in binnen 3 minuten geïnstalleerd. Nadat Lotus Notes opnieuw is opgestart is de plug-in zichtbaar in de sidebar en staat hij in "my widgets".

Test resultaat:

De plug-in is geïnstalleerd in 53 seconden.

Case 2: Informatie (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Relaties.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Case 3: Informatie (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Personeel.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Case 4: Informatie (3)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Post.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Case 5: Informatie (4)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert achter elkaar elke gegevensbron in CSP in de database Context Dossier.

Uitkomst:

De plug-in geeft informatie weer over de geselecteerde gegevensbron sneller dan wanneer de gebruiker naar de module navigeert.

Case 6: Filter (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Case 7: Filter (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Case 8: Filter (3)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Case 9: Filter (4)

Geslaagd: **ja**/~~nee~~

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Daarna klikt de gebruiker op de button filter en op de button kolommen.

Uitkomst:

Als de gebruiker op de button filter klikt gebeurt er niets.

Case 10: Filter (5)

Geslaagd: **ja**/~~nee~~

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Case 11: Filter (6)

Geslaagd: **ja**/~~nee~~

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Case 12: Filter (7)

Geslaagd: **ja**/~~nee~~

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Case 13: Filter (8)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold'. Daarna klikt de gebruiker op de button 'filter' en kiest de kolom waarop gefiltert moet worden. Als laatste wordt de filterwaarde ingevuld.

Uitkomst:

In de plug-in is informatie te zien die overeenkomt met de filterwaarde.

Case 14: Vernieuwen (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker klikt op de knop "vernieuwen".

Uitkomst:

De plug-in wordt vernieuwd.

Case 15: Vernieuwen (2)

Geslaagd: **ja**/nee

Beschrijving:

De timer loopt na 5 minuten af.

Uitkomst:

De plug-in wordt vernieuwd.

Case 16: Weergave (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen.

Uitkomst:

De kolommen die de gebruiker verbergt zijn niet meer zichtbaar in de plug-in.

Case 17: Weergave (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en toont één of meerdere kolommen.

Uitkomst:

De kolommen die de gebruiker toont zichtbaar in de plug-in.

Case 18: Gemak (1)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties wordt weergegeven in de plug-in.

Case 19: Gemak (2)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt weergegeven in de plug-in.

Case 20: Gemak (3)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt weergegeven in de plug-in.

Case 21: Gemak (4)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold'. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt weergegeven in de plug-in.

Case 22: Gemak (5)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Case 23: Gemak (6)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Case 24: Gemak (7)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Case 25: Gemak (8)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en stelt de filter in. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt gefiltert op de waarde die voor het herstarten ingevoerd was weergegeven in de plug-in.

Case 26: Gemak (9)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron alle actieve relaties in de database Relaties. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron alle actieve relaties uit de database Relaties weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Case 27: Gemak (10)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron medewerkers per afdeling in de database Personeel. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron medewerkers per afdeling in de database Personeel wordt weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Case 28: Gemak (11)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron uitgaande post per client in de database Post. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron gegevensbron uitgaande post per client in de database Post wordt in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Case 29: Gemak (12)

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker selecteert de gegevensbron dossier overzicht van sjablonen in de database Context Dossier. Dan klikt de gebruiker op de button 'hold' en verbergt één of meerdere kolommen. Dan wordt CSP opnieuw opgestart.

Uitkomst:

De gegevensbron dossier overzicht van sjablonen in de database Context Dossier wordt weergegeven in de plug-in en de kolommen die voor het herstarten verborgen waren zijn dat nog steeds.

Case 30: Deinstallatie

Geslaagd: **ja**/nee

Beschrijving:

De gebruiker klikt met de rechtermuisknop op de plug-in in "my widgets" en kiest remove.

Uitkomst:

De plug-in wordt binnen 3 minuten verwijderd, nadat Lotus Notes opnieuw is opgestart is de plug-in gedeïnstalleerd.

Test resultaat:

De plug-in is gedeïnstalleerd in 25 seconden.

14 Stappenplan updatesite (1)

Deze bijlage wordt digitaal gestuurd, met dit document.

15 Stappenplan updatesite (2)

Deze bijlage wordt digitaal gestuurd, met dit document.

16 Stappenplan updatesite (3)

Deze bijlage wordt digitaal gestuurd, met dit document.

17 Checklist volledigheid plan van aanpak

© 2008 Roel Grit/Wolters-Noordhoff bv Groningen/Houten

Projectmanagement 1

Checklist volledigheid plan van aanpak

Deze checklist hoort bij het boek *Projectmanagement* ISBN 978 90 01 80262 2 door Roel Grit, uitgeverij Wolters-Noordhoff.

Zie ook <http://www.grit-projectmanagement.wolters.nl>.

A Algemeen, lay-out en taalgebruik van het plan van aanpak

Is het voorblad compleet?

- ☐ Bevat het de tekst: 'Plan van aanpak'?
- ☐ Bevat het de naam van het project?
- ☐ Bevat het eventueel een ondertitel?
- ☐ Bevat het de naam van de organisatie en de naam van de opdrachtgever?
- ☐ Bevat het de plaats en datum van gereedkomen van het plan van aanpak?
- ☐ Is het groepsnummer/de code van de projectgroep aanwezig?
- ☐ Bevat het de naam van de opdrachtnemer?
- ☐ Bevat het alle correcte namen van alle projectleden en hun correcte mailadres?

Lay-out

- ☐ Is een inhoudsopgave met paginanummers aanwezig?
- ☐ Zijn de hoofdstukken genummerd?
- ☐ Zijn de paginanummers op de bladzijden aanwezig?
- ☐ Zijn de hoofdstukken duidelijk aangegeven?
- ☐ Klopt de hoofdstukindeling met hoofdstuk 5 en zijn alle hoofdstukken aanwezig?

Taalgebruik

- ☐ Is het plan van aanpak helder geschreven?
- ☐ Wordt er correcte taal gebruikt (gebruik van spelling- en grammaticacontrolle)?
- ☐ Is de taal zakelijk? Geen ik, jij, jullie!
- ☐ Worden niet te lange zinnen gebruikt?
- ☐ Is het plan van aanpak 'puntig' geschreven, met een duidelijke structuur en opsommingen?
- ☐ Worden onbekende begrippen eerst uitgelegd alvorens ze worden gebruikt?
- ☐ Vindt geen rolwisseling plaats tijdens het plan van aanpak? Dat wil zeggen: wordt het plan van aanpak steeds vanuit dezelfde persoon bekeken?

B De hoofdstukken van het Plan van Aanpak

Hoofdstuk 1: Achtergronden

- ☐ Is een korte beschrijving van de organisatie waarin het project zich afspeelt opgenomen?
- ☐ Wat 'doet' de organisatie?
- ☐ Is duidelijk welke afdelingen bij het project zijn betrokken?
- ☐ Zijn de achtergronden van 'groot' naar 'klein' beschreven?
- ☐ Bijvoorbeeld: eerst het bedrijf, dan de vestiging, vervolgens de afdeling, dan het project.

© 2008 Roel Grit/Wolters-Noordhoff bv Groningen/Houten

2 Projectmanagement

- ☐ Is dit project een vervolg op een eerder project?
- ☐ Hoe is dat project afgelopen?
- ☐ Is een beschrijving van de geschiedenis van het project aanwezig?
- ☐ Is duidelijk waarom de opdrachtgever dit project wil?
- ☐ Welke invloed heeft het eerdere project op het huidige project?
- ☐ Welke relatie heeft dit project met andere uit te voeren projecten?
- ☐ Is het duidelijk welke partijen een rol spelen bij het project (interne partijen, externe partijen)?
- ☐ Wat is de rolverdeling tussen de partijen?
- ☐ Is het duidelijk wat de relatie is tussen de opdrachtgever, de projectgroep en andere partijen?
- ☐ Is duidelijk welke de opdrachtgevende organisatie is?

- ☐ Is duidelijk wie de opdrachtgever (persoon) is?
- ☐ Is duidelijk welke de opdrachtnemende organisatie is?
- ☐ Is duidelijk wie de opdrachtnemende persoon (projectleider) is?
- ☐ Is er eventueel een schriftelijke opdracht aanwezig?

Hoofdstuk 2: Projectopdracht

- ☐ Wordt duidelijk waarom het project wordt uitgevoerd?
- ☐ Zijn de doelstellingen duidelijk?
- ☐ Zijn de doelstellingen Smart?
- ☐ Zijn er eventueel subdoelstellingen geformuleerd?
- ☐ Heeft het project een duidelijke, liefst originele, pakkende naam?
- ☐ Is de opdracht een gevolg van de doelstelling of probleeminstelling?
- ☐ Is exact duidelijk wat het eindproduct van het project is (inclusief de eisen die aan het product worden gesteld)?
- ☐ Is het eindproduct voldoende gespecificeerd?

Hoofdstuk 3: Projectactiviteiten

- ☐ Is er een puntsgewijze opsomming van activiteiten?
- ☐ Zijn er voldoende activiteiten gedefinieerd?
- ☐ Zijn er geen activiteiten vergeten?
- ☐ Zijn de activiteiten gegroepeerd?
- ☐ Zijn de activiteiten binnen een groep van dezelfde 'orde van grootte'?
- ☐ Is het plan van aanpak zelf ook als activiteit opgenomen?
- ☐ Is de afhandeling van het project niet vergeten?

Hoofdstuk 4: Projectgrenzen

- ☐ Is voldoende aangegeven tot hoever het project doorgaat ('lengte')?
- ☐ Is voldoende duidelijk wat niet bij het project hoort ('breedte')?
- ☐ Is duidelijk aan welke voorwaarden moet worden voldaan voor het welslagen van het project?

Hoofdstuk 5: Producten

- ☐ Zijn voldoende producten gedefinieerd?
- ☐ Is het plan van aanpak ook als product gedefinieerd?
- ☐ Zijn eventuele belangrijke gebeurtenissen ook als product gedefinieerd?
- ☐ Zijn ontwerpen, rapporten en dergelijke niet vergeten?
- ☐ Is het eindproduct niet vergeten?

© 2008 Roel Grit/Wolters-Noordhoff bv Groningen/Houten

Projectmanagement 3

Hoofdstuk 6: Kwaliteit

- ☐ Is de waarborging van de kwaliteit van het eindproduct beschreven?
- ☐ Is de waarborging van de kwaliteit van de tussenproducten beschreven?
- ☐ Zijn voldoende controles ingebouwd om de kwaliteit te bewaken?
- ☐ Zijn er normen te controle van de kwaliteit genoemd?
- ☐ Wordt afgeweken van de heersende normen?
- ☐ Zijn terugkoppelingen met opdrachtgever genoemd?
- ☐ Zijn gebruikte methoden genoemd?
- ☐ Zijn gebruikte technieken genoemd?
- ☐ Zijn afspraken opgenomen over te gebruiken hulpmiddelen (onder andere software)?
- ☐ Wordt het vragen van extern advies genoemd?
- ☐ Worden producten in fasen opgeleverd (bijvoorbeeld conceptrapporten, goedkeuring, definitief rapport)?

Hoofdstuk 7: Projectorganisatie

Organisatie

- ☐ Zijn de functies binnen de projectgroep verdeeld?
- ☐ Staan er namen bij de functies?
- ☐ Zijn namen, adressen, telefoonnummers en e-mailadressen van alle projectleden genoemd?
- ☐ Is het projectsecretariaat geregeld?
- ☐ Is de beschikbaarheid van de deelnemers genoemd?
- ☐ Zijn bevoegdheden vastgelegd?

Informatie

- ☐ Is genoemd hoe de communicatie met de opdrachtgever is geregeld?
- ☐ Is genoemd hoe de interne communicatie binnen de projectgroep is geregeld?

- ☐ Is de urenverantwoording of tijdregistratie geregeld?
- ☐ Is de vergaderfrequentie geregeld?
- ☐ Is voorzien in een weekverslag door projectleden aan de projectleider?
- ☐ Is voorzien in een weekverslag door de projectleider aan de opdrachtgever?
- ☐ Is de archivering geregeld?

Hoofdstuk 8: Planning

- ☐ Klopt de planning met de activiteiten uit hoofdstuk 3?
- ☐ Is een computerplanning aanwezig?
- ☐ Is de planning realistisch?
- ☐ Is rekening gehouden met vertragende besluitvorming?

Hoofdstuk 9: Kosten en baten

- ☐ Bevat het overzicht voldoende informatie?
- ☐ Bevat het overzicht een korte toelichting?
- ☐ Zijn de baten niet vergeten?
- ☐ Zijn eventuele aannames uitgelegd?
- ☐ Is het overzicht realistisch?
- ☐ Zijn de kosten en baten niet te precies opgegeven? (Het gaat immers om een schatting).

© 2008 Roel Grit/Wolters-Noordhoff bv Groningen/Houten

4 Projectmanagement

Hoofdstuk 10: Risico's

- ☐ Stelt de opdrachtgever dit hoofdstuk op prijs?
- ☐ Is de bijlage 'Risicoanalyse' gebruikt?
- ☐ Als het spreadsheetmodel (Excel) is gebruikt, is dan een conclusie getrokken?
- ☐ Is het hoofdstuk voldoende degelijk?

18 IBM install Expeditor Toolkit

Deze bijlage wordt digitaal gestuurd, met dit document.