



Technische Universiteit Delft

Hogeschool  
van Utrecht



Advies- en Ingenieursbureau



# KOOP

Koppeling Onderzoek, Onderwijs en Praktijk

## Scriptie

Auteur: Erwin Hendriks

Mediatheek HvU



0300 524 0563

SCRIPTIE IE 2004 HEND



Technische Universiteit Delft

Hogeschool  
van Utrecht

Advies- en Ingenieursbureau



# KOOP

Koppeling Onderzoek, Onderwijs en Praktijk

MEDIATHEEK HOOGESCHOOL VAN UTRECHT  
FACULTEIT TECHNISCHE WETENSCHAPPEN  
DECEKADREES, N. 1106061  
POSTADRES: PO BOX 12142  
3500 AD UTRECHT  
TEL. 030 - 230 51 48

Student:	E.N.J. Hendriks
Studentnr:	1106061
Datum:	september 2003 – januari 2004
Eerste examiner:	Dhr. H. Plessius
Stageplaats:	DHV Water Leusden / Amersfoort
Begeleider	Dhr. P. Nijdam



## **Samenvatting**

Dit document is een afstudeerscriptie geschreven door Erwin Hendriks voor de opleiding Information Engineering. Het afstuderen heeft plaats gevonden bij DHV Water in opdracht van de TU Delft.

De scriptie heeft betrekking op het project KOOP. KOOP staat voor Koppeling, Onderwijs, Onderzoek en Praktijk. De applicatie is bereikbaar via een website waarbij waterzuiveringen gesimuleerd kunnen worden. Door de presentatie van simulatie resultaten wordt het voor de gebruiker inzichtelijk hoe effectief de waterzuiveringsprocessen werken.

Het project is vooraf gegaan door het project KOOP – Demo. Koop – Demo was een test-applicatie waarbij de haalbaarheid van het concept getoetst is.

De applicatie KOOP is opgezet als module in een portal. In de portal zijn verschillende veel voorkomende functies verwerkt. Hierdoor kan alle aandacht gaan naar het project KOOP.

Het project KOOP is een onderdeel van het totale project Stimela. Onder Stimela valt o.a. het programmeren van de simulatieomgeving (project KOOP), het ontwerpen en programmeren van zuiveringsmodellen en het definiëren van verschillende water en zuiverings instellingen.

De hoofddoelen van het afstudeerproject KOOP zijn:

- Een dynamische omgeving creëren waarbij mogelijkheden zijn tot uitbreidingen.
- Onderzoeken naar de meest gebruikersvriendelijke interface.

Er is een database structuur opgezet waar de wateren (waters) en waterzuiveringsopstellingen (plants) worden opgeslagen. Daarnaast worden bij het uitvoeren van een simulatie de gesimuleerde resultaten opgeslagen waardoor de gegevens van de simulatie sneller toegankelijk zijn. Voor het maken van de applicatie is gebruik gemaakt van de .NET in combinatie met een SQL database.

Door dit project is het duidelijker geworden hoe de gebruikers interface eruit hoort te zien, hoe de werking en opzet van de applicatie kan zijn en hoe de database is opgebouwd. Dit kan als basis dienen voor verdere uitwerkingen.

Tijdens het realiseren van het project is er geconstateerd dat de opzet van de database enkele aanpassingen vereist. Deze aanpassingen hebben geen invloed op dit project, maar zijn wel wenselijk voor het eindresultaat van Stimela.

## **Voorwoord**

Voor het maken van dit document en het uitvoeren van het project hebben verschillende mensen mij terzijde gestaan.

Hierbij gaat het om mijn afstudeerbegeleider van de Hogeschool van Utrecht, Henk Plessius. Hij heeft mij geholpen met het opzetten van mijn scriptie en aanverwante documenten.

De stagebegeleider bij DHV Water, Pieter Nijdam. P. Nijdam heeft mij geholpen met de programmering en ontwikkeling van het Project KOOP.

De projectleider van het KOOP Project, Peter de Moel.

P. de Moel heeft geholpen bij de ontwikkeling van de GUI en deze ook van de nodige commentaren voorzien.

Bedankt voor jullie inspanning en toewijding.

## Inhoudsopgave

	pagina
<b>Hoofdstuk 1. Inleiding.....</b>	<b>6</b>
1.1    Uitgangssituatie.	7
1.2    Scriptie opbouw.	8
<b>Hoofdstuk 2. De organisatie.....</b>	<b>9</b>
2.1    Stakeholders	9
2.2    Betrokkenen	10
2.3    Opdrachtgever: TU Delft	11
2.4    Uitvoerder: DHV	12
2.4.1    PA-ICT	12
2.5    De Afstudeerder	12
2.6    Hogeschool van Utrecht	12
<b>Hoofdstuk 3. Projectomschrijving.....</b>	<b>13</b>
3.1    Doelstelling	13
3.2    Kwaliteitseisen	13
3.3    Succesfactor	13
3.4    Opdracht	14
3.5    Samenvatting Plan van aanpak	15
3.6    Aanpakmethode	16
3.6.1    Ontwerp & Bouw Iteratie	17
3.6.2    Implementatie	17
3.6.3    Documentatie	17
3.7    Het haalbaarheidsonderzoek en bedrijfsanalyse	17
<b>Hoofdstuk 4. Projectbespreking.....</b>	<b>18</b>
4.1    Architectuur	19
4.1.1    Architectuur	19
4.1.2    Opzet programmering	20
4.2    Implementatie aanpak	21
4.3    Programmering	22
4.3.1    Portals	22
4.3.2    Programmering Stimela	23
4.4    Database	24
4.4.1    KOOP - Demo Database opzet	25
4.4.2    Opzet KOOP Database	26
4.4.3    Uiteindelijke database opzet KOOP	27
4.5    Bespreking functies	28
4.6    Bespreking projectdoorloop	30
4.7    Resultaat, commentaar, acties en aanbeveling van de prototype	31
<b>Hoofdstuk 5. Evaluatie.....</b>	<b>34</b>
<b>Hoofdstuk 6. Aanbevelingen.....</b>	<b>35</b>
<b>Hoofdstuk 7. Conclusie.....</b>	<b>37</b>
<b>Hoofdstuk 8. Evaluatie.....</b>	<b>38</b>
<b>Literatuurlijst</b>	<b>39</b>
<b>Bijlage 1: Plan van aanpak</b>	<b>40</b>
<b>Bijlage 2: Functioneel ontwerp</b>	<b>59</b>
<b>Bijlage 3: Technisch ontwerp</b>	<b>83</b>
<b>Bijlage 4: Portals</b>	<b>100</b>

## Hoofdstuk 1. Inleiding.

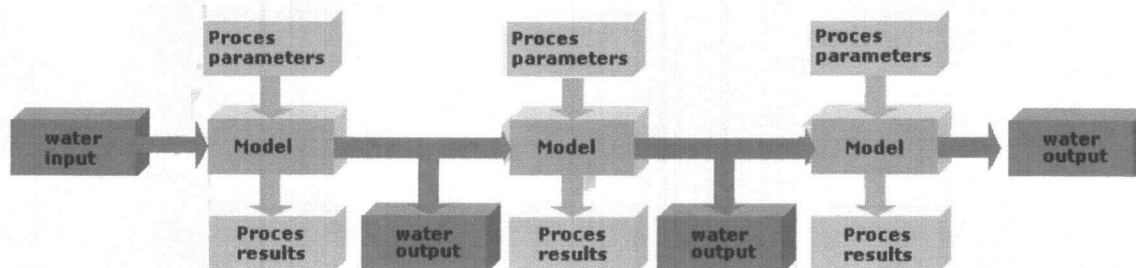
Er zijn verschillende typen water in de wereld, hiervan kan men niet elk type water drinken of lozen in openbare wateren. Dit water moet eerst gezuiverd worden. Omdat elk type water uit verschillende eigenschappen bestaat, zal er bij elk type water weer een andere zuivering moeten worden toegepast. Deze zuiveringen kunnen simpele zuiveringen bij industrieën zijn of zuiveringen voor de complete drinkwatervoorziening van een stad.

De stageopdracht is een vervolg van een eerder uitgevoerd project: "KOOP - Demo". Het project KOOP staat voor Koppeling Onderzoek, Onderwijs en Praktijk voor drinkwater en wordt uitgevoerd in opdracht van de TU Delft. De opzet van KOOP is het simuleren van waterzuiveringsopstellingen. De Universiteit kan deze gebruiken als lesmateriaal en voor het ontwerpen en testen van nieuwe modellen / opstellingen.

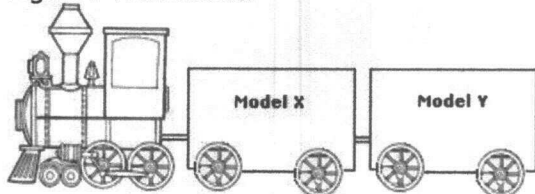
Momenteel zijn er modellen die de berekeningen voor waterzuivering uitvoeren. Alleen zijn deze modellen van elk proces afzonderlijk (geen geïntegreerde modellen) en zijn deze niet of beperkt toegankelijk via het Internet. Daarnaast is voor het opzetten van modellen gespecialiseerde kennis nodig en hierdoor moeilijk als leermodel in te zetten. Met dit project is het mogelijk om vanaf het Internet, eventueel tegen betaling, berekeningen voor de waterzuiveringsprocessen en de gehele waterzuivering te simuleren.

KOOP is een applicatie waarmee waterzuiveringen worden gesimuleerd. Elke gebruiker moet een bepaalde input (water input, zie Figuur "Blok-schema") geven aan de simulatie. Op dit type water wordt een waterzuivering toegepast. Elke waterzuiveringsbedrijf (plant) is een aaneenschakeling van waterzuiveringsprocessen (modellen). Elk proces heeft een water en een procesparameterset waarmee gerekend wordt. Elke model geeft een dataset terug waarin de wateroutput en de procesresultaten zijn verwerkt. Na het simuleren van een waterzuivering krijgt de gebruiker een presentatie van het behaalde eindresultaat en alle tussenresultaten van het water na de toepassing van elk model.

Figuur: Blokschema



Figuur: Treinmodel



De applicatie is te vergelijken met een modeltrein waar iedere modelmaker zijn eigen wagon (model) kan maken. Al deze wagonnetjes kunnen achter elkaar gezet worden totdat de trein volledig is (plant). Het doel van KOOP hierbij is te zorgen voor de rails zodat de trein kan rijden.

### **1.1 Uitgangssituatie.**

Het project KOOP (dit project) is vooraf gegaan door het project KOOP - Demo. Het KOOP - Demo project dient als prototype voor dit project, hierdoor kon inzichtelijk worden gemaakt waar de knelpunten waren van het project.

Het KOOP - Demo project is uitgevoerd tussen september 2002 en januari 2003, en deze vond plaats op de "Kennis Satelliet Amersfoort". Aan dit project hebben 4 projectleden deelgenomen, waaronder ikzelf. Daarnaast heeft KOOP nog steeds dezelfde opdrachtgever (ir. P. de Moel), en is de begeleiding vanuit school in beide gevallen ondergebracht bij Ir. H. Plessius.

In de applicatie van KOOP - Demo zijn onderdelen verwerkt die in dit project terugkomen. Deze onderdelen zijn: het aanmaken, muteren en verwijderen van plants (waterzuiveringscompositie) en waters en het uitvoeren van een simulatie

In het KOOP - Demo project zijn bepaalde delen van het project zo goed als vastgelegd, zoals delen van de database structuur en onderdelen waar de applicatie aan moet voldoen.

Hieronder staan enkele beperkingen weergegeven van de KOOP - Demo applicatie.

#### **Simuleren.**

- De simulatie kon alleen worden uitgevoerd met 1 watersample.
- De berekeningen die werden uitgevoerd zijn in het script verwerkt en niet berekend door een reken-server.

#### **Water**

- Er kon maar 1 watersample worden aangemaakt.
- Indien er een nieuw watereigenschap moest worden bijgeplaatst (bijvoorbeeld de parameter Nikkel), moest dit in het script worden aangepast.

#### **Plant**

- De werking van het onderdeel "plant" is naar wens uitgevoerd.

Alle documentatie van het "KOOP - Demo" project zal dienen als uitgangsdokumentatie voor het KOOP project. Dit is niet meegeleverd als bijlage.



## **1.2 Scriptie opbouw.**

Hoofdstuk 2 en 3 hebben betrekking op de organisatie en de beschrijving van het project. Hierbij worden de betrokkenen en de stakeholders besproken. En bij de bedrijving van het project wordt gedacht aan welke eisen het project moet voldoen, welke de succesfactoren er zijn en hoe het project is aangepakt.

Hoofdstuk 4 richt zich op de besluitvorming aangaande het gehele project. Dit houdt in dat er wordt uitgelegd welke technieken en methodes er zijn gebruikt voor de uitvoering van dit project. In hoofdstuk 5 worden de behaalde resultaten besproken.

Tijdens en na het realiseren van het project is het duidelijker wat er verbeterd moet worden. In hoofdstuk 6 wordt er aangegeven welke aanbevelingen er zijn en hoe dit tot stand is gekomen.

In hoofdstuk 7 wordt er in het kort besproken of de doelstelling en de succesfactor van het project zijn behaald.

De evaluatie vindt plaats in hoofdstuk 8, hierbij wordt de mening van de stagiair gegeven over het project. Daarnaast wordt er aangegeven hoe de problemen binnen dit project de volgende keer kunnen worden verbeterd/vermeden.

Aan het eind van de scriptie zijn bijlagen verwerkt die betrekking hebben op dit document. Elke bijlage is een op zichzelf staand document die elk afzonderlijk te lezen zijn. Het gaat hierbij om het "plan van aanpak", het "functioneel ontwerp", het "technisch ontwerp" en het "verdiepingsverslag over portals".

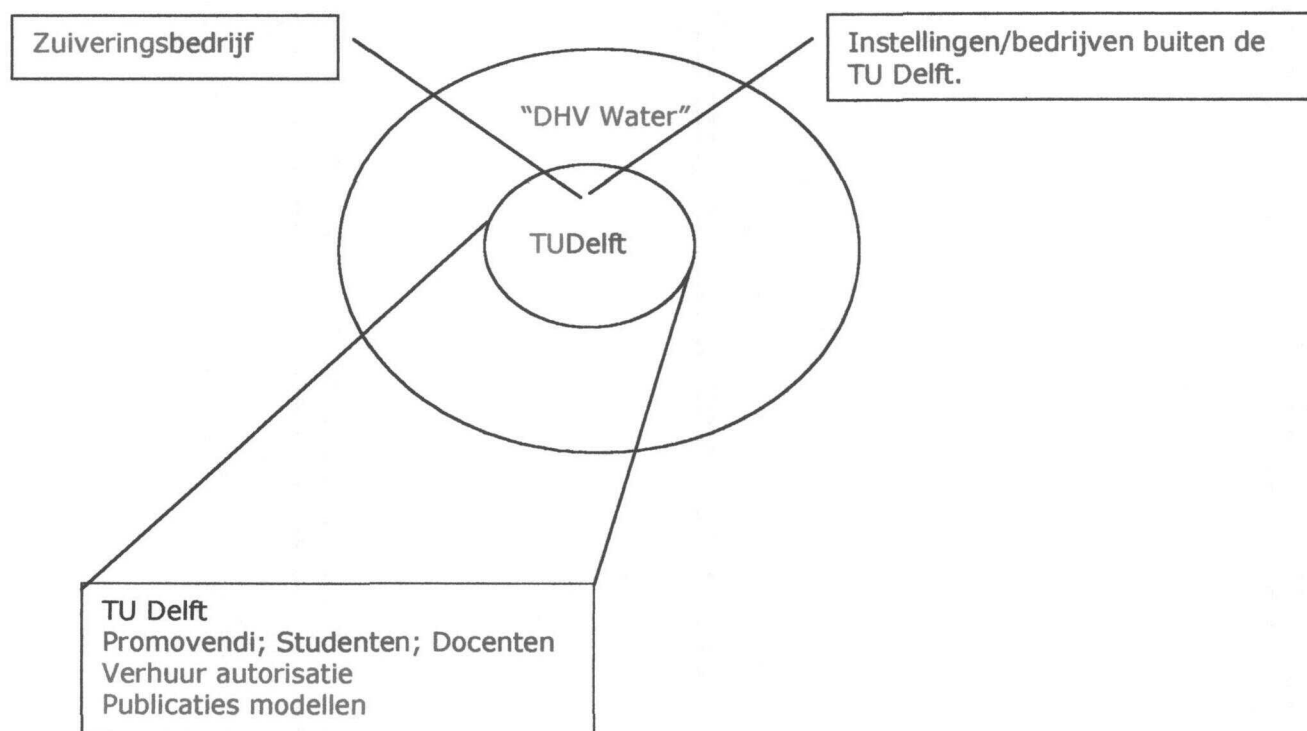
## Hoofdstuk 2. De organisatie

In dit hoofdstuk worden de organisaties besproken die betrekking hebben op het project KOOP. Hierbij wordt bekeken welke organisaties er belang hebben voor het slagen van het project (figuur: Stakeholders) en welke organisaties er betrokken zijn bij het ontwerp van de applicatie (figuur: betrokkenen)

### 2.1 Stakeholders

Er zijn verschillende bedrijven die belang hebben voor het welslagen van het project KOOP en het totale project Stimela. Deze belanghebbende kunnen zich zowel op financieel als op onderzoeksgebied bevinden. Hieronder staat weergegeven wie die belanghebbende (stakeholders) zijn. Uitleg van de twee belangrijkste onderdelen, namelijk TU Delft en DHV, zijn te vinden in paragraaf 2.3 en 2.4.

**Figuur: Stakeholders**

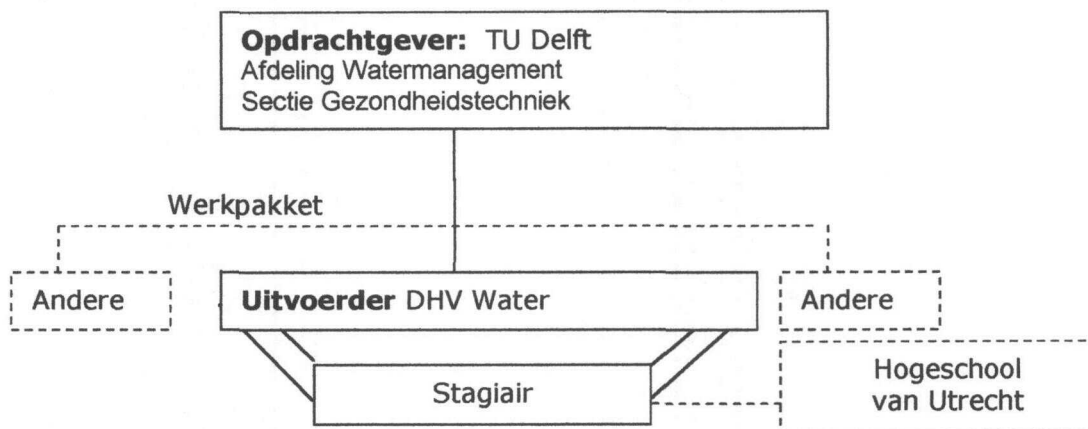


Rood = financieel belang  
Blauw = onderzoeksbelang

## 2.2 Betrokkenen

In het onderstaande figuur is weergegeven welke organisaties er betrokken zijn bij de ontwikkeling van het project KOOP. Deze betrokkenen zijn de opdrachtgever TU Delft en één van de uitvoerenden van de opdrachtnemer, DHV Water, Afdeling PA-ICT. De stagiair is werkzaam onder DHV Water. Buiten het project staande maar wel opgenomen in het diagram is de schoolbegeleider van de Hogeschool van Utrecht. Het onderstaande diagram is verder uitgewerkt in de volgende paragrafen.

**Figuur: Betrokkenen**





### **2.3 Opdrachtgever: TU Delft**

De TU Delft (technische universiteit Delft) is een instelling met 3 core-competenties. Dit zijn:

- Het lesgeven aan studenten;
- Het uitvoeren van onderzoeken;
- Publiceren van bevindingen.

De afdeling waar dit project voor wordt uitgevoerd is Afdeling Watermanagement Sectie Gezondheidstechniek. Deze sectie geeft onderwijs en verricht onderzoek op 3 vakgebieden die de basis vormen voor "de waterketen", te weten:

- Drinkwatervoorziening;
- Riolering;
- Afvalwaterzuiveringen.

De zorg voor een goede waterkwaliteit voor mens en milieu is het belangrijkste aandachtspunt van de gezondheidstechniek. Hiertoe is kennis van chemie, microbiologie, waterzuiveringen, waterwinning en watertransport onmisbaar.

Het project heeft 2 hoofddoelen voor de TU Delft wat betreft onderzoek en onderwijs:

- KOOP als ondersteuning van het lesmateriaal, waarbij studenten spelenderwijs kennis maken met waterzuiveringen;
- Promovendi kunnen bevindingen doorvoeren op modellen of modellen zelf ontwikkelen, aanpassen en toetsen.

Daarnaast heeft de TU Delft financiële belangen voor het welslagen van het project: Dit zijn:

- Het verkopen van toegangsrechten, zodat andere instellingen of bedrijven gebruik kunnen maken van KOOP;
- De bevindingen die zijn uitgevoerd in het programma KOOP publiceren.

De projectleider is Dhr Ir. P.J. de Moel die de opdracht "Stimela" heeft aangenomen. Het project "Stimela" is opgedeeld uit verschillende werkpakketten, waarvan 1 werkpakket project KOOP genaamd. Voor DHV Water is Dhr. P.J. de Moel de opdrachtgever van "KOOP". Hierbij wordt verwacht dat hij "KOOP" alsmede "Stimela" tot een goed resultaat leidt.

## **2.4 Uitvoerder: DHV**

DHV is een internationaal advies- en ingenieursbureau. DHV verleent diensten en biedt duurzame oplossingen aan in de markten:

- Bouw, Industrie en Telecommunicatie
- Luchthavens
- Mobiliteit en Infrastructuur
- Ruimtelijke Ordening en Milieu
- Water

De ruim 3.500 professionals werken vanuit een maatschappelijke betrokkenheid aan innovatieve concepten. Erkende expertise wordt via een hecht kennisnetwerk met sterke lokale aanwezigheid beschikbaar voor klanten in de publieke en private sector.

De diensten omvatten: management consultancy, advisering, ontwerp en engineering, project-, contract- en exploitatiemanagement

### **2.4.1 PA-ICT**

DHV PA-ICT (procesautomatiseringsafdeling en ICT) is de afdeling waar dit project is uitgevoerd. Enkele kerncompetenties van deze afdeling zijn:

- Procesautomatisering
- Strategische advisering
- Internet applicaties
- Advanced Control
- Meet en regel techniek
- Modellerings

Het belang dat DHV heeft voor het welslagen van het project KOOP, is zowel op onderzoek als op financieel gebied. Het financiële onderdeel is omdat DHV opdracht heeft gekregen dit deel van de applicatie te realiseren, en hiervoor betaald krijgt. Op onderzoeksgebied heeft dit bedrijf ook profijt voor het welslagen van het project, omdat dit bedrijf makkelijker zelf onderzoek kan doen naar mogelijke vernieuwingen van zuiveringsinstallaties.

## **2.5 De Afstudeerder**

De stagiair is een 4<sup>e</sup> jaar Information Engineering student. Als voorhistorie heeft deze student het project KOOP - Demo (prototype van KOOP). De reden voor de keuze van dit project is het opdoen van .NET programmeerkennis.

Zijn belang voor het slagen van dit project is, dat dit project betrekking heeft op het afstuderen en het behalen van zijn diploma.

## **2.6 Hogeschool van Utrecht**

De afstudeerder wordt vanuit school begeleid door Dhr. Ir. H. Plessius. Dhr. Plessius was afdelingshoofd en docent op de opleiding Information Engineering. Tijdens deze werkzaamheden is hij begeleider geweest van KOOP - Demo.

## Hoofdstuk 3. Projectomschrijving

### 3.1 Doelstelling

Aanvankelijk opgestelde doelstelling:

Het project heeft als doelstelling het creëren van een simulatieomgeving waarbij potentiële klanten, studenten en promovendi inzicht krijgen in waterzuiveringsprocessen.

Omdat de opzet van de gebruikersinterface parallel loopt met de ontwikkeling van de applicatie, zal hierbij iteratieslagen plaatsvinden. Na het uitvoeren van enkele iteratieslagen is de doelstelling in overeenstemming met de opdrachtgever aangescherpt voor het project KOOP.

Het project KOOP met de stage als kader zal als subdoel krijgen:

Ontwerp van een overzichtelijke gebruikersinterface, waarbij minimaal 1 model kan worden gesimuleerd.

### 3.2 Kwaliteitseisen

Het primaire eindproduct is een werkende webapplicatie, ook wel een "virtueel atelier" genoemd, die minimaal aan de volgende eisen zal voldoen:

- WEB enabled;
- gebruikersauthenticatie met login + gebruikersmanagement;
- interface voor configuratie van:
  - water eigenschappen (input voor het model);
  - plant parameters;
  - model parameters;
- koppeling met rekenmodule Stimela;
- presentatie van resultaten;
- opslag van gebruikersgegevens en resultaten in een database.

Aandachtspunten:

- Duidelijke en overzichtelijke GUI
- Modulaire opbouw
- Flexibiliteit

### 3.3 Succesfactor

*Wanneer is het project succesvol?*

De kwaliteit van de te leveren applicatie is sterk afhankelijk van de in de bovenstaande alinea "Kwaliteitseisen" genoemde voorwaarden die aan de applicatie gesteld worden, en de reeds genoemde doelstelling. De gecreëerde webapplicatie zal in staat zijn om minimaal een zuiveringproces te simuleren. De applicatie zal zodanig worden ontwikkeld dat verdere uitbreiding mogelijk is. Daarvoor moet bij de overdracht van de informatie over de applicatie compleet en overzichtelijk zijn.

### 3.4 Opdracht

De opdracht in precieze bewoording luidt als volgt:

Bouw een internetapplicatie die de volgende functionaliteiten heeft:

- Een inlogprocedure voor de gebruikers;
- Een interface voor de configuratie van water, plant en model parameters;
- Een XML koppeling tussen de webapplicatie en Stimela;
- Het genereren van een simulatierapport over een waterzuivering;
- Output in de vorm van een rapport, eventueel ondersteund door een grafiek;
- De historische gegevens van een simulatie opslaan in de database. Hierdoor kan er snel naar vorige resultaten kan worden teruggezocht;
- Modellen die instelbaar zijn in Statisch, Dynamisch en Stochastisch met elk 3 graden van complexiteit (simpel, normaal en complex).

Figuur: Instelbaarheid modellen

	Statisch	Dynamisch	Stochastisch
Simpel	X	X	X
Normaal	X	X	X
Complex	X	X	X

In het bovenstaande figuur is weergegeven welke type modellen er aanwezig zijn in Stimela. Een statisch, dynamisch en stochastisch model houdt in:

- Statisch (zonder het tijdsaspect): Hierbij wordt 1 watersample door een plant gesimuleerd en het resultaat van het water weergegeven.
- Dynamisch (met het tijdsaspect): Hierbij worden meerder watersamples gesimuleerd. Als resultaat worden meerdere wateren weergegeven.
- Stochastisch (prognose): Door het uitvoeren van zuiveringen treed er vaak vervuiling (in b.v. een zandbed) of slijtage op. Door middel van de stochastische simulatie is het mogelijk deze veranderingen van de zuivering in het resultaat op te nemen en een toekomstige inschatting te maken.

### **3.5 Samenvatting Plan van aanpak**

Het Plan van aanpak bevat de aspecten over hoe de aanpak en de applicatie is opgezet. Enkele kernwoorden hiervan zijn: wat het verleden is, wat het programma moet uitvoeren en op welke wijze het geheel wordt aangepakt.

Als achtergrond van dit project ligt de applicatie KOOP - Demo. De documentatie die hierbij werd opgesteld ligt ten grondslag van dit project.

Het hoofddoel van de applicatie is het toegankelijk maken van verschillende processen voor waterzuivering ter assistentie van het lesmateriaal. Hierbij kan een gebruiker een bepaald water en een plant (waterzuiveringsbedrijf) selecteren, en aflezen wat het resultaat is van het water.

De applicatie aanpak zal worden uitgevoerd volgens de DSDM methode in combinatie met het MoSCoW Principe. Dit houdt in dat alle functionaliteiten zijn opgedeeld in bepaalde graden van belangrijkheid. Indien een bepaald gedeelte is afgerond, zal deze worden gecontroleerd door de opdrachtgever van de TU Delft en de stagebegeleider van DHV.

Na het opzetten van de applicatie kan de gebruiker een plant en een waterzuivering simuleren. De resultaten worden weergegeven op het scherm met de aanduiding of dit voldoet aan de eisen die gesteld worden aan het water.

Voor meer informatie over de aanpak van het project, risicofactoren, documentfasering en applicatiefasering wordt er verwezen naar Bijlage 1 "Plan van aanpak".

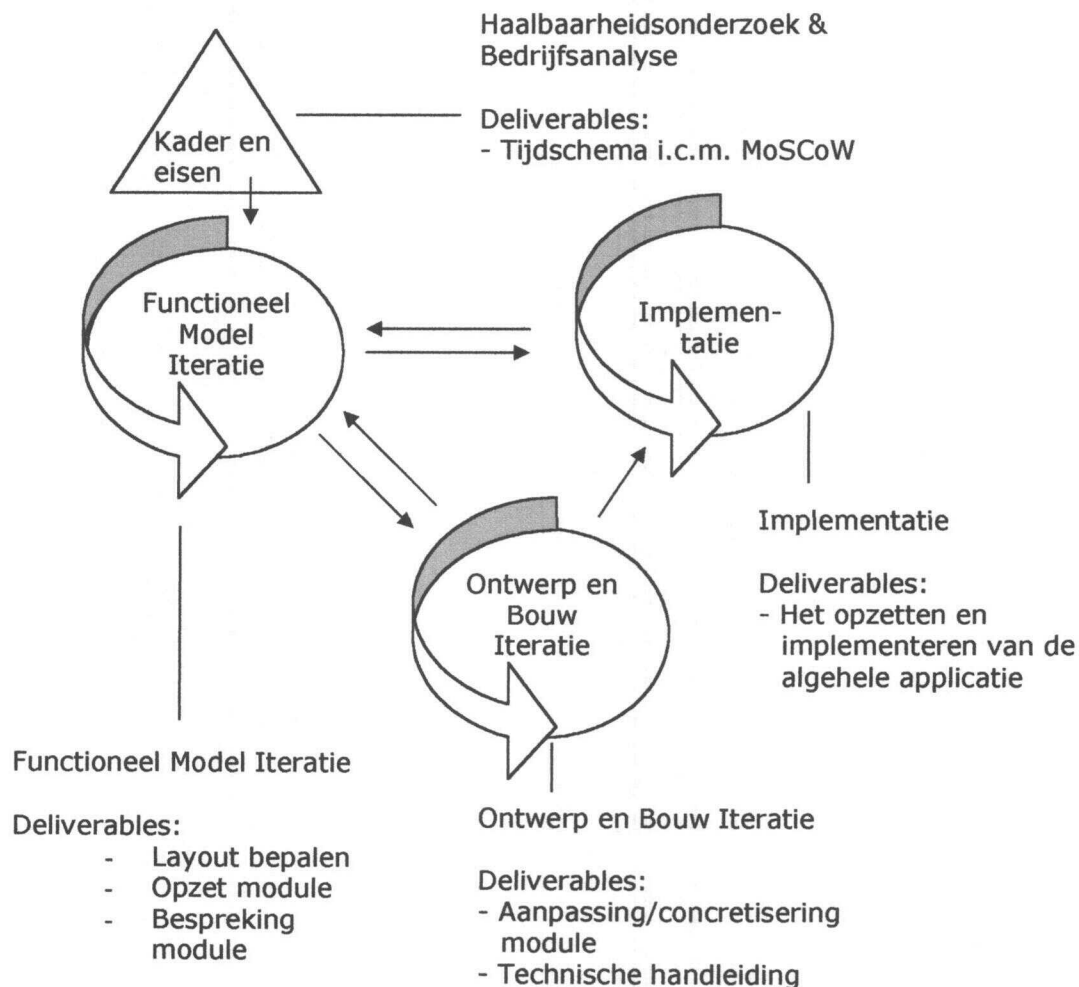
### 3.6 Aanpakmethode

Voor het stageproject zijn bepaalde randvoorwaarden gesteld. Dit is het tijdsbestek waarin het project loopt en de personele bezetting. Hierdoor blijven de functionaliteiten waaraan het product moet voldoen het enige flexibele aspect.

Door deze randvoorwaarden is er gekozen voor een op DSDM (Dynamic Systems Development Method) gebaseerde opzet, met behulp van een incrementplanning. De incrementen worden door middel van het MoSCoW principe opgesteld ("must have", "should have", "could have" en "would have").

Hoe het MoSCoW principe is toegepast in dit project wordt nader uitgelegd in het functioneel ontwerp.

Figuur: Opzet DSDM methode





### 3.6.1. Functionele Model Iteratie

In deze fase wordt geïdentificeerd wat gedaan moet worden in de betreffende incrementen (een increment is een opdeling van de functionaliteiten). Deze incrementen worden opgesteld met behulp van het MoSCoW principe. Deze keuzes worden aan de hand van de documenten uit de bedrijfsanalyse gemaakt en besproken met de opdrachtgever.

In deze fase moet het duidelijk worden wat er moet worden uitgevoerd in elk betreffend increment, dit is vermeld in het ontwerpdocument. Hoe de modules worden opgesteld staat vermeld in de technische documentatie.

### 3.6.1 *Ontwerp & Bouw Iteratie*

Na het definiëren van het increment wordt deze uitgevoerd tot een werkend prototype. Indien er aanpassingen zijn wordt dit verwerkt in het technisch en het functioneel ontwerp, onder het hoofdstuk "Aanpassingen". Hierbij zal het project weer terugvallen naar de Functionele Model Iteratie.

### 3.6.2 *Implementatie*

De implementatiefase zal enkele keren worden binnen dit project worden uitgevoerd. De opdrachtgever kan hierbij gemakkelijk een beeld vormen over de voortgang en functionaliteiten van KOOP.

### 3.6.3 *Documentatie*

Het functioneel ontwerp bevat een "Beschrijving Bedrijfsgebied" (BB), dit houdt in dat er wordt gedefinieerd welke gebruikers gebruik maken van de applicatie; wat de gebruikers ermee willen doen; en wat krijgen de gebruikers te zien. Ook is er een weergave gemaakt van de sitemap van de applicatie.

Het technisch ontwerp bevat alle technische aspecten van de applicatie KOOP. Hieronder valt de "Entity Relationship Diagram" (ERD). Dit bevat de structuur van de database met onderlinge relaties. In het technisch ontwerp wordt ook de "Beschrijving Systeemarchitectuur" (BS) nader beschreven. Hierin wordt het platform en opbouw van de applicatie gedefinieerd.

## 3.7 **Het haalbaarheidsonderzoek en bedrijfsanalyse**

De haalbaarheid van het project is in het plan van aanpak nader gedefinieerd. Door de uitvoering van het "KOOP - Demo" project is de technische haalbaarheid bewezen. De enige echte onzekerheden komen door het gebruik van de vrij nieuwe programmeertaal .NET, en het tijds kader waarin het moet worden gerealiseerd. Om te zorgen dat de kwaliteit van "KOOP" gewaarborgd blijft, is in het plan van aanpak een planning opgesteld aan de hand van het MoSCoW principe.

Door het onzekerheidsaspect dat veroorzaakt wordt door het tijds pad is er besloten voor een DSDM methode. Normaal wordt DSDM gebaseerd op het raamwerk voor Rapid Application Development (RAD). Dit houdt in dat het MoSCoW principe wordt toegepast binnen een bepaald tijdsbestek wat niet aanpasbaar is. In dit project wordt het MoSCoW binnen het DSDM principe als geheel toegepast. Hiervoor is gekozen omdat de prioriteit van enkele functionaliteiten hoger zijn dan het tijdsbestek waarop het wordt gecreëerd.

## **Hoofdstuk 4. Projectbespreking**

In dit hoofdstuk worden de keuzes en ontwikkelingen van Project KOOP omschreven. De te bespreken paragrafen gaan zowel over de techniek alsmede de methode van het project.

### *Produkt*

- 4.1: Architectuur: In deze paragraaf komt de structuur van de software en programmering aan de orde.
- 4.2: Implementatie aanpak: In deze paragraaf is de hardwarematige opzet weergegeven en uitgelegd
- 4.3: Programmering: In deze paragraaf gaat het om de vraag welke programmeertaal er gebruikt kunnen worden. Daarnaast is gekeken welke portal het best te gebruiken is voor het KOOP Project en Stimela. Hierbij zijn verschillende mogelijkheden naast elkaar gezet en met elkaar vergeleken.
- 4.4: Database: De opzet van de database heeft al een voortraject doorlopen. Hierdoor zijn de grote lijnen van de database al zichtbaar geworden. Door de sturing van de opdrachtgever en door de aanbevelingen, beschreven bij KOOP – Demo, is er een nieuwe architectuur opgezet.
- 4.5: Functies: In deze paragraaf wordt besproken wat er is gerealiseerd in elke prototype.

### *Proces*

- 4.6: Projectverloop: In dit hoofdstuk wordt in vogelvlucht de doorloop van het project doorgenomen.
- 4.7: Bespreking evaluatie met opdrachtgever: Bij het maken van de prototypes heeft er een terugkoppeling plaatsgevonden met de opdrachtgever. Deze terugkoppelingen zijn verwerkt in deze paragraaf.



## 4.1 Architectuur

### 4.1.1 Architectuur

1)

De applicatie is opgebouwd volgens de thin-cliënt architectuur. Dit houdt in dat de gebruiker niets hoeft te installeren voor het gebruik van de applicatie, maar via de web browser (de GUI) toegankelijk is. De reden voor deze keuze is om de drempel tot gebruik zo laag mogelijk te maken.

2)

In de laag van de .NET Server (webserver) bevindt zich de applicatielaag. Hier staan alle regels in die betrekking hebben voor de uitvoering van de applicatie (wanneer, hoe, waar, en wat er gedaan moet worden.) De applicatie zorgt daarnaast voor de communicatie naar de andere onderdelen. Daarnaast bevindt zich op de webserver de database. Hierin worden de, door de gebruiker aangemaakte, waters, plants en simulaties opgeslagen. De reden waarom de applicatie en de database op dezelfde server staan, komt omdat alles wat met "Stimela" te maken heeft gecentraliseerd is op één server. Indien later blijkt dat het beter is om dit toch nog op te delen is dit gemakkelijk aan te passen in het script, door op de server de adresverwijzing aan te passen.

In dit onderdeel worden de wensen van de gebruiker opgevangen door de applicatie in HTML formaat. Deze zal de wensen van de klant uitvoeren, bijvoorbeeld de uitvoering van een simulatie. Hierbij wordt de betreffende data uit de database van de server gehaald, door middel van SQL en omgezet naar XML formaat. Na het converteren naar XML worden deze gegevens via het http formaat verstuurd naar de Stimela server (de rekenmodule kan zich op elke plaats op het Internet bevinden. In deze versie wordt de rekenmodule op de Stimela server gebruikt).

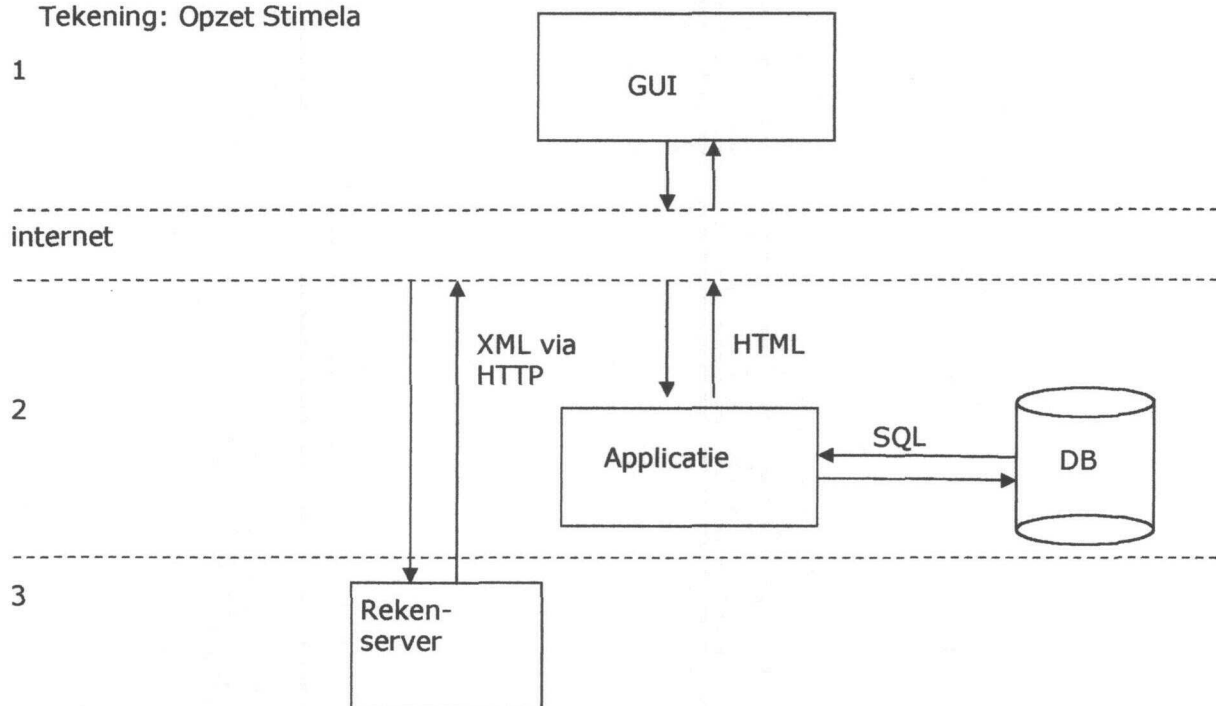
3)

Stimela gaat met deze data rekenen (hoe de berekeningen plaats vinden valt buiten de scope). Na het uitvoeren van de berekening zal de server de uitkomst retourneren.

2)

De data zullen worden opgeslagen in de database van de applicatie en de gebruiker wordt op de hoogte gesteld van de berekende resultaten.

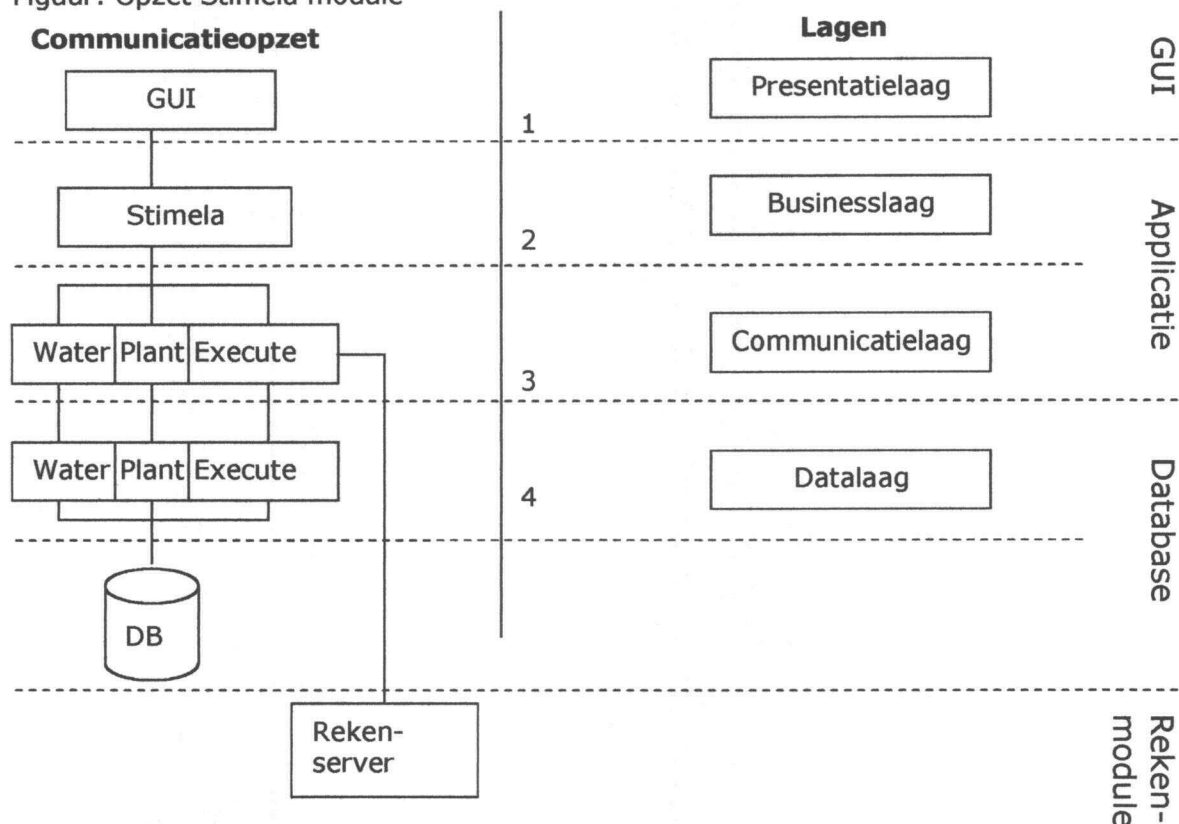
Tekening: Opzet Stimela



#### 4.1.2 Opzet programmering

De programmering van de applicatie is zo opgezet dat deze uit één totale module bestaat, namelijk de "Stimela" module. Deze module zorgt voor het opslaan, muteren en verwijderen van plant en water gegevens en het uitvoeren van waterzuiveringssimulaties.

Figuur: Opzet Stimela module



De opzet van de module Stimela is gelijk aan de opzet van de portal en bestaat uit 4 lagen (zie bovenstaande tekening).

##### 1) Presentatielaag

Als eerste is de presentatielaag met daarin de GUI (Graphical User Interface), dit is wat de gebruiker gepresenteerd krijgt. Indien de gebruiker een actie uitvoert in de GUI wordt dit doorgegeven aan de Businesslaag.

##### 2) Businesslaag

De Businesslaag bepaald wat de acties zijn van het programma. Een voorbeeld van een actie kan zijn: het veranderen van de GUI of een aanvraag doen voor het uitlezen van de database.

##### 3) Communicatielaag

Deze communicatielaag bestaat uit 3 classes, namelijk de water, plant en execute class. Indien de communicatie te maken heeft met het aanmaken; aanpassen of verwijderen van een water/plant wordt de class "Waters"/"Plants" aangesproken met daarin bepaalde functies. Indien er een simulatie wordt uitgevoerd, wordt de class "Execute" aangesproken. Deze zorgt dan naast de communicatie naar de datalaag ook voor de communicatie met de rekenserver.

##### 4) Datalaag

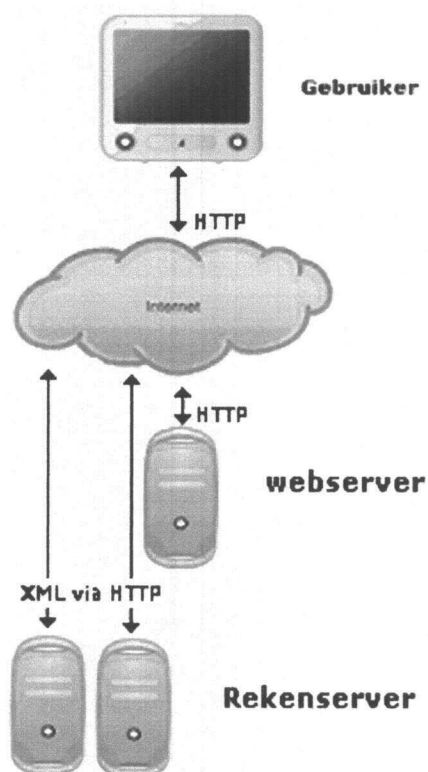
Voor toepassingen op de database wordt er een procedure benaderd.

Deze procedure kan het invoeren, uitlezen, aanpassen en verwijderen uit de database bevatten. De resultaten worden weer teruggegeven aan de communicatielaag.

## 4.2 Implementatie aanpak

Elke willekeurige gebruiker kan via het Internet op de Stimela site komen door middel van een webbrowser. Via het Internet heeft deze gebruiker contact met de webserver. De applicatie op de webserver is het centrale contactpunt van de gehele applicatie en elk onderdeel van de applicatie zal alleen hiermee communiceren. Op de webserver staat naast de applicatie ook de database. Aanvragen van de gebruiker gaan via de applicatie naar de database en zullen ook op deze wijze worden geretourneerd. Daarnaast zorgt de applicatie op de webserver voor de communicatie met de rekenmodules. Dit zal eveneens plaatsvinden via het Internet volgens XML formaat.

Figuur: Opzet Stimela implementatie



### 4.3 Programmering

Bij het maken van de applicatie "Stimela" moeten verschillende functionaliteiten gemaakt worden. Veel van deze functies (bijvoorbeeld inlogprocedure, rechten aan onderdelen en onderhoud, etc.) zijn veelvoorkomend en hierdoor de basis functionaliteiten van een website.

Deze basis functionaliteiten kunnen zelf gemaakt worden, en dan vanaf het beginpunt worden opgebouwd of zijn voorgeprogrammeerd in een portal. Voor het maken van deze keuze is eerst op een rij gezet wat de voor- en nadelen zijn van het gebruik van een portal ten opzichte van een programmeeromgeving. Deze zijn hieronder uitgewerkt.

#### Voordelen portal t.o.v. zelf programmering

- De programmering van standaard functionaliteiten zijn al uitgevoerd;
- Bij zelfprogrammering moet het wiel opnieuw worden uitgevonden;
- Gratis (in veel gevallen);
- Veel online hulp d.m.v. forums;
- Verminderd de programmeertijd.

#### Nadelen portal t.o.v. zelf programmering

- Type database is zelf te kiezen bij zelfprogrammering;
- Architectuur is zelf op te zetten bij zelfprogrammering;
- Communicatiewijze is zelf te ontwerpen (bijvoorbeeld de keuze tussen thin of fat cliënt) bij zelfprogrammering;
- De GUI is makkelijk te wijzigen en aan te passen bij zelfprogrammering;
- De maker zit niet vast aan de voorwaarden van de portal bij zelfprogrammering.

#### 4.3.1 Portals

In het algemeen worden de volgende eigenschappen toegeschreven aan een portal: Het betreft een startpagina, in de vorm van een unieke URL voor elke gebruiker binnen een organisatie, waarin verschillende functies/mogelijkheden zijn opgenomen.

In de vorige paragraaf is een overzicht gegeven waar men rekening mee moet houden bij de keuze van een portal ten opzichte van zelf programmering van basis functionaliteiten. De eerste indruk zal zijn dat er teveel nadelen aan verbonden zijn.

#### Hoe is dit opgelost.

Als eerst zijn de eisen van KOOP genoteerd, dus:

- Welke database er gebruikt wordt;
- Hoe de communicatiewijze plaats moet vinden;
- Welke functionaliteiten in KOOP moeten komen;
- Hoe wordt de portal gepresenteerd aan de gebruiker.

Daarna zijn er een 12 - tal portals onderzocht en vergeleken met de eisen van KOOP (zie bijlage 4: verslag Portals). Hierdoor wordt de portal gekozen die het dichtst bij de wensen van de opdrachtgever ligt, en het scheelt veel tijd in programmeer- en uitzoekwerk.

Uit het onderzoek was gebleken dat Dotnetnuke en Dotnetnuke XXL het beste uit de test kwamen voor het project KOOP. De Dotnetnuke en Dotnetnuke XXL zijn daarna met elkaar vergeleken (via reacties op forums). Hieruit bleek dat Dotnetnuke XXL meer functionaliteiten bezat maar daarnaast een stuk minder stabiel was.

Om deze reden is er tenslotte gekozen voor de meest stabiele omgeving, Dotnetnuke.



#### 4.3.2 Programmering Stimela

In de vorige paragraaf is besproken dat er gekozen is voor een portal. Een portal bevat veel basis functionaliteiten maar niet alle functies die betrekking hebben op KOOP. Deze moeten zelf geprogrammeerd worden.

Bij de keuze van een programmeertaal wordt met verschillende aspecten rekening gehouden. Deze zijn o.a.:

- De kennis van de stagiair en/of programmeur;
- De ondersteuning van de cliënt omgeving (browsers);
- Ondersteunende hulpmiddelen door b.v. forum, helpfiles, ontwerpomgeving, e.d.;
- Psychologisch aspect van de programmeur (de eigen mening van de programmeur over de programmeertaal);
- Keuze van de opdrachtgever;
- Keuze van het operation system.

De keuze van de programmeertaal stond aan het begin van het project al vast. Deze keuze was het gebruik van .NET. De reden hiervoor was de geïntegreerde XML koppeling. Daarnaast kan de gebruiker gebruik maken van zijn eigen keuze programmeertaal, zoals Visual Basic, Java en C. Aangezien de meeste ervaring bij Visual Basic.Net ligt is hiervoor gekozen.

Wel zijn er nadelen verbonden aan deze programmeertaal, omdat het prototype gemaakt is in ASP – Visual Basic script. Dit is zodanig verschillend van opzet, dat dit niet kan worden hergebruikt.

Een tweede nadeel is dat de kennis van ASP.Net minder is dan de kennis van ASP. Hierdoor kan de doorlooptijd voor de realisatie vergroten.

Naast de keuze van een programmeertaal kan er ook nog een keuze gemaakt worden voor een ontwikkelomgeving. De keuze van de programmeeromgeving is Visual Studio.NET. Er is hier geen onderzoek naar gedaan.

Dit heeft een drietal redenen:

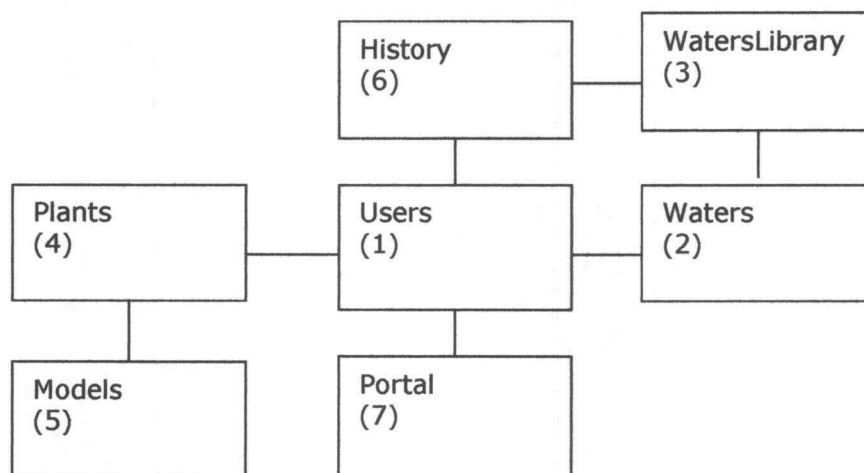
- De programmeertaal was al vastgelegd;
- DHV had Visual Studio.NET al aangeschaft;
- De portal is geprogrammeerd in Visual Studio.NET waardoor het makkelijker was hierop door te gaan.

#### 4.4 Database

De totale website heeft een database die is op te splitsen in 2 onderdelen. Dit zijn de onderdelen voor Dotnetnuke (de portal), en de onderdelen voor het onderdeel "Stimela". In het onderdeel van Dotnetnuke worden de standaard gegevens opgeslagen, zoals gebruikers, inhoud van de site, etc. Aan Dotnetnuke is verder niets veranderd, zodat de module Stimela ook werkt bij andere Dotnetnuke installaties. In het onderdeel van "Stimela" worden alle data, die betrekking hebben op een waterzuiveringssimulaties, opgeslagen.

De database is opgezet in SQL Server, deze keuze was al gemaakt voordat de stagiair aan de opdracht was begonnen. Hierbij is als reden gegeven dat SQL Server de reputatie heeft als meest betrouwbare server onder Windows, en deze wordt vaak gebruikt als database voor veel gebruikers (tot 32.000 gebruikers). Daarnaast is er ook veel kennis van SQL Server binnen DHV aanwezig.

De uiteindelijke databaseopzet is onderverdeeld in 7 verschillende aandachtsgebieden, waarbij elk aandachtsgebied zijn eigen tabellen bevat. In de volgende paragrafen wordt deze databaseopzet vergeleken met de database van KOOP – Demo en de eerste opzet van KOOP. In paragraaf 4.4.3 wordt de uiteindelijke KOOP database besproken.

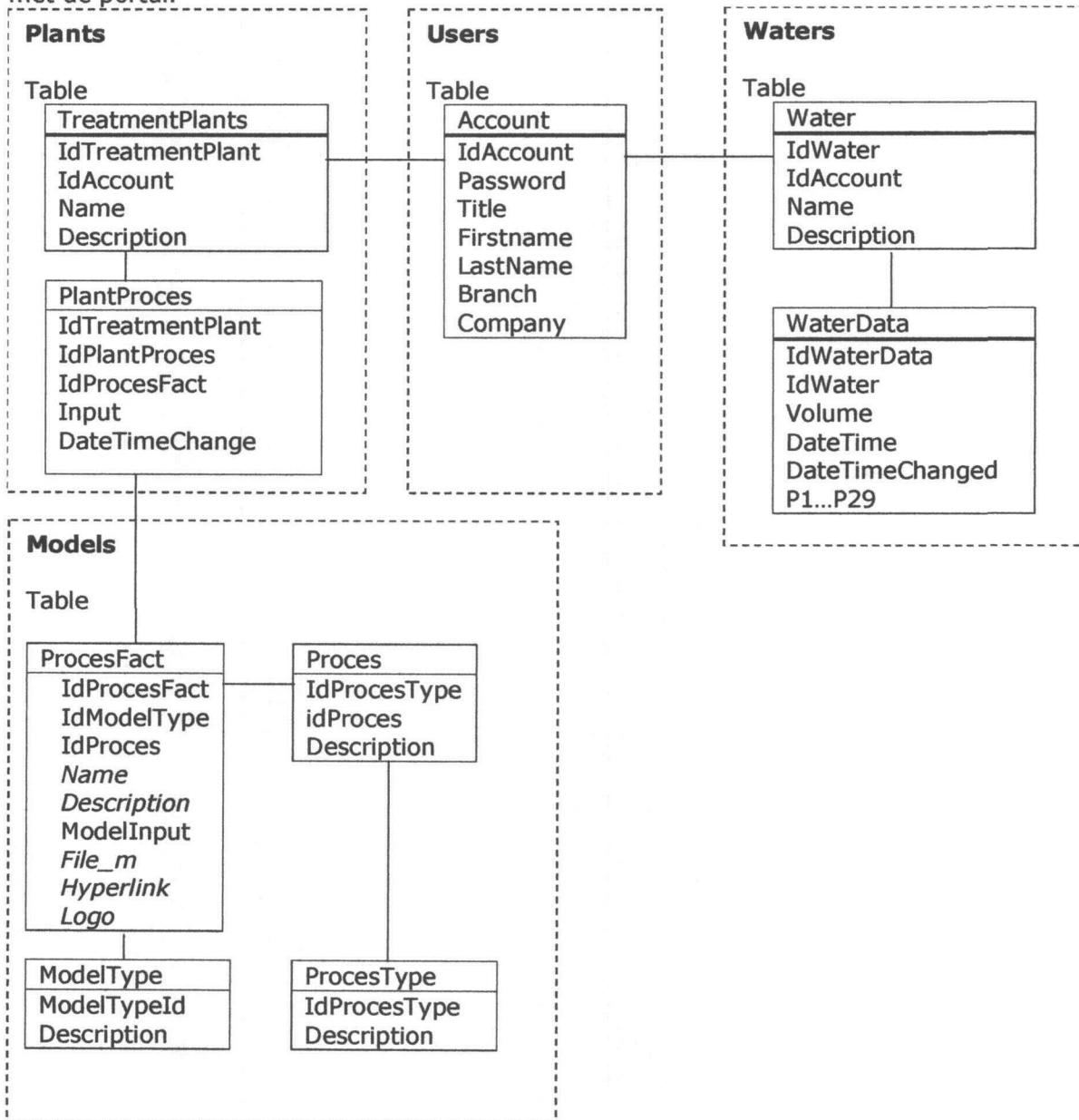


1. Users: Bijna alle opgeslagen data in de database is gekoppeld onder de naam van de gebruiker. De naam en andere persoonsgegevens staan opgeslagen in de tabel "Users". Dit maakt de tabel "Users" als centrale database waaruit ook de portal wordt aangestuurd.
2. Waters: In de tabellen van "Waters" staan alle gegevens van water opgeslagen.
3. WatersLibrary: Alle gegevens zoals de eenheid, default-waarden en beschrijvingen, die betrekking hebben op watereigenschappen, zijn opgeslagen in de tabellen van "WatersLibrary".
4. Plants: Naast dat de gebruiker "Waters" kan aanmaken, kan deze ook een plant samenstellen. Deze plant zal bestaan uit verschillende processtappen. De plant met de processen wordt opgeslagen in dit aandachtsgebied.
5. Models: Een proces wordt uitgevoerd volgens een bepaald model dat bij het proces hoort, deze modellen worden onder het aandachtsgebied "Models" opgeslagen.
6. History: In het "History" onderdeel wordt gesimuleerde data (Water en Plant) opgeslagen.
7. Portal: Dit is het onderdeel van de database die betrekking heeft op de Portal Dotnetnuke.

#### 4.4.1 KOOP - Demo Database opzet

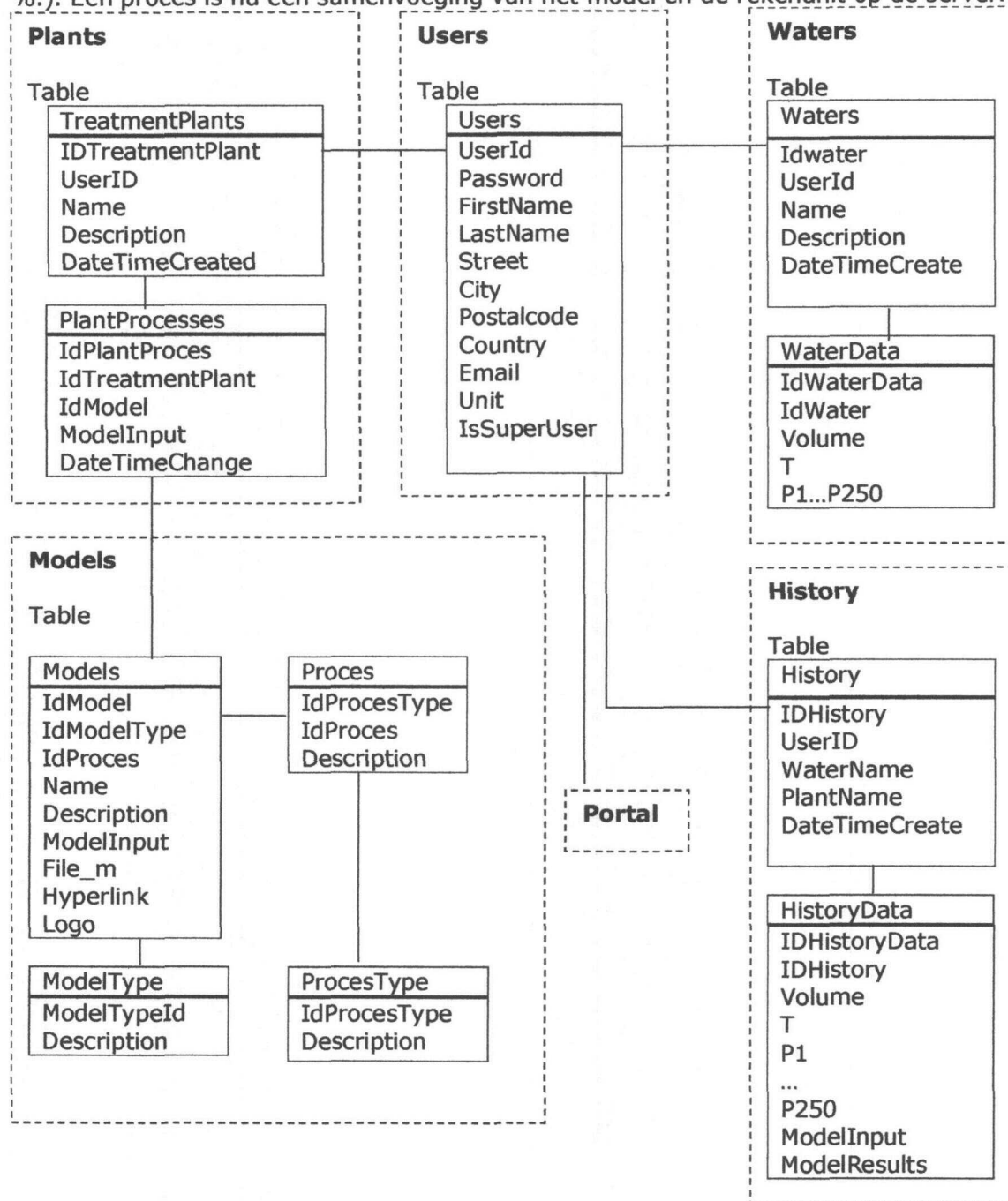
De database had al gestalte voordat er aan het project KOOP was begonnen. Deze database was opgezet in MS Access 2000 voor het project KOOP - Demo. Hieronder is een weergave gemaakt van de opzet van KOOP - Demo. De database is hierbij opgedeeld in de hiervoor genoemde aandachtsgebieden. Zoals nu duidelijk zichtbaar is, zijn niet alle aandachtsgebieden aanwezig in deze opzet. In deze tabellen is alleen de mogelijkheid gecreëerd voor het opstellen van water en plant, waarna er een simulatie kan worden uitgevoerd.

De eerste aanpassingen aan deze onderstaande database zijn: zorgen voor history tabellen waar de simulatie kan worden opgeslagen, en het zorgen voor een koppeling met de portal.



#### 4.4.2 Opzet KOOP Database

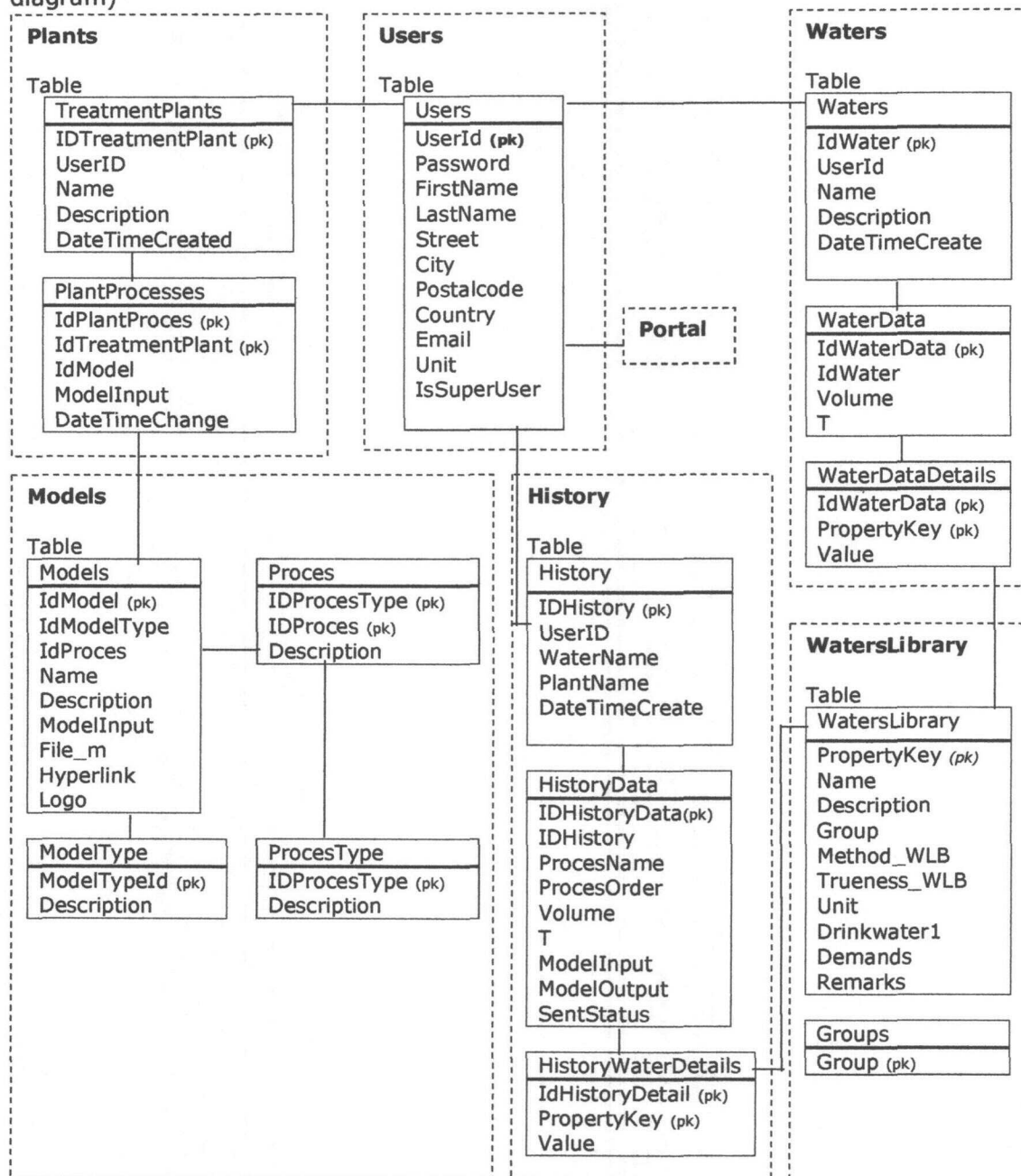
Als eerste is de database van KOOP - Demo gekoppeld aan de Dotnetnuke database. Hierdoor verandert de opzet van de Users-tabel van de Users-tabel van Dotnetnuke. Door deze verandering verandert ook de koppeling naar de andere tabellen (van AccountId → UserId). Daarnaast is er in tabel "Treatmentplant" en "waters" de DateTimeCreated bijgeplaatst in verband met onderhoud (b.v. alles ouder dan 1 jaar verwijderen). Ook is er in deze fase een "History" tabel ontworpen, waar alle simulatieresultaten van de gebruikers wordt opgeslagen. De tabel "ProcesFact" is hernoemd naar de tabel "Models". De benaming ProcesFact staat voor Proces Details, dus dat hier alles over het proces te vinden is. Dit is echter niet zo, in deze tabel wordt weergegeven hoe het proces werkt (bijvoorbeeld: hoeveelheid lood vermindert met 10 %). Een proces is nu een samenvoeging van het model en de rekenunit op de server.





#### 4.4.3 Uiteindelijke database opzet KOOP

Tijdens het opzetten van de vorige databaseopzet, zijn er beperkingen geconstateerd voor de eventuele uitbreiding van WaterProperties (P1 tot P250 van tabel HistoryData & WaterData). Daarnaast was het niet mogelijk via die opstelling eigenschappen aan WaterProperties te koppelen, zoals de eenheid, default-waarden, naam en dergelijke). Dit is opgelost door het bijplaatsen van de WatersLibrary (geselecteerd in onderstaand diagram)



## 4.5 Bespreking functies

Het project is onderverdeeld in functies, deze functies hebben elk hun eigen prioriteiten gebaseerd op het MoSCoW principe.

De knelpunten waar nog aan gewerkt moeten worden zijn:

- Het simuleren van meerdere watersamples binnen 1 simulatie
- Kleurweergave van water bij het (niet) voldoen aan de wettelijke eisen.
- Het ombouwen van de User Interface van prototype 2 naar prototype 3.

Hieronder staat weergegeven wat er is uitgevoerd bij elk prototype. Hierbij zal opvallen dat er bij prototype 1 meer is uitgevoerd dan bij de andere prototypes. Dit komt door de iteratie slagen die gemaakt zijn aan het einde van elk prototype evaluatie.

Uitvoering	Portal	Prototype 1	Prototype 2	Prototype 3
------------	--------	-------------	-------------	-------------

### MUST HAVE

Opzetten Login Procedure	X			
Opzetten Register Procedure	X			
Opstellen van een nieuw model	-	X	X	X
Opslaan van Modelparameters in XML formaat	-	X	X	X
Opstellen van een nieuwe plantcompositie	-	X	X	X
Opstellen van een nieuw water	-	X		
Opstellen van een simulatie	-	X	X	X
Per model versturen naar Stimela	-	X	X	X
Presenteren van de resultaten van de simulatie	-	X	X	X

### SHOULD HAVE

Weergeven modelparameters	-	X	X	X
Lijst van beschikbare modellen	-	X	X	X
Aanpassen van beschikbare modellen	-			
Weergeven plantcompositie	-	X	X	X
Lijst van beschikbare plantcomposities	-	X	X	X
Aanpassen van beschikbare plantcomposities	-	X	-	-
Weergeven waterproperties	-	X	-	-
Lijst van beschikbare watertypes	-	X	X	X
Aanpassen van beschikbare watertypes	-	X	-	-
Modellen plants en waters verwijderen	-	X	X	-
Gebruikers verwijderen	X			
Gebruikers autorisatie	X			
Inleidende tekst homepage	-	-	-	-

**COULD HAVE**

Online aanpassen van de homepage	X			
Online aanpassen plant, model en waterpage	X			
Verwijderen van modellen plants en waters	-	Alleen plants en waters	Alleen plants en waters	Alleen plants en waters
Historie verwijderen door beheerder	-	-	-	-
Onderhoud aan de site	X			
Weergeven gebruikershistorie	-	vervallen	vervallen	vervallen
Weergeven van historie resultaten	-	vervallen	vervallen	vervallen

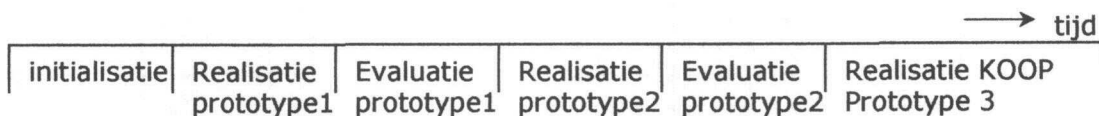
**WOULD HAVE**

Foutcontroles	X	-	-	-
Veranderen van modelparameters	-	X	-	-
Verwijderen van historische gegevens door gebruiker	-	-	(automatisch)	(automatisch)
Link van login naar registreren	X			
Encrypted password	-	-	-	-
Melding als registreren gelukt is	X			
Gebruikersvoorwaarden	X			
Model plant en water introtekst	-	-	-	-



#### 4.6 Bespreking projectdoorloop

In dit hoofdstuk wordt uitgelegd op welke wijze het project is doorlopen. Hierbij wordt elk aspect in het kort toegelicht.



##### *Initialisatie*

Voordat er een operationele opzet is gemaakt van de applicatie, is de documentatie (ontwerpdokument en gebruikershandleiding) van KOOP - Demo aandachtig doorgenomen. Na dit ontwerp is kritisch gekeken naar mogelijke punten die voor verbetering vatbaar zijn. Daarnaast zijn er in overleg met de opdrachtgever extra functionaliteiten toegevoegd.

Na deze fase waren alleen de wensen die er in KOOP moest komen duidelijk gemaakt, en niet hoe het eruit moest zien. Toen het duidelijk was dat er gebruik werd gemaakt van een portal, is er een opzet gemaakt van een combinatie tussen de portal en KOOP - Demo. Deze opzet is verwerkt in het functioneel ontwerp (zie bijlage 3)

Het opgestelde document is verstuurd naar de opdrachtgever, waarna er direct is begonnen met het ontwerpen van het prototype. Aangezien de stagiair nog niet bekend was met de programmeertaal .Net, en de opzet van de portal (DotNetNuke) was hier ruim de tijd voor gepland.

##### *Realisatie /evaluatie prototype 1 (6 november 2003)*

Na het realiseren van het prototype vond er terugkoppeling plaats naar de opdrachtgever. Hierbij is veel commentaar gegeven over de opzet van de GUI (de verdieping is te vinden in hoofdstuk 4.7. Resultaat, commentaar, acties en aanbevelingen.)

##### *Realisatie /evaluatie prototype 2 (4 december 2003)*

Door de verduidelijking van de aandachtsgebieden is het resultaat van het tweede prototype beter beoordeeld door de opdrachtgever. Daarnaast had de opdrachtgever nieuwe ideeën voor verbetering van de GUI.

##### *Realisatie KOOP - prototype 3 (tot eind januari 2004)*

De uiteindelijke realisatie zal plaatsvinden na het inleveren van de scriptie.

#### 4.7 Resultaat, commentaar, acties en aanbeveling van de prototype

Zoals eerder aangegeven is dit project volgens de DSDM methode uitgevoerd. Dit houdt in dat eerst het kader en de eisen zijn opgesteld, waarna het ontwerp van het prototype wordt gemaakt. Na het prototype worden deze besproken en vindt er terugkoppeling plaats.

Prototype 1, Evaluatie	Prototype 2, Evaluatie	Eind evaluatie
6 november 2003	4 december 2003	Eind januari 2004

##### *Prototype 1, evaluatie*

###### **Gerealiseerd:**

- Aanmaken muteren en verwijderen van waters en plants.
- Uitvoering van de simulatie voor 1 watersample, waarbij de technische opzet zo was gemaakt dat elke watersample afzonderlijk verstuurd werd.

De GUI van prototype 1 had een "Wizard" opzet, dus veel kleine schermpjes die doorlopen moest worden voor het maken van een plant of water.

###### **Commentaar:**

Dhr. Ir. P. de Moel heeft het prototype nagekeken en hierop reactie gegeven. Zijn commentaar was dat er teveel werk was besteed aan het aanmaken en muteren van water/plant gegevens, terwijl de nadruk ligt bij het simuleren. Tevens was de walkthrough door de schermen niet naar zijn wens (te onoverzichtelijk). Als reactie hierop gaf hij aan dezelfde presentatie te krijgen als bij "KOOP - Demo". Daarnaast was de gedachtegang van het afzonderlijk versturen van watersamples niet juist, maar moet die in 1 blok worden verstuurd naar het proces.

###### **Actie:**

Na de reactie van de opdrachtgever zijn er acties ondernomen voor uitvoering en documentatie.

###### **Actie 1**

Om te zorgen dat de GUI voor de opdrachtgever naar wens is, is er besloten de site online te zetten waardoor de opdrachtgever direct terugkoppeling kan geven indien er wat fout gaat.

###### **Actie 2**

Er is een nieuwe lay-out ontworpen voor "Stimela". Deze lay-out is kortgesloten met de opdrachtgever en verwerkt in de technische en functionele documentatie onder het hoofdstuk aanpassingen.

###### **Actie 3**

Afspraak gemaakt voor de presentatie van prototype 2. Hierbij was de doorlooptijd een stuk korter dan bij prototype 1, dit had verschillende redenen:

- De stagiair had de programmeertaal .NET geleerd;
- De werking van de portal was bekend;
- Er hoefde minder documentatie gemaakt te worden (alleen aanvullingen);
- Snellere terugkoppeling, dus eerder de mogelijkheid tot aanpassingen.

###### **Actie 4**

De gehele opzet van de site moest worden aangepast en omdat de programmeertaal beter bekend was bij de stagiair, is besloten de site opnieuw te maken. Hierdoor was de site sneller opgezet en beter geprogrammeerd (structuur van de programmering en de functies verbeterd).

## Prototype 2, Evaluatie Gerealiseerd:

Hieronder is een grafische weergave gemaakt van de GUI van prototype 2.

Home · koop demo  
december 11, 2003

### Stimela

**Selecteren, muteren en aanmaken van water en plant**

**Select water**  
1 Water ▾  
Details  
Add New

↔

**Select Plant**  
1 Model ▾  
Details  
Add New

→

**Simuleren**  
Run

**Hieronder kan per proces de watersampel input & output, model input & output bekenen worden**

**Processes** IronCoagulationFit2 ▾ Details

**De resultaten van de simulatie worden hieronder weergegeven**

Output Water			
Name Property	Result	Demands	Unit
Cryptosporidium	20	10	
Escherichia coli	20	20	/100 ml
Enterococci	20	10	/100 ml
Enteroviruses	20	10	

= Tekst omringt met dit kader is als comment toegevoegd aan deze GUI

Hieronder volgt een korte omschrijving van de functionaliteiten:

- Bij het selecteren van een water of plant is het mogelijk deze te muteren of een nieuwe water/plant toe te voegen aan de lijst;
- Indien er een water en een plant is geselecteerd kan de gebruiker op "Run" drukken. Hierbij wordt de simulatie uitgevoerd, en komt het onderste deel (het deel onder de "Run" knop tevoorschijn);
- In de lijst van "Processes" (zie tekening), worden de processen weergegeven in volgorde van simulatie. Deze processen kunnen elk bekeken worden op:
  - Wat de water input is;
  - Wat de water output is (en of dit voldoet aan de wettelijke bepalingen (Demands));
  - Wat de eigenschappen zijn van het model;
  - Wat de teruggekregen waarde is van het model.
- Als onderste tabel staat de hele reeks met uitput waarden van de plant weergegeven.



**Commentaar:**

De reactie op de weergave van de site was positief. De presentatie van de modellen afzonderlijk vond Dhr. P. de Moel een leuk initiatief. Zijn commentaar was wel dat het meer op een spoorrails moest lijken.

**Actie:**

De acties die zijn ondernomen door de terugkoppeling van de opdrachtgever zijn:

**Actie 1**

Het opzetten van een nieuwe GUI. Deze GUI is tijdens de bespreking met de opdrachtgever opgesteld en heeft geresulteerd in de onderstaande tekening.

**Simulation results:**

water
Proces
water
Proces
water

Indien er een water en een plant is gekozen wordt de opstelling weergegeven van de simulatie. De gebruiker kan hierbij de eigenschappen bekijken per proces / water.

Door het klikken van een link achter deze worden de details getoond. Indien het water niet voldoet aan de eisen zal het water van kleur veranderen (rood – fout, groen – goed)

**Actie 2**

Er is zo snel mogelijk een afspraak gemaakt met de opdrachtgever voor verdere uitwerking van dit ontwerp. Omdat de uitwerking van dit ontwerp nog niet gereed is voordat de documenten op school moeten worden ingeleverd, zijn deze aanpassingen nog niet verwerkt in het technisch en functioneel ontwerp.

## Hoofdstuk 5. Evaluatie

In hoofdstuk 3.4 is de opdracht besproken. In dit hoofdstuk wordt Prototype 2 vergeleken met deze opdracht.

Behaald	Niet behaald
<ul style="list-style-type: none"><li>• Een inlogprocedure voor de gebruikers.</li><li>• Een interface voor de configuratie van water en plant.</li><li>• Een XML koppeling tussen de webapplicatie en de rekenmodule.</li><li>• Het genereren van een simulatierapport van een waterzuivering</li><li>• Output in de vorm van een rapport (lay-out rapport is nog niet naar wens)</li><li>• De historische gegevens van een simulatie opslaan in de database.</li><li>• Uitvoering van simulatie met een statische complexiteit</li></ul>	<ul style="list-style-type: none"><li>• Een interface voor de configuratie van modellen.</li><li>• Uitvoering van simulatie met een dynamische en stochastische complexiteit</li></ul>

Waarom zijn sommige functies niet behaald?

Door de late iteratieslag uitgevoerd van prototype 1 naar prototype 2 is veel tijd verloren gegaan. Hierdoor moesten de prioriteiten verlegd worden. Voor het aanmaken van een model is hierdoor in prioriteit verlaagd, omdat deze in dit stadium weinig toegevoegde waarde heeft.

Uitvoering van simulatie met een dynamische en stochastische complexiteit kon de stagiair niet voor elkaar krijgen. Voor het maken van het volgende prototype krijgt de stagiair hiervoor hulp.

De succesfactor van dit project is dat de applicatie minimaal een zuiveringsproces moet simuleren. Daarnaast moet de applicatie zo zijn ontwikkeld dat verdere uitbreiding mogelijk is. Het prototype 2 voldoet hieraan, dus is deze volgens de succesfactor behaald.



## Hoofdstuk 6. Aanbevelingen

In dit hoofdstuk worden de aanbevelingen verwerkt die betrekking hebben op het project KOOP. Als eerste worden 3 aanbevelingen gedaan die te maken hebben met de database opzet. De aanbevelingen die betrekking hebben op de GUI en functionaliteiten zijn verwerkt in "Aanbevelingen prototype 3" (Eind van dit hoofdstuk). Hierbij zal getracht worden deze uit te voeren binnen prototype 3.

Voor de applicatie zoals deze nu is opgezet, zijn geen aanbevelingen. Wel moest er rekening worden gehouden met de opzet van de portal, maar deze leverde geen belemmeringen op.

### Database Aanbeveling 1

De architectuur die nu gebruikt wordt, is gebaseerd op het afzonderlijk versturen van een watersample, dit bleek niet de juiste insteek te zijn.

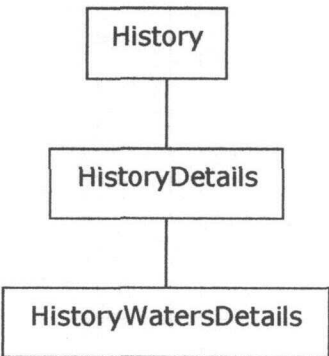
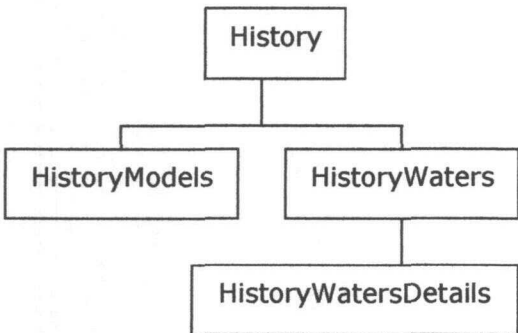
De input bevat alle waterdata, dus alle watersamples die aanwezig zijn in het water. Al deze data wordt samengevoegd en verstuurd naar het eerste model. Dit model moet intelligent genoeg zijn om te herkennen of het om een simpel water gaat (1 sample) of een dynamisch water (n samples).

In het script is deze opzet al veranderd, maar de database kan nog verder worden genormaliseerd. De reden voor het nog niet veranderd hebben van de database is de overweging tussen de impact bij verandering en de tijdsplanning voor het realiseren van de doelstelling.

### Aanbeveling

In tabel HistoryDetails wordt nu bij elke watersample de modelgegevens opgeslagen, terwijl dit niet nodig is. Hierbij raad de stagiair aan de tabel HistoryDetails van KOOP te splitsen in HistoryModels en HistoryWaters. Deze keuze heeft zowel voor- als nadelen. Het voordeel hierbij is dat er geen onnodige data wordt opgeslagen en de database is beter genormaliseerd. Door de verbetering met behulp van normalisatie is de database makkelijker te begrijpen voor buitenstaanders.

Een nadeel is wel dat de database mogelijk langzamer wordt en er meer moet worden opgeslagen in verband met tabelkoppelingen. Maar dit nadeel is nauwelijks merkbaar voor gebruikers.

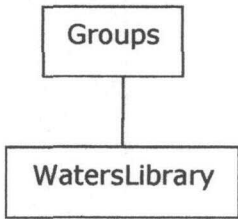
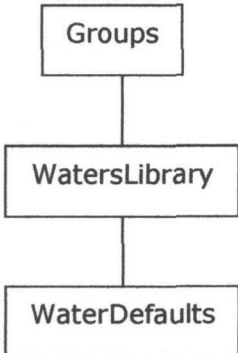
Huidige situatie Aandachtsgebied "History"	Aanbeveling Aandachtsgebied "History"
 <pre>graph TD; History[History] --- HistoryDetails[HistoryDetails]; HistoryDetails --- HistoryWatersDetails[HistoryWatersDetails];</pre>	 <pre>graph TD; History[History] --- HistoryModels[HistoryModels]; History[History] --- HistoryWaters[HistoryWaters]; HistoryWaters --- HistoryWatersDetails[HistoryWatersDetails];</pre>

### Database Aanbeveling 2

Water kan op verschillende locaties gewonnen worden, een voorbeeld hiervan is grondwater, slootwater, afvalwater, etc. Elk water heeft zo zijn eigen specifieke kenmerken (properties). In de uiteindelijke "Stimela" applicatie moet er van deze verschillende type wateren een keuze worden gemaakt waardoor de default-waarden van het water veranderd. In de database opzet op dit moment is maar één default-waarden opgenomen, welke niet uitbreidbaar is.

#### Aanbeveling

Indien het gewenst is meerdere watertypen op te nemen in de database, moeten de default-waarden uit "WatersLibrary" worden gehaald en moet hiervan een nieuwe tabel gemaakt worden (WatersDefault), met daarin de NaamTypeWater+Defaultwaarde (zie hieronder weergegeven).

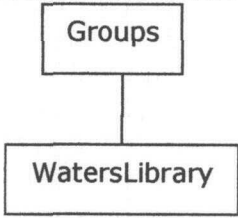
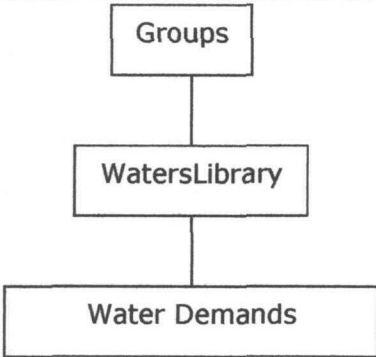
Huidige situatie Aandachtsgebied "WatersLibrary"	Aanbeveling Aandachtsgebied "WatersLibrary"
	

### Database Aanbeveling 3

Elk land stelt andere eisen aan de samenstelling van het water. In de applicatie die tot nu toe is gerealiseerd, is alleen rekening gehouden met de eisen die in Nederland worden gesteld.

#### Aanbeveling

Aangezien dit een internationale applicatie moet worden is het aan te raden uit de "WatersLibrary" de Demands te halen en een aparte tabel te vormen met daarin de Demands met b.v. land en doeleinden (slootwater, drinkwater en dergelijke) te verwerken (zie hieronder weergegeven).

Huidige situatie Aandachtsgebied "WatersLibrary"	Aanbeveling Aandachtsgebied "WatersLibrary"
	

### **Aanbeveling prototype 3**

Na het inleveren van de scriptie is er nog een maand over voor het werk aan prototype 3 (tot einde januari). In deze maand zal getracht worden de volgende functionaliteiten en GUI aanpassingen te realiseren:

1. Uitvoering van simulaties voor meerdere watersamples
2. Verandering van de GUI
3. Kleurweergave bij water afwijkend van de regeringseisen
4. Invoeren van watersamples
5. Aanpassen van processen (veroorzaakt door verandering van de GUI)
6. Aanpassen van watereigenschappen (veroorzaakt door verandering van de GUI)

Deze functionaliteiten staan in volgorde van belangrijkheid aangegeven en zullen ook in deze volgorde worden uitgevoerd.

Naast de oplevering van deze functionaliteiten zal er een technische handleiding worden geschreven die betrekking heeft op het ontwerp van "Stimela"

### **Hoofdstuk 7. Conclusie**

De applicatie die tot nu toe gerealiseerd is, komt overeen met de doelstelling die van tevoren gesteld is. Door het uitvoeren van de applicatie kan de gebruiker inzicht krijgen wat er gebeurt bij een waterzuivering. Hierbij worden de resultaten van een simulatie getoond en kan er dieper op de plant worden ingegaan, zodat de processen ook te bekijken zijn.

De opzet van de GUI krijgt ook steeds meer vorm en duidelijkheid. Dit is met name gekomen door de iteratieslagen binnen dit project. Hierbij is na elke prototype bespreking nagegaan wat er allemaal moet worden veranderd.

De definitieve opzet wordt getracht in prototype 3 te realiseren.

Het enige minpunt van prototype 2 over de werking, is dat er maar één watersample kan worden gesimuleerd. De juiste werking voor meerdere watersamples zal wel worden gerealiseerd in prototype 3.



## **Hoofdstuk 8. Evaluatie**

### **Proces**

#### *Evaluatie DSDM*

De combinatie tussen onderzoek – prototyping – feedback was goed bevallen. Wel had er bij nader inzien in een eerder stadium een feedback fase moeten komen bij prototype 1. Door de reactie van de opdrachtgever ben ik erachter gekomen dat het moeilijker is vanuit een document een visualisatie van de applicatie te maken. Daarnaast, doordat ik projectlid was van "KOOP - Demo", is er teveel als vanzelfsprekend aangenomen, waardoor er te snel onjuiste eigen conclusies werden getrokken.

#### *Evaluatie MoSCoW*

In het beginstadium is er volgens het MoSCoW principe een belangenafweging gemaakt tussen verschillende functies van de applicatie. Bij nader inziens bleek dit niet nodig te zijn omdat er niet veel verder is gekomen dan de "Must" en "Should have". Wel heeft dit principe mij duidelijk gemaakt welke functies er belangrijk waren en welke konden worden overgeslagen (wenselijk waren).

### **Techniek**

#### *Evaluatie programmeertaal*

Visual Studio.Net is een duidelijke en overzichtelijke programmeeromgeving. Hierbij is veel aandacht geschonken aan OO gemoduleerd werken. Wel was het moeilijk om de weg te vinden tussen de vele mogelijkheden die deze applicatie ondersteunt.

#### *Evaluatie Portal*

Waar ik erg over te spreken ben is de gebruikte portal Dotnetnuke. Deze was zeer goed opgebouwd, gebruikersvriendelijk en makkelijk uitbreidbaar. Ook alle documentatie over de portal was goed te vinden op het Internet.

Op de website van Dotnetnuke is aandacht geschonken aan veel voorkomende problemen door de opzet van een discussie forum en het heeft mij ook enkele malen weer op het rechte pad gezet.

### **Begeleiding**

#### *Evaluatie begeleiding school*

Ongeveer eens per maand is er een overleg geweest tussen stagiair en begeleider. Dit had voornamelijk betrekking op de documentatie en voortgang van het project. Tijdens deze bijeenkomsten zijn er mondelinge afspraken gemaakt, zoals inleverdata voor het rectificeren van de documentatie. Hierop is geen commentaar nodig.

#### *Evaluatie begeleiding DHV*

De begeleider voelt zich betrokken bij het project en heeft in het beginstadium goed meegedacht aan documentatie en applicatie. Aan het eind van het project had de begeleider het zelf druk waardoor hij minder tijd aan mij kon spenderen. Hierdoor moest ik vaak zelf op zoek gaan naar de antwoorden. Dit kost natuurlijk meer tijd maar heeft hetzelfde resultaat en een hoger leermoment.

Daarnaast had ik vrij veel moeilijkheden om mijn begeleider als begeleider te zien en niet als opdrachtgever. Hierdoor kon het voorkomen dat sommige vragen bij de verkeerde persoon terechtkwamen.

## **Wat had ik anders aan moeten pakken**

### *Probleemconstatering 1*

Het contact met de opdrachtgever was vanuit mijn kant te weinig. Dit was in een eerder stadium geconstateerd en hierop gereageerd door het online zetten van het, tot dan toe, behaalde resultaat van de applicatie. Het resultaat hiervan was ook niet naar wens, omdat de site te weinig werd vernieuwd (reden: vernieuwen kost veel tijd (met name de aanpassingen in het SQL script) en je moet zelf ook tevreden zijn met het resultaat voordat je het online zet.

### *Oplossing:*

In plaats van sporadische afspraken met de opdrachtgever, vaste terugkoppeldata met niet al te veel tussentijd. Hierdoor is het makkelijker te sturen door de opdrachtgever.

### *Probleemconstatering 2*

De opdrachtformulering was in het begin niet duidelijk voor dit project. Hierbij was verwarring tussen de opdrachtformuleringen van het project "Stimela" (totale project) en het "Koop" project.

### *Oplossing:*

Dit is eigenlijk een gevolg van de eerste probleemconstatering. Door meer communicatie kan dit probleem in een eerder stadium worden verholpen.



## Literatuurlijst

### Geraadpleegde boeken

Titel boek	Auteur	Uitgeverij	ISBN #	Jaar van uitgave
ASP.NET In 24 uur	Joe Martin Brett Tomson	Pearson Education	90-430- 0539-8	2002
XML In 24 uur	Charles Ashbacher	Pearson Education	90-430- 0420-0	2001
DSDM	Jennifer Stapleton	Academic service	90-395- 1091-1	2001
SQL Leerboek	F. van de lams	Academic Service	90-395- 0755-4	1998
Usability Engineering	M.B. Rosson en J.M. Carrol	Academic Service	1-55860- 712-9	2002
Literatuur KOOP- Demo	Meryem Soykan, Maxim van Ommen Kloeke, Naser Bakhshi Erwin Hendriks	nvt	nvt	2002

### Geraadpleegde Websites

Site	Onderwerp
<a href="http://www.computable.nl/artikels/archief2/d51ra2lo.htm">http://www.computable.nl/artikels/archief2/d51ra2lo.htm</a>	Meerwaarde DSDM
<a href="http://www.revjon.com/ftb/">http://www.revjon.com/ftb/</a>	freetextBox
<a href="http://www.array.nl/release/artikelen/2002/2_leeuw_Prince2.pdf">http://www.array.nl/release/artikelen/2002/2_leeuw_Prince2.pdf</a>	Prince 2
<a href="http://www.w3schools.com">http://www.w3schools.com</a>	Tutorials
<a href="http://Sourceforge.net/search">http://Sourceforge.net/search</a>	DNN modules
<a href="http://www.lucaslabs.net/desktopDefault.aspx">http://www.lucaslabs.net/desktopDefault.aspx</a>	DNN modules
<a href="http://www.dotnet247.com/247reference/messages/21/108937.aspx">http://www.dotnet247.com/247reference/messages/21/108937.aspx</a>	Date&time now
<a href="http://www.dynst.com/desktopdefault.aspx">http://www.dynst.com/desktopdefault.aspx</a>	Manuals and Handbooks
<a href="http://www.w3.org">http://www.w3.org</a>	World wide web consortium
<a href="http://www.microsoft.com/isapi/redir.dll?prd=ie&amp;pver=6&amp;ar=IStart">http://www.microsoft.com/isapi/redir.dll?prd=ie&amp;pver=6&amp;ar=IStart</a>	Vragen

Hogeschool  
van Utrecht

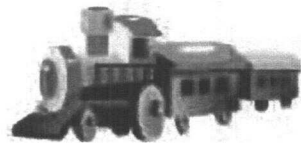


Advies- en Ingenieursbureau

Plan van Aanpak

# KOOP

Koppeling Onderzoek, Onderwijs en Praktijk



## Bijlage 1

Student:  
Studentnr:  
Eerste examiner:  
Stagebedrijf:

E.N.J. Hendriks  
1106061  
H. Plessius  
DHV Water

## **Samenvatting**

Dit document bevat de aspecten hoe de applicatie wordt opgezet, dus hoe het verleden is; wat het programma moet doen; en op welke wijze het word aangepakt.

Als achtergrond van dit project loopt de applicatie KOOP demo. Waarbij ik tevens projectlid was. Deze documentatie wordt als grondslag gehouden van dit project.

Het hoofddoel van de applicatie is het toegankelijk maken van verschillende processen voor waterzuivering ter assistentie van het lesmateriaal. Hierbij kan een gebruiker een bepaald water en een plant (zuiveringsbedrijf) selecteren, en hierbij kijken wat het resultaat is van het water.

De applicatie KOOP is te vergelijken met een modeltrein waar iedere modelmaker zijn eigen wagon (model / proces) achter kan plaatsen. Wel moet de wagon aan bepaalde eisen (Properties) voldoen, zodat het aangekoppeld kan worden aan de trein.

De applicatie aanpak zal worden uitgevoerd volgens de DSDM methode in combinatie met het MoSCoW Principe. Dit houdt in dat alle functionaliteiten zijn opgedeeld in bepaalde graden van belangrijkheid. Indien een bepaald gedeelte is afgerond, zal deze worden gecontroleerd door de opdrachtgever en stagebegeleider.

Na het opzetten van de applicatie kan de gebruiker een model en een plant selecteren en een waterzuivering simuleren. De resultaten worden weergegeven op het scherm met de aanduiding of dit voldoet aan de eisen die gesteld worden aan het water.

## Inhoudsopgave

<b>Hoofdstuk 1: Introductie .....</b>	<b>43</b>
<b>Hoofdstuk 2: Uitgangssituatie .....</b>	<b>44</b>
2.1 <i>Uitgangsdokumentatie .....</i>	<i>44</i>
<b>Hoofdstuk 3: Projectomschrijving .....</b>	<b>45</b>
3.1. <i>Doelstelling .....</i>	<i>45</i>
3.2. <i>Kwaliteitseisen .....</i>	<i>45</i>
3.3. <i>Succesfactor .....</i>	<i>45</i>
3.4. <i>Opdracht .....</i>	<i>46</i>
3.5. <i>Risicofactoren.....</i>	<i>47</i>
<b>Hoofdstuk 4: Projectfasering.....</b>	<b>48</b>
4.1 <i>Inleiding.....</i>	<i>48</i>
4.2 <i>Applicatiefasering .....</i>	<i>48</i>
4.3 <i>Documentfasering .....</i>	<i>49</i>
<b>Hoofdstuk 5: Projectkader .....</b>	<b>50</b>
5.1 <i>Inleiding.....</i>	<i>50</i>
5.2 <i>Projectorganisatie.....</i>	<i>50</i>
5.3 <i>Voorwaarden aan de stagebegeleider.....</i>	<i>51</i>
5.4 <i>Voorwaarden aan de schoolbegeleider.....</i>	<i>51</i>
<b>Hoofdstuk 6: Planning.....</b>	<b>52</b>
6.1 <i>Milestones .....</i>	<i>53</i>
<b>Hoofdstuk 7: Goedkeuring.....</b>	<b>54</b>
<b>Hoofdstuk 8: Aanpassingen.....</b>	<b>55</b>
<b>Bijlage 1: Thin-Client Architectuur .....</b>	<b>58</b>

## **Hoofdstuk 1:           Introductie**

Dit document dient voor het inzichtelijk krijgen hoe het project KOOP wordt aangepakt. Het is geschreven door stagiaire E. Hendriks studerende aan de Hogeschool van Utrecht opleiding Information Engineering, in opdracht van DHV Water.

Het plan van aanpak is geschreven voor Dhr de Moel als kickoff van het project "Koop". Daarnaast dient dit document als naslagwerk voor Dhr. P. Nijdam (stagebegeleider(DHV Water – PA-ICT)) en Dhr H. Plessius (afstudeerbegeleider (Hogeschool van Utrecht)).

Het project Koop staat voor Koppeling Onderzoek, Onderwijs en Praktijk voor drinkwater.

KOOP wordt een applicatie waarmee waterzuiveringen worden gesimuleerd. Elke gebruiker kan een bepaalde input (type water) geven aan de simulatie. Op dit type water wordt een waterzuivering toegepast. Elke waterzuivering (plant) is een aaneenschakeling van waterzuiveringsprocessen (modellen) en is in zijn complexiteit instelbaar. De modellen worden door professionals en / of promovendi opgesteld en aangepast.

Na het simuleren van een waterzuivering krijgt de gebruiker een presentatie van alle tussen resultaten van het water na het toepassen van elk model, en het behaalde eind resultaat.

De essentie van de applicatie is dat gebruikers over de gehele wereld deze berekeningen kunnen uitvoeren.

In de volgende hoofdstukken worden de volgende punten behandeld:

- Uitgangssituatie
- Projectresultaat
- Projectfasering
- Projectkader
- Planning

Verder zijn als bijlagen bij dit "Plan van Aanpak" de volgende onderdelen opgenomen:

Bijlage 1:       "Thin Client architectuur"

Hierin staat de architectuur vermeldt die betrekking heeft op het project KOOP.



## **Hoofdstuk 2:                   Uitgangssituatie**

Momenteel zijn er al wel modellen die de berekeningen voor waterzuivering uitvoeren. Alleen zijn deze modellen van elk proces afzonderlijk (dus geen geïntegreerde modellen) en zijn deze niet toegankelijk via het internet. Daarnaast is voor het opzetten van modellen gespecialiseerde kennis nodig en dus moeilijk als leermodel te gebruiken.

Met dit project wordt het mogelijk om vanaf het Internet, eventueel tegen betaling, berekeningen voor de waterzuiveringsprocessen en de gehele waterzuivering procedure te kunnen uitvoeren.

Het project KOOP is vooraf gegaan met het project KOOP Demo. Hierdoor zijn bepaalde delen van het project al vastgelegd, zoals structuur van de database.

Het budget dat ter beschikking is gesteld door de TU Delft voor het project Koop bedraagt ± 700.000 euro. Hierbij is de helft van het bedrag verkregen door het College van Bestuur, de andere helft door de TU Delft zelf.

### **2.1       Uitgangsdokumentatie**

Als uitgangsdokumentatie is er beschikking over de volgende documenten:

Inleiding watermanagement

Alle documentatie van het "KOOP Demo" project.

## Hoofdstuk 3: Projectomschrijving

### 3.1. Doelstelling

Het project heeft de volgende doelstellingen:

Een simulatieomgeving te creëren waarbij de potentiële klanten, studenten en promovendi door middel van een webapplicatie inzicht krijgen in, en uitvoeren van, simulaties met zuiveringsprocessen van water.

### 3.2. Kwaliteitseisen

Het primaire eindproduct is een werkende webapplicatie, ook wel "virtueel atelier" genoemd, die minimaal aan de volgende eisen zal voldoen:

- WEB enabled;
- gebruikersauthenticatie met login + gebruikersmanagement;
- interface voor configuratie van:
  - water eigenschappen (input voor het model);
  - plant parameters;
  - model parameters;
- koppeling met rekenmodule Stimela;
- presentatie resultaten;
- opslag van gebruikersgegevens en resultaten in een database.

Aandachtspunten:

- Duidelijke en overzichtelijke GUI
- Modulaire opbouw
- Flexibiliteit

De op te leveren documenten zijn:

- Plan van Aanpak
- Functioneel ontwerp
- Technisch ontwerp
- Functionele handleiding
- Technische handleiding

### 3.3. Succesfactor

*Wanneer is het project succesvol?*

De kwaliteit van de te leveren applicatie is sterk afhankelijk van de in paragraaf 3.2 genoemde eisen die aan de applicatie gesteld worden. De gecreëerde webapplicatie zal in staat zijn om minimaal een zuiveringproces te simuleren. De applicatie zal zodanig worden ontwikkeld zodat verdere uitbreiding mogelijk is. Daarvoor moeten ook de documentatie en de beschrijving van de applicatie compleet en overzichtelijk zijn.

### 3.4. Opdracht

De opdracht in precieze bewoording luidt als volgt:

Bouw een internetapplicatie die de volgende functionaliteiten heeft:

- Een inlogprocedure voor de gebruikers;
- Een interface voor de configuratie van water, plant en model parameters.
- Een XML koppeling tussen de webapplicatie en Stimela;
- Het genereren van een simulatie van een waterzuivering;
- Output in de vorm van een rapport, eventueel ondersteund door een grafiek.
- De historische gegevens opslaan in de database. Hierdoor kan er snel naar vorige resultaten van de simulatie worden teruggezocht.

Tevens zal de applicatie bestaan uit meerdere facetten die in niveau van complexiteit verschillen, namelijk simpel, normaal en complex. De drie verschillende facetten zijn: Statisch(zonder het tijdsaspect), Dynamisch(met het tijdsaspect) en Stochastisch(prognose).

	Statisch	Dynamisch	Stochastisch
Simpel			
Normaal			
Complex			

### **3.5. Risicofactoren**

Om het project soepel te laten verlopen is het nodig dat er een risicoanalyse wordt gemaakt. Hierdoor kan men beter voorbereid zijn op de eventuele tegenslagen die tijdens het uitvoeren van het project kunnen voorkomen.

Hieronder worden er een aantal risico's genoemd die een dreiging voor het goede afloop van het project kunnen vormen. Er wordt ook aangegeven hoe men de onderstaande risico's kan vermijden.

1)

Ziekte van de stagiaire zou de productiviteit, en dus het eindresultaat negatief beïnvloedt. Dit risico kan niet vermeden worden, wel kan er door een gedegen planning het risico worden verminderd.

2)

Door een defect aan de apparatuur zou het project vertraging kunnen lopen. Dit is te voorkomen door regelmatig een back-up te maken. Hierop is besloten van elke dag een historie file op te slaan. Hierdoor kan ook het risico van verkeerd opslaan voorkomen worden.

3)

Door onvoldoende kennis van de stagiaire is het mogelijk opstartproblemen te krijgen. Hierdoor zou het mogelijk zijn dat de applicatie niet aan de gewenste eisen voldoet. Om dit risico te minimaliseren is er een boek van asp.net en xml aangeschaft. Daarnaast wordt er gebruik gemaakt van het MoSCoW principe, waardoor er meer zekerheid komt in de realisatie van de minimale eisen.

4)

Verkeerde interpretatie van de gemaakte afspraken omtrent het project zou ertoe leiden dat het eindresultaat niet naar wens is van de opdrachtgever. Dit risico is nog groter geworden omdat de informatie niet meer uit eerste hand is, maar de meeste informatie verkregen wordt door de begeleider bij DHV.

Doordat het vorige project "KOOP Demo" naar wens was afgesloten, kan hier veel informatie uit onttrokken worden. Dit voorkomt grotendeels eventuele miscommunicatie. Daarnaast is de stagiair bij dit project betrokken is geweest.

5)

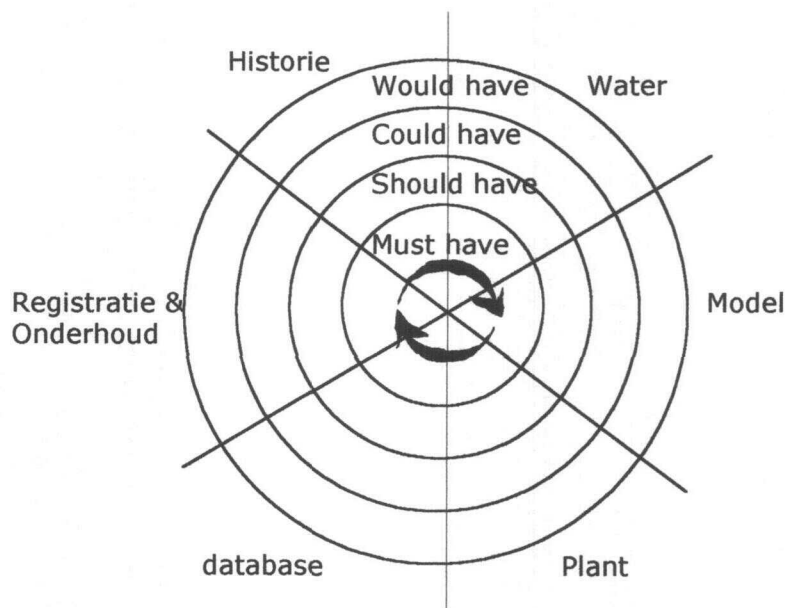
De aangekondigde reorganisatie binnen het bedrijf DHV. Indien mijn stagebegeleider ontslagen gaat worden, zit ik zonder. Hier is nog geen oplossing voor gevonden, omdat dit sterk afhangt van de situatie.

**4.1 Inleiding**

De projectfasering is opgedeeld in 3 hoofdfasen, namelijk de oriënterende fase waarin de stagiaire documentatie tot zich kan nemen van het bedrijf tot opdracht; de applicatiefasering waarin het maken van de applicatie opgedeeld is (zie hoofdstuk 4.2); en de documentatie fase (zie hoofdstuk 4.3).

**4.2 Applicatiefasering**

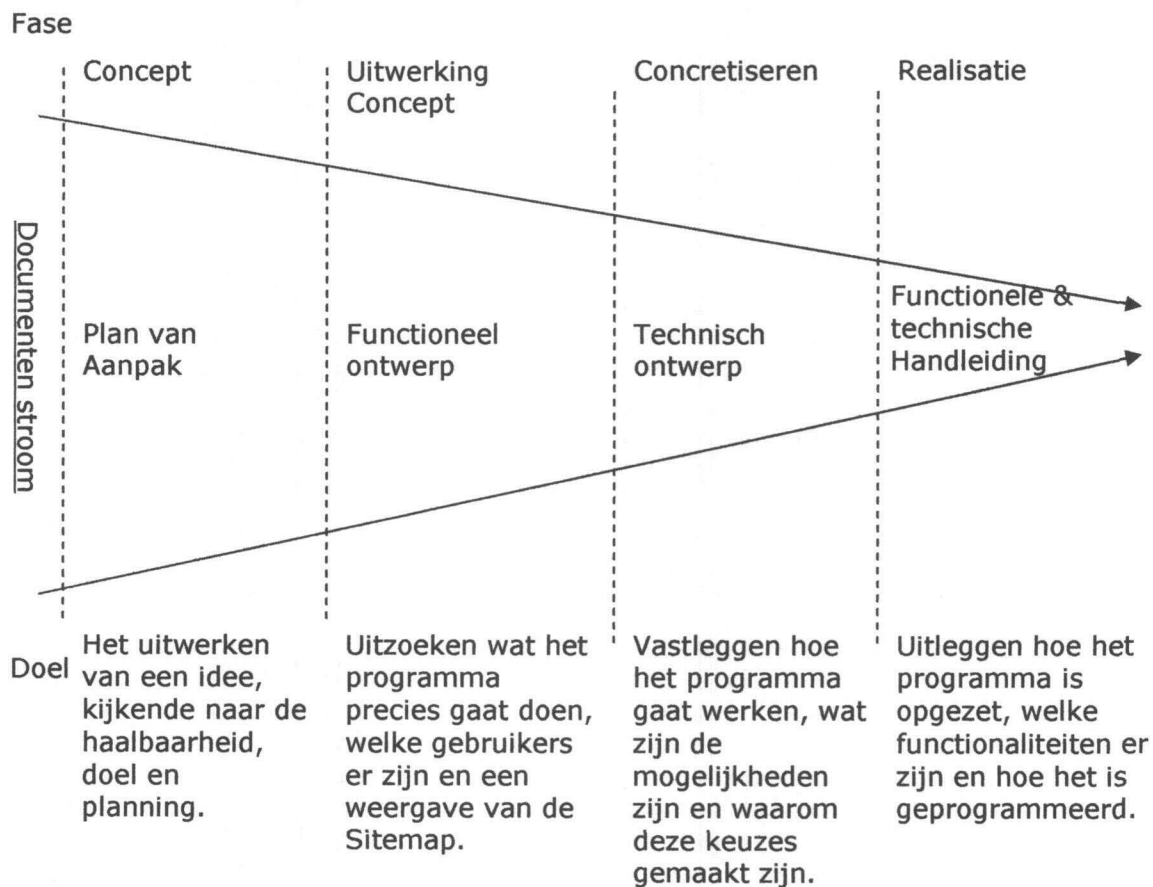
In de fasering wordt het project opgedeeld in verschillende facetten middels het MoSCoW principe (MoSCoW is een schaalverdeling van 4 graden: Must have; should have; could have; would have; waarin een selectie wordt gemaakt op het belang van de fase). Hieronder is een weergave gemaakt hoe het project is opgezet. Er wordt hierbij als een spiraal naar buiten gewerkt, beginnende met must have. Het MoSCoW principe wordt toegepast binnen het kader van DSDM. Hierbij wordt bij het overschrijden van een lijn (b.v. van Must naar Should have) de applicatie besproken en getest. Door deze toepassing heeft de klant snel een indicatie waar het project zich bevindt, of het naar zijn wens is en of het project moet worden bijgestuurd.





### 4.3 Documentfasering

De documentfasering is opgedeeld in 4 fases, deze staan weergegeven in de onderstaande tekening. Hierbij staat per fase het op te leveren document en het doel van dit document vermeld. Hoewel de documenten hieronder staan opgedeeld kunnen/zullen deze veranderen naar mate het project qua inhoud veranderd.



## **Hoofdstuk 5:           Projectkader**

### **5.1     Inleiding**

Voor het welslagen van het totale project is het projectkader belangrijk. Het project moet gerealiseerd worden binnen bepaalde beperkende randvoorwaarden.

Hier wordt aandacht gegeven aan de volgende aspecten:

De projectorganisatie

Voorwaarden aan de stagebegeleider (Dhr. P. Nijdam)

Voorwaarden aan de schoolbegeleider (Dhr. H. Plessius)

### **5.2     Projectorganisatie**

De projectorganisatie is de vorm die het tijdelijke samenwerkingsverband aangeeft. Hierbij wordt er van twee hulpmiddelen gebruik gemaakt.

- De profielschets
- Het organogram

In de profielschets wordt duidelijk gemaakt wat voor soort functionarissen in het project zijn en wat hun functie is. In dit project worden er vier profielschetsen beschreven die belang hebben voor het welslagen van dit project. De functionarissen voor dit project worden hieronder genoemd.

#### *De stagebegeleider:*

De stagebegeleider heeft de taak de stagiaire te begeleiden met het uitvoeren van zijn stageproject. Hierbij wordt hij geacht van het project op de hoogte te zijn en de stagiaire hierover voldoende informatie te kunnen geven. De stagebegeleider bij DHV Water (project KOOP) is P. Nijdam.

#### *Hoofd Projectleider:*

De opdrachtgever van dit project is ir. P.J. de Moel, toegevoegd docent binnen de Sectie Gezondheidstechniek van de Afdeling Watermanagement binnen de Faculteit Civiele Techniek en Geowetenschappen (TU Delft leerstoel drinkwater). Zijn taak is informatie te verstrekken wat betrekking heeft op het project en duidelijk aan te geven wat zijn wensen zijn.

#### *De stagiaire:*

De stagiaire heeft de verantwoordelijkheid voor de dagelijkse gang van zaken in zijn aandachtsgebied. De taken en verantwoordelijkheden van de stagiaire zijn:

- De voortgang van het project bewaken en indien nodig andere te informeren hierover.
- Actief meedenken, meewerken en initiatieven tonen met betrekking tot het project KOOP.

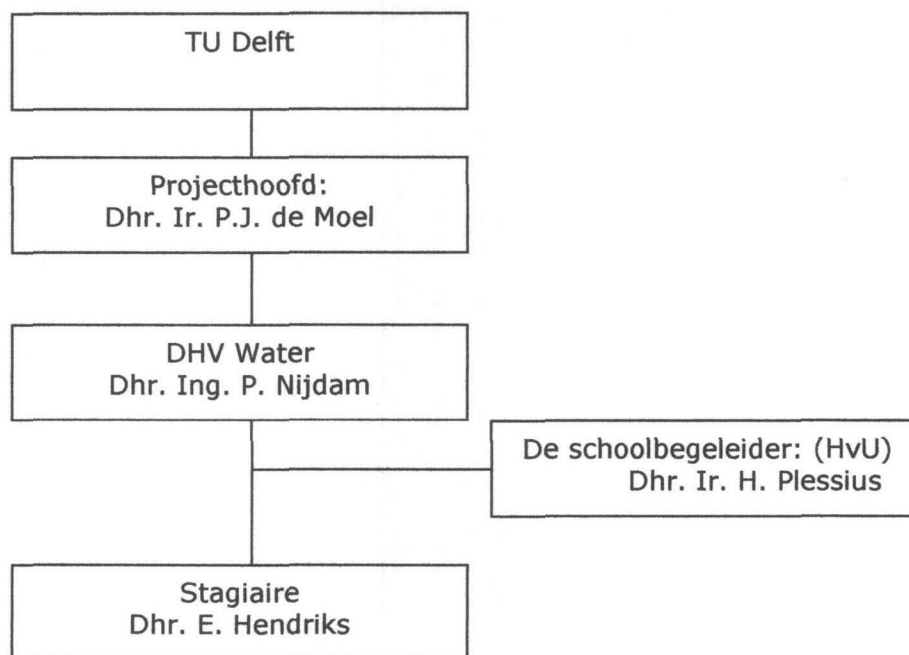
De stagiaire is E. Hendriks (opleiding Information Engineering aan de Hogeschool van Utrecht.)

#### *De schoolbegeleider:*

De schoolbegeleider van dit stageproject is ir. H. Plessius, tevens eerste examinator van de stagiaire.

Hieronder wordt door middel van een organogram zichtbaar gemaakt wat de onderdelen (de organen) van de projectorganisatie zijn en hoe deze ten opzichte van elkaar staan.

#### Organogram



### **5.3 Voorwaarden aan de stagebegeleider**

Er zijn voorwaarden die de stagebegeleider dient te realiseren voor het welslagen van het project. Deze voorwaarden worden hieronder aangegeven. Van de stagebegeleider wordt verwacht dat deze:

- middelen en/of de benodigde informatie ter beschikking aanbied;
- goed bereikbaar is (via mail, mobiele telefoon);
- actief deelneemt aan het project;
- tijdig een melding geeft indien hij niet tevreden is met het behaalde resultaat;
- beslissingsbevoegd zijn.

### **5.4 Voorwaarden aan de schoolbegeleider**

Er zijn voorwaarden die de schoolbegeleider dient te realiseren voor het welslagen van het project. Deze voorwaarden worden hieronder aangegeven. Van de schoolbegeleider wordt verwacht dat deze:

- goed bereikbaar is (via mail, mobiele telefoon);
- tijdig zal melden indien hij niet tevreden is met het behaalde resultaten.
- de verwachtingen en plichten aan de stagiaire duidelijk kenbaar zal maken.
- een beoordeling zal geven over de rapporten van het project KOOP.

## Hoofdstuk 6: Planning

	Weeknummers																											
	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	1	2	3	4						
Oriënteren	X																											
Documentatie doorlezen																												
Functioneel ontwerp																												
Technisch ontwerp																												
Functionele handleiding																												
Technische handleiding																												
Must haves (indeling zie Hoofdstuk 4)																												
Should haves (indeling zie Hoofdstuk 4)																												
Could haves (indeling zie Hoofdstuk 4)																												
Would haves (indeling zie Hoofdstuk 4)																												
Scriptie																												

 Vervachte week met deze fase bezig  
 Reserve week



## **6.1 Milestones**

### **31 oktober 2003**

Alle "Must haves" zijn gerealiseerd. Dit houdt in dat er een prototype is, dat voornamelijk aan de onderstaande eisen voldoet:

- Het maken van een plant;
- Het koppellen van een model aan een plant;
- Het maken van een water;
- Het koppellen van een serie water properties aan een water;
- Het selecteren van een water en een plant waarmee kan worden gesimuleerd.

Het doel hiervan is een presentatie te geven aan de klant (Ir. P. de Moel)

### **7 oktober 2003**

Technische handleiding af van het bovenstaande

### **23 oktober 2003**

Functioneel ontwerp af

### **23 oktober 2003**

Analyse portals af

### **4 december 2003**

Alle "Should haves" zijn gerealiseerd. Het bestaande prototype is nu uitgegroeid met meer functionaliteiten. De eisen hiervoor worden nader beschreven in het technische ontwerp.

### **12 december 2003**

Alle documentatie gerealiseerd ten opzichte van de aanvullende should have functionaliteiten.

### **17 december 2003**

Inleveren scriptie

### **9 januari 2004**

Alle "Could haves" zijn gerealiseerd. De eisen hiervoor worden nader beschreven in het technische ontwerp

### **16 januari 2004**

Alle documentatie gerealiseerd ten opzichte van de aanvullende "Could have" functionaliteiten.

### **23 januari 2004**

Alle "Would haves" zijn gerealiseerd. De eisen hiervoor worden nader beschreven in het technische ontwerp

### **30 januari 2004**

Alle documentatie gerealiseerd ten opzichte van de aanvullende "Could have" functionaliteiten.

Functionele en technische handleiding af



## **Hoofdstuk 7: Goedkeuring**

Ondergetekenden gaan akkoord met de inhoud van dit Plan van Aanpak.

Naam	Functie	Datum	paraaf
<b>Ir. H. Plessius</b>	<b>Schoolbegeleider</b>		
<b>P. Nijdam</b>	<b>Stagebegeleider</b>		

## Hoofdstuk 8:           Aanpassingen

Dhr. Ir. P. De Moel heeft op 6 november 2003 het prototype nagekeken. Deze was niet geheel tevreden over de presentatie van de GUI aan de gebruiker. Zijn commentaar was dat een niet bekende gebruiker de weg zou kwijtraken in de applicatie. Hierdoor zou het product geen potentiële gebruikers trekken.

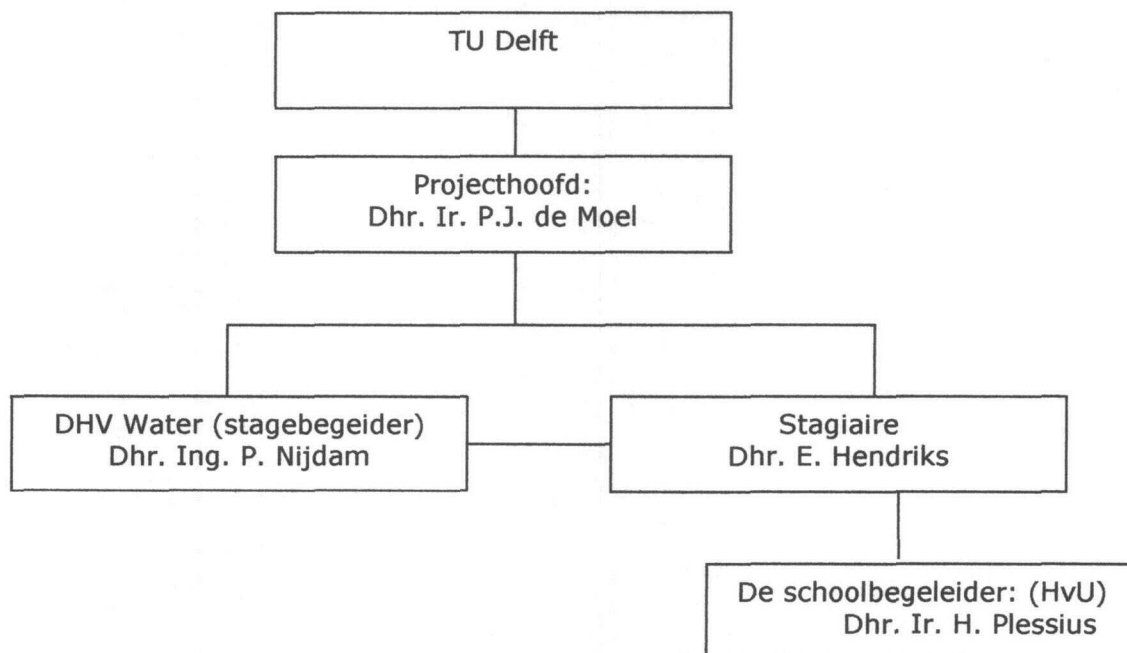
Op 7 november is er opnieuw een grafische opzet gemaakt aan de hand van het prototype KOOP Demo.

Op 12 November is de applicatie KOOP online gegaan met de aangepaste GUI. Hierbij is nu het voordeel dat het projecthoofd direct terugkoppeling kan geven indien deze verwacht dat het verkeerd gaat. Daarbij is het nu mogelijk om alle documentatie voor deze applicatie online te presenteren.

Een aanpassing in de tijdsplanning was bij de vernieuwing van de GUI noodzakelijk, deze aanpassing staat op de volgende pagina weergegeven, inclusief de nieuwe milestones.

### Organogram

Na de controle van het eerste prototype door dhr Ir. P. De Moel is gebleken dat het organogram vermeld in hoofdstuk 5.2 niet geheel correct was. De kennis van de GUI over project "KOOP" ligt vooral bij het projecthoofd. Deze beslist welke functionaliteiten in Koop moeten zitten en hoe het gepresenteerd moet worden aan de gebruiker.



# Aanpassing Planning

	Weeknummers																						
	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	1	2	3	4	
Oriënteren																							
Documentatie doorlezen																							
Functioneel ontwerp																							
Technisch ontwerp																							
Functionele handleiding																							
Technische handleiding																							
Must have's (indeling zie Hoofdstuk 4)																							
Should have's (indeling zie Hoofdstuk 4)																							
Could have's (indeling zie Hoofdstuk 4)																							
Would have's (indeling zie Hoofdstuk 4)																							
Scriptie																							

 Vervachte week met deze fase bezig

 Reserve week

## **Milestones**

### **Week 51**

Alle "must haves" af

### **Week 53**

Technische handleiding af van alle "must haves"

### **17 december 2003**

Inleveren scriptie

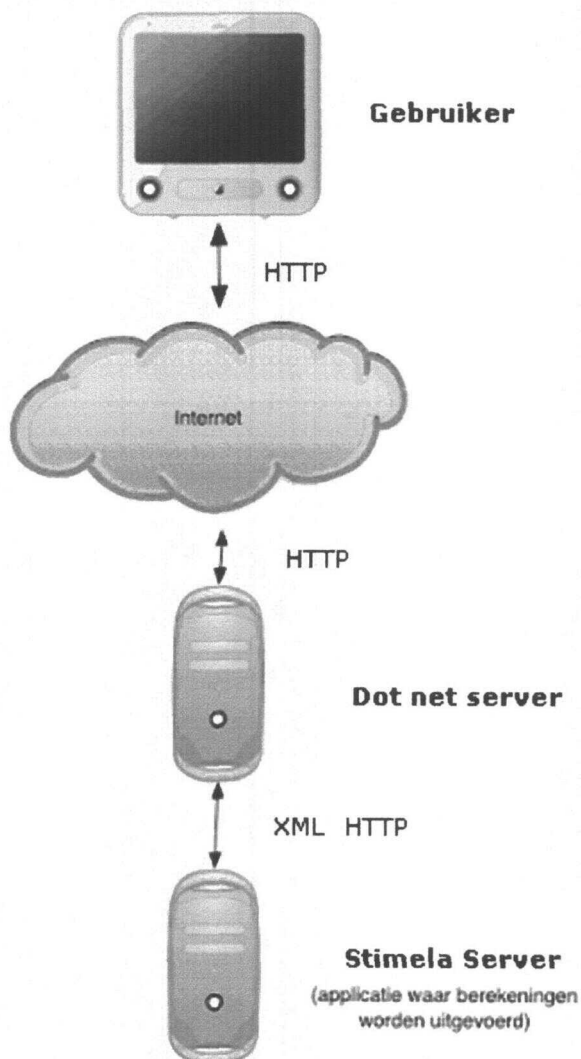
### **Week 2**

Zo veel mogelijk af van de "should haves"

### **Week 4**

Technische en functionele handleiding af



**Bijlage 1:****Thin-Client Architectuur****Architectuur Hardware****Uitleg:**

Een gebruiker kan inloggen in de applicatie KOOP via het internet. In deze applicatie kan de gebruiker een type water en een model kiezen.

De wensen van de gebruiker worden opgevangen door de Server. Deze zal de wensen van de klant met de betreffende data uit de database van de server halen, en omzetten naar XML.

Na het converteren naar XML worden deze gegevens met http formaat verstuurd naar Stimela (dit kan ook een andere server zijn ergens op de wereld)

Stimela gaat met deze data rekenen (hoe de berekeningen plaats vinden valt buiten de scope), en zal de uitkomst naar de server retourneren.

De data zal worden opgeslagen in de database van de Server. En de gebruiker wordt op de hoogte gesteld van de berekende resultaten.

Hogeschool  
van Utrecht



Advies- en Ingenieursbureau

Functioneel ontwerp

# KOOP

Koppeling Onderzoek, Onderwijs en Praktijk



## Bijlage 2

Student:  
Studentnr:  
Eerste examiner:  
Stagebedrijf:

E.N.J. Hendriks  
1106061  
H. Plessius  
DHV Water

## **Samenvatting**

Dit document heeft betrekking op het project KOOP (Koppeling onderzoek, onderwijs en Praktijk). Het doel van de applicatie is het simuleren van een plant door studenten.

In totaal kunnen er 4 gebruikers worden gedefinieerd, namelijk de modelgebruiker (b.v. student), de modelbouwer (b.v. promovendi of externe bedrijven) en de niet geregistreerde gebruiker. De applicatie wordt onderhouden door de systeembeheerder.

Elke gebruiker heeft zijn eigen specifieke mogelijkheden.

- Namelijk een modelgebruiker gebruikt alleen de beschikbare processen voor het opstellen van een plant. De (gast) modelgebruiker kan een simulatie uitvoeren en de resultaten bekijken.
- De modelbouwer daarentegen kan wel eigen processen opstellen. Daarnaast kan hij net als de gebruiker plants en water veranderen en simuleren.
- De niet geregistreerde gebruiker heeft de mogelijkheid alleen de van tevoren opgestelde plants en waters uit te voeren.
- De systeembeheerder is hoofdzakelijk voor het onderhoud van de website en database.

## Inhoudsopgave

<b>Hoofdstuk 1: Introductie .....</b>	<b>62</b>
1.1 <i>Inleiding .....</i>	<i>62</i>
1.2 <i>Doel.....</i>	<i>63</i>
<b>Hoofdstuk 2: De gebruikers.....</b>	<b>64</b>
2.1 <i>De modelbouwer .....</i>	<i>64</i>
2.2 <i>De modelgebruiker .....</i>	<i>64</i>
2.3 <i>De systeembeheerder .....</i>	<i>65</i>
2.4 <i>De niet geregistreerde gebruiker .....</i>	<i>65</i>
<b>Hoofdstuk 3: Opzet sitemap .....</b>	<b>66</b>
<b>Hoofdstuk 4: GUI.....</b>	<b>68</b>
<b>Hoofdstuk 5: Projectaanpak .....</b>	<b>69</b>
<b>Hoofdstuk 6: Database opzet.....</b>	<b>70</b>
<b>Hoofdstuk 7: Aanpassingen .....</b>	<b>71</b>
<b>Hoofdstuk 8: Database opzet.....</b>	<b>73</b>
<b>Bijlage 1: Modules.....</b>	<b>74</b>
<b>Bijlage 2 Functionaliteiten .....</b>	<b>78</b>
<b>Bijlage 3: Database attributes.....</b>	<b>80</b>



## Hoofdstuk 1:       Introductie

### 1.1     Inleiding

Dit ontwerpdocument heeft betrekking op het project "KOOP". Het is geschreven in opdracht van DHV Water, door de stagiaire Erwin Hendriks.

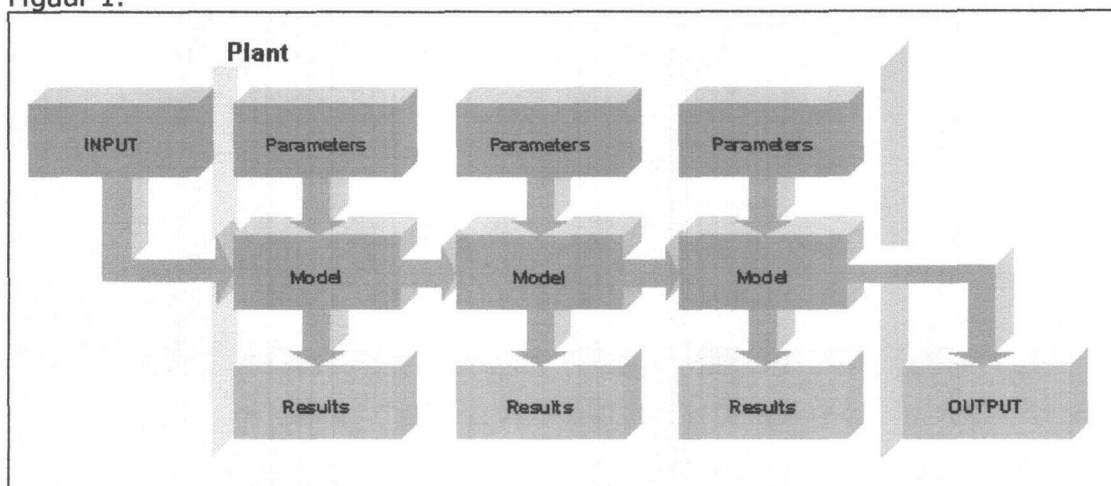
Het project KOOP staat voor Koppeling Onderzoek, Onderwijs en Praktijk voor drinkwater.

KOOP wordt een applicatie waarmee waterzuiveringen worden gesimuleerd. Elke gebruiker moet een bepaalde input (type water, zie figuur 1) geven aan de simulatie. Op dit type water wordt een waterzuivering toegepast. Elke waterzuivering (plant) is een aaneenschakeling van waterzuiveringsprocessen (modellen) en is in zijn complexiteit instelbaar. De modellen worden door professionals en / of promovendi opgesteld en aangepast.

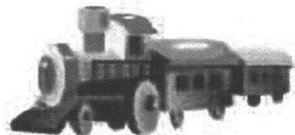
Na het simuleren van een waterzuivering krijgt de gebruiker een presentatie van alle tussen resultaten van het water na het toepassen van elk model, en het behaalde eind resultaat.

De essentie van de applicatie is dat gebruikers over de gehele wereld deze berekeningen kunnen uitvoeren.

Figuur 1.



De opzet van KOOP is dat modelmakers op een gemakkelijke manier modellen kunnen plaatsen en aanpassen op de applicatie koop. Dit is aan de hand van het onderstaande tekening (trein) uitgelegd.



De applicatie KOOP is te vergelijken met een modeltrein waar iedere modelmaker zijn eigen wagon (model) achter kan plaatsen. Wel moet de wagon aan bepaalde eisen (properties) voldoen, zodat het aangekoppeld kan worden aan de trein.

Voor maken van een ontwerpdocument is het nodig om de wensen van de opdrachtgever duidelijk in kaart te brengen. Er is hieronder een onderscheid gemaakt in verschillende aspecten van het functioneel ontwerp. Hierbij gaat het om: de gebruikers (wat mogen/kunnen de gebruikers), de sitemap en de aanpak van het project.

## **1.2 Doel**

Het doel van dit ontwerpdocument is het inzichtelijk krijgen van de functionele specificaties waar de applicatie KOOP aan moet voldoen. Hierbij is de centrale vraag: Wat zijn de gebruikersmogelijkheden, waar bevinden die zich en wat gebeurt er dan. Ook wordt de manier waarop in het kort besproken.

## **Hoofdstuk 2: De gebruikers**

### **2.1 De modelbouwer**

De gebruikertype "modelbouwer" is aangemaakt om personen autorisatie te geven voor het aanmaken van een model, plant of watertype.

Hoofdstuk 3 geeft een grafische weergave van de gebruikersmogelijkheden van de modelbouwer.

De modelbouwer krijgt alle lees en schrijfrechten voor het opmaken van een eigen model, plant en water. Tevens kan de maker van een bepaald model, plant of watertype toegangsrechten toekennen, zoals: leesbaar/aanpasbaar voor alle modelbouwer(s) en uitvoerrechten voor gebruikers. Door deze mogelijkheid kunnen modellen, plants en watertypen die nog in aanmaak zijn onzichtbaar gemaakt worden voor andere gebruikers.

De modellen, plants en watertypen moet wel voldoen aan bepaalde randvoorwaarden, deze voorwaarden zullen in de technische documentatie nader worden behandeld.

### **2.2 De modelgebruiker**

De modelgebruiker heeft als doel het inzichtelijk maken van de toepasbaarheid van modellen en/of plants onder een bepaalde toestand van het water (input).

Hoofdstuk 3 geeft een grafische weergave van de gebruikersmogelijkheden van de modelgebruiker.

De modelgebruiker kan zelf een bepaalde input water aanmaken. Op het water zal hij of zij een bepaalde plant kunnen toepassen. Deze plant is door de gebruiker zelf op te zetten door het selecteren van modellen. De modellen zijn **niet** aanpasbaar door een modelgebruiker.

Na het opzetten van een plant en het maken van een water kan de gebruiker de opzet simuleren. De resultaten van de simulatie krijgt de gebruiker gepresenteerd.

### **2.3 De systeembeheerder**

De systeembeheerder is voornamelijk voor het onderhoud van de applicatie KOOP. De grafische weergave is weergegeven in Hoofdstuk 3.

De systeembeheerder kan verschillende berichten op de homepage (startpagina) zetten. Hierbij denkende aan het lanceren van nieuwe modellen of plants. Melding voor het onderhoud aan de web server, etc.

Daarnaast heeft de systeembeheerder de mogelijkheid gebruikers te verwijderen, autorisatie aan te passen en nieuwe gebruikers aan te maken.

Een bepaald model moet voldoen aan bepaalde randvoorwaarden. Deze voorwaarden worden beheerd met Parameterbeheer. De systeembeheerder heeft de mogelijkheid parameters van de modellen te veranderen. Hierbij denkende aan de eenheid, default waarde en de omschrijving van een parameter.

Model / plant / water beheer zal nodig zijn voor het verwijderen van verouderde of onjuist modellen.

Historie beheer zal nodig zijn omdat alle berekende data wordt opgeslagen. Naargelang de applicatie gebruikt zal worden, zal ook de historische data zich opstapellen. D.m.v. het verwijderen van verouderde data zal de performance van de database toenemen.

### **2.4 De niet geregistreeerde gebruiker**

Een gebruiker die nog niet heeft ingelogd moet ook een mogelijkheid hebben verschillende composities uit te voeren. Deze composities zijn van tevoren gedefinieerd en zijn niet aanpasbaar door deze gebruiker.

Deze gebruiker krijgt ook de mogelijkheid de eigenschappen van de processen en plants en het water te zien (zie Hoofdstuk 3).

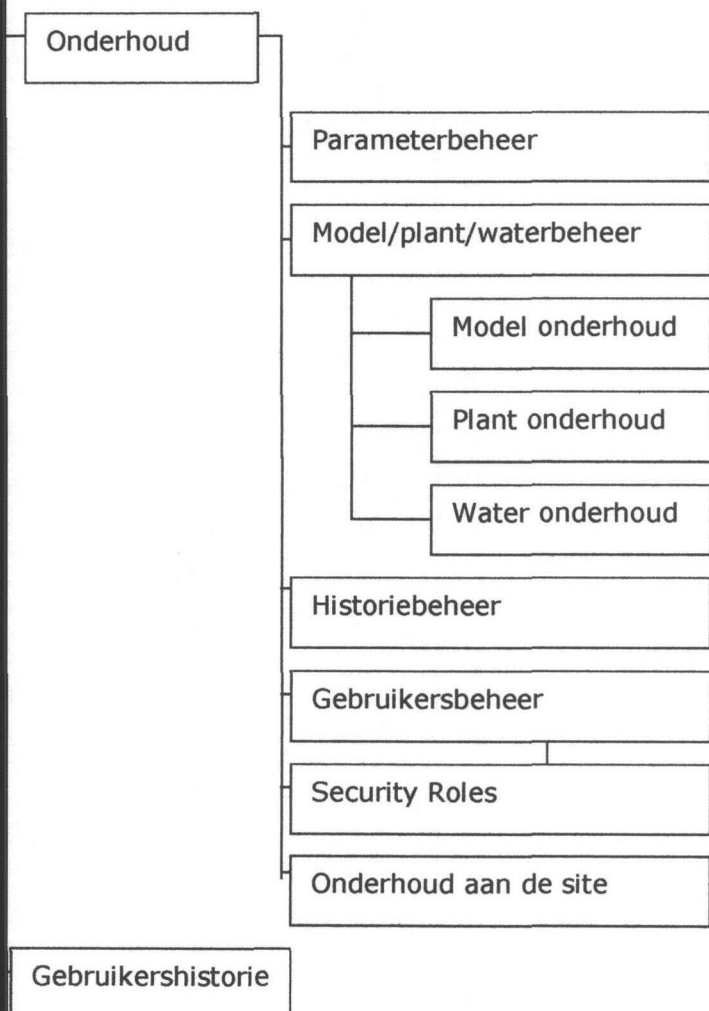
Door deze mogelijkheid heeft de gebruiker een beeld wat deze applicatie allemaal kan, waardoor de drempel tot registreren kleiner kan worden.



### Hoofdstuk 3: Opzet sitemap

		Toegang tot de modules			
		Model bouwer	Model gebruiker	Systeem beheerder	Niet geregistreeerde gebruiker
Home		X	X	X	X
	Aanpassen homepage	-	-	X	-
Login / Logout		X	X	X	X
	Registreren	X	X	X	X
Model		X	X	X	X
	Geef model parameters weer	X	X	X	X
	Verander model parameters	X	-	X	-
	Opzetten nieuw model	X	-	X	-
Plant		X	X	X	X
	Geef Plant compositie weer	X	X	X	X
	Verander Plant compositie	X	X	X	-
	Opzetten nieuwe plant	X	X	X	-
Water		X	X	X	X
	Geef water properties weer	X	X	X	X
	Verander water properties	X	X	X	-
	Opzetten nieuw water	X	X	X	-
Execute		X	X	X	X

## Vervolg modules



## Vervolg toegang tot de modules

Model Bouwer	Model Gebruiker	Systeem Beheerder	Niet geregistreeerde gebruiker
-	-	X	-
-	-	X	-
-	-	X	-
-	-	X	-
-	-	X	-
-	-	X	-
-	-	X	-
-	-	X	-
-	-	X	-
X	X	X	-

De hoofdmodules zijn Model, Plant, Water en execute. De modules Model, Plant en water zijn hoofdzakelijk voor het aanpassen van de eigenschappen of om een nieuwe opzet te maken. De module execute is voor het simuleren van het geselecteerde water en plant. Waarbij de resultaten worden weergegeven op het scherm.

Voor de uitvoering van dit project wordt gebruik gemaakt van een portal. Dit is de portal DotNetNuke. De grafische presentatie van de KOOP modules zal worden aangepast aan de presentatie van DotNetNuke.

Voor meer informatie zie bijlage 1.

## Hoofdstuk 4: GUI

Het programma zal worden gebruikt om waterzuiveringen te simuleren. Hierdoor is het logisch om de presentatie voornamelijk in het lichtblauw uit te voeren. Een voorbeeld van de applicatie presentatie is hieronder weergegeven.

The image shows two panels of a GUI. The top panel, titled 'Basic information', contains a circular button labeled 'Busv' on the left. To its right, there are three rows of text: 'Name Plant' with the value 'TestPlant', 'Location' with 'Amersfoort', and 'Date / Time' with '1 oktober 2003 11:30'. The bottom panel, titled 'Module A', contains two text input fields labeled 'Tekst 1' and 'Tekst 2'. Below these fields are three buttons labeled 'Save', 'Back', and 'Forward'.

Tijdens de communicatie met de database is het mogelijk dat de gebruiker te lang moet wachten zonder enig zichtbaar resultaat. Tijdens het uitvoeren van handelingen zal er een indicator komen die weergeeft of het programma bezig is of niet.

Het lettertype zal worden vastgesteld op:

### Normale teksten:

Font type: Verdana  
Font grootte: 10  
Font style: Normal

### Kop teksten:

Font type: Verdana  
Font grootte: 10  
Font style: bold

### Link teksten:

Font type: Verdana  
Font size: 10  
Font style: normal  
supplementary: underline

De positie waar de modules terechtkomen is aan de klant zelf te bepalen, hierin is het mogelijk de modules te verschuiven van positie.

## **Hoofdstuk 5:        Projectaanpak**

Voor het stageproject zijn bepaalde randvoorwaarden gesteld. Dit is het tijdsbestek waarin het project loopt en de personele bezetting. Hierdoor blijven de eisen waaraan het product moet voldoen het enige flexible aspect.

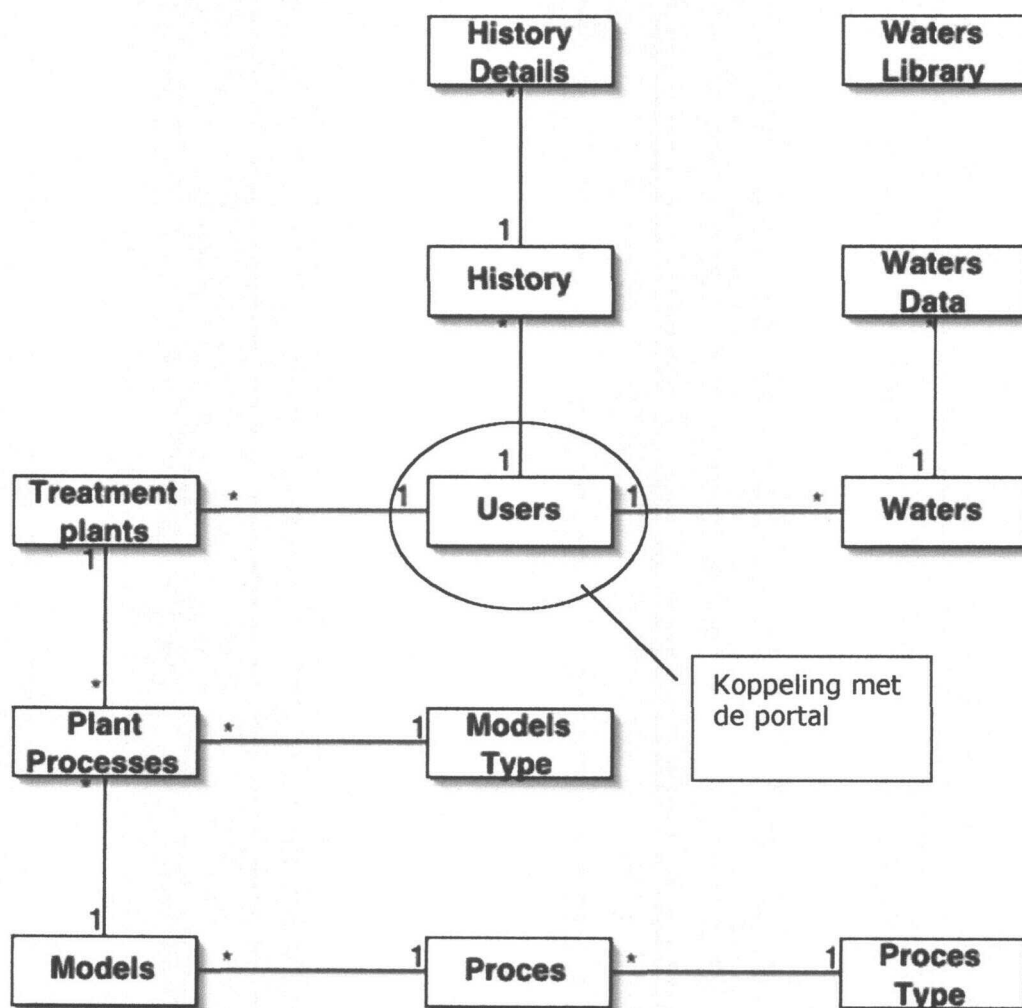
Door deze opzet is er gekozen voor een DSDM (dynamic systems development method) opzet, met behulp van een incrementplanning. De incrementen worden door middel van het MoSCoW principe toegepast (Must have, Should have, Could have en Would have).

Een overzicht van het MoSCoW principe is weergegeven in Bijlage 2 van dit document.



## Hoofdstuk 6: Database opzet

Hieronder is de applicatie database weergegeven. Voor een uitgebreide versie van de database, zie bijlage 3 (van dit document). Het gedeelte aan de linker kant van de tabel Users is voor het opslaan van de Plantgegevens en de hieraan gekoppelde processen. De rechterkant van de tabel Users is voor het opslaan van de waters van de gebruiker, met de bijbehorende parameters. Boven de tabel Users worden alle gesimuleerde data opgeslagen en is die door de gebruiker terug te lezen.



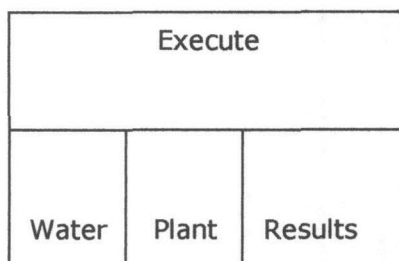
## Hoofdstuk 7: Aanpassingen

Na de presentatie van het prototype van KOOP is er veel commentaar gegeven op de GUI.

De presentatie is aangepast aan de hand van een mail gekregen van Ir. De Moel aan de hand van de terugkoppeling van de GUI opzet van KOOP.  
(6 november 2003).

Als hoofdmodule is er de module Stimela, dit is de overkoepelde module van Execute, Water en Plant. In deze module kan de gebruiker direct een simulatie uitvoeren, maar ook zijn of haar keuze aanpassen naar eigen inzichten.

Module Stimela



### Presentatie

#### Execute :

\$Dropdownbox Water	\$Dropdownbox Plant	Run
<u>Details</u>	<u>Details</u>	
New	New	

#### Waters

Water name	\$Watename	Update Save Delete
Description	\$Description	
Validator \$watename	Add Watersample	

Watersample	0	1
Time	\$31-01-2003 11:00:00	\$31-01-2003 11:00:01
Volume	\$2 m3	\$3 m3
Propertie1	\$ Propertie1	\$ Propertie1
Propertie2	\$ Propertie2	\$ Propertie2
Propertie3	\$ Propertie3	\$ Propertie3
etc		

## Plants

Plant name	<b>\$Plantname</b>	<a href="#">Update</a> <a href="#">Save</a> <a href="#">Delete</a>
Description	<b>\$Description</b>	
Validator \$Plantname	<a href="#">Add model</a>	

proces	<b>\$modeltype</b>	<b>\$Proces</b>	<b>\$ProcesType</b>
Proces 2	<b>\$modeltype</b>	<b>\$Proces</b>	<b>\$ProcesType</b>

## Results Simulation

Parameters	Input	Output	Result
Time	Input	Output	Green
Volume	Input	Output	Green
Parameter 1	Input	Output	Red
Parameter 2	Input	Output	Red
Parameter 3	Input	Output	Green
etc			

## Results Processes

Lijst van de processen

[View Proces input](#)

[View proces Output](#)

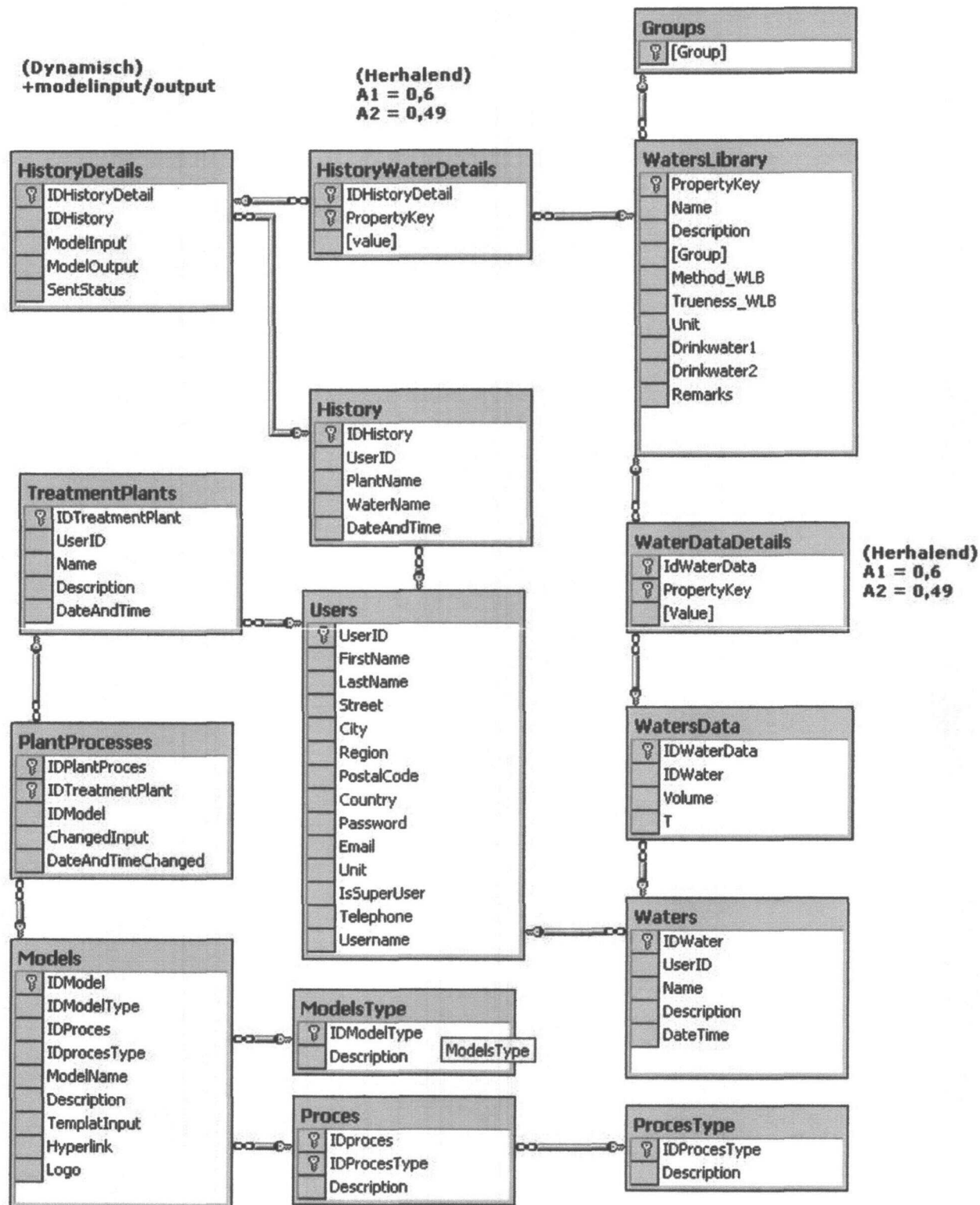
## Water Results

Input		output	Result
Time	Input	Output	Green
Volume	Input	Output	Green
Parameter 1	Input	Output	Red
Parameter 2	Input	Output	Red
Parameter 3	Input	Output	Green
etc			

Legenda:

Onderstreept is een link button  
\$ zijn variabele uit de database

## Hoofdstuk 8: Database opzet





## **Bijlage 1: Modules**

### **Bijlage 1.1 Schermfuncties**

#### **Bijlage 1.1.1 Inleiding**

In dit hoofdstuk wordt de functie van het scherm uitgelegd. Hierbij gaat het erom wat de gebruiker kan doen in het scherm. Hierin wordt niet behandeld wie er rechten hebben op deze pagina, zie hiervoor hoofdstuk 2.2 de gebruikers en naar bijlage 1 van dit document.

#### **Bijlage 1.1.2 Home**

De gebruiker komt binnen in het "home" gedeelte van de website. Hierop komt een stukje algemene tekst over de applicatie. Tevens kunnen hier nieuw modellen worden geïntroduceerd en overige mededelingen worden geplaatst, zoals aankondigingen serveronderhoud e.d.

##### **Bijlage 1.1.2.1 Aanpassen van de homepage**

In dit scherm is er een mogelijkheid de homepage aan te passen.

#### **Bijlage 1.1.3 Login/logout**

In dit scherm heeft de gebruiker de mogelijkheid zich in te loggen. Indien de gebruiker nog niet is geregistreerd, heeft diegene in dit scherm een mogelijkheid voor registratie.

##### **Bijlage 1.1.3.1 Registreren**

Hierin kan de gebruiker zich aanmelden bij KOOP. Daarbij zullen bepaalde persoonsgegevens worden gevraagd.

#### **Bijlage 1.1.4 Model**

In dit scherm "model" zal een algemeen overzicht gegeven van modeleigenschappen worden gehouden wat een model inhoud. Misschien wat algemene informatie, etc. Het ontwerpen van de tekst zal door de modelbouwer worden geschreven.

##### **Bijlage 1.1.4.1 Weergave modelparameters**

In dit scherm kan men een selectie maken van de, voor de gebruiker, beschikbare modellen. Na het kiezen van een model wordt er een de lijst met parameters van het gekozen model weergegeven. In dit scherm is tevens een link gemaakt voor het veranderen van het model.

##### **Bijlage 1.1.4.2 Wijzig modelparameters**

Voor het veranderen van de modelparameters moet eerst een selectie worden gemaakt van de beschikbare modellen. Na het selecteren van een model worden de parameters weergegeven en is naar wens te veranderen. Dit model is op te slaan onder de bestaande naam of onder een nieuwe naam.



#### Bijlage 1.1.4.3 Opzetten nieuw model

Voor het opzetten van een nieuw model krijgt de gebruiker een default invoerscherm waarin de naam en parameters zijn in te voeren en / of te veranderen.

#### Bijlage 1.1.5 Plant

In het scherm "plant" zal een algemeen tekst en overige informatie worden geplaatst met betrekking tot Plants.

##### Bijlage 1.1.5.1 Weergave plant compositie

Met deze functie kan de gebruiker de compositie van de plant zien. Dit zal worden weergegeven met een lijst met gebruikte modellen.

##### Bijlage 1.1.5.2 Wijzig plant compositie

Voor het veranderen van de plant compositie moet eerst een selectie worden gemaakt van de beschikbare plants. Na het selecteren van een plant wordt de compositie weergegeven, waarna deze is te veranderen. Deze plant is op te slaan onder de bestaande naam of onder een nieuwe naam.

##### Bijlage 1.1.5.3 Opzetten nieuwe plant

Voor het opzetten van een nieuwe plant krijgt de gebruiker een default invoerscherm waarin de naam van de plant moet worden weergegeven. Tevens krijgt deze een lijst met mogelijke modellen en dieptegraat waaruit de gebruiker kan kiezen.

#### Bijlage 1.1.6 Water

In het scherm "water" zal een algemeen tekst en overige informatie worden geplaatst die betrekking heeft op water.

##### Bijlage 1.1.6.1 Weergave waterproperties

In dit scherm kan er een selectie worden gemaakt van de, voor de gebruiker beschikbare, waters. Na het kiezen van een type water wordt er een lijst met properties van het gekozen water weergegeven. In dit scherm is tevens een link gemaakt voor het veranderen van het water.

##### Bijlage 1.1.6.2 Wijzig waterproperties

Voor het veranderen van de waterproperties moet eerst een selectie worden gemaakt van de beschikbare waters. Na het selecteren van een water worden de properties weergegeven en is dit naar wens te veranderen. Dit water is op te slaan onder de bestaande naam of onder een nieuwe naam.

##### Bijlage 1.1.6.3 Opzetten nieuw water

Voor het opzetten van een nieuw water krijgt de gebruiker een default invoerscherm waarin de naam van het water moet worden weergegeven. Tevens krijgt deze een lijst met properties die moeten worden ingevoerd.

#### Bijlage 1.1.7      Execute

In dit scherm kan de gebruiker een keuze maken uit een door de gebruiker aangemaakte type water en een type plant en de mate van complexiteit. Na deze keuzes kan de gebruiker deze simulatie draaien waarna de resultaten zullen worden gepresenteerd (zie bijlage 1.1.7.1).

#### Bijlage 1.1.7.1      Presentatie

Alle gegeven die worden gegenereerd door de Stimela Server, zijn op XML formaat gemaakt. In eerste instantie worden deze gegevens in een tabel gepresenteerd aan de gebruiker. Later moet het mogelijk zijn statische, dynamische en stochastische grafieken te genereren.

#### Bijlage 1.1.8      Onderhoud

Voor het optimaal presteren van de site en database moet er onderhoud gepleegd worden. In dit venster is te zien wanneer welk onderhoud is gepleegd. Bijvoorbeeld wanneer de historie voor het laatst is verwijderd. Wanneer voor het laatst de parameterbeheer is aangepast, etc.

#### Bijlage 1.1.8.1      Parameterbeheer

In deze pagina kan de beheerder de parameter definities aanpassen. Dit is om op een makkelijke manier eenduidigheid te creëren in de database.

#### Bijlage 1.1.8.2      Model / plant / waterbeheer

Hierin kan de beheerder oude modellen, plants en waters verwijderen op een simpele manier. Indien de beheerder op model (plant of water) drukt krijgt deze een lijst met de beschikbare modellen met de datum laatst gebruikt. Door een druk op de knop kan deze verwijderd worden.

#### Bijlage 1.1.8.3      Historiebeheer

Deze pagina is voor het verwijderen van historische gegevens, met hetzelfde principe als het model / plant / waterbeheer.

#### Bijlage 1.1.8.4      Gebruikersbeheer

Deze pagina is voor het verwijderen van oud gebruikers, met hetzelfde principe als het model / plant / waterbeheer.

#### Bijlage 1.1.8.5      Security Roles

Bij security roles is het mogelijk de rechten van de gebruiker aan te passen. B.v. van modelgebruiker naar modelbouwer.

#### Bijlage 1.1.8.6      Onderhoud aan de site

In dit onderdeel van de module kan de site onderhouden worden.

In dit scherm kan de gebruiker zijn historische berekeningen terug zien. De historische gegevens worden gelimiteerd tot 5 berekeningen per gebruiker. Na het selecteren van een historische berekening van de gebruiker zullen deze gegevens naar de "presentatie" (zie bijlage 1.1.7.1) worden doorgegeven.

## Bijlage 2 Functionaliteiten

Uitvoering	Aanwezig in de Portal	Is al verwerkt in de database	Nog realiseren
------------	-----------------------	-------------------------------	----------------

### MUST HAVE

Opzetten Login Procedure	X	X	-
Opzetten register Procedure	X	X	-
Opstellen van een nieuw model	-	-	X
Opslaan van Modelparameters in XML formaat	-	-	X
Opstellen van een nieuwe plantcompositie	-	-	X
Opstellen van een nieuw water	-	-	X
Opstellen van een simulatie	-	-	X
Per model versturen naar Stimela	-	-	X
Presenteren van de resultaten van de simulatie	-	-	X

### SHOULD HAVE

Weergeven modelparameters	-	-	X
Lijst van beschikbare modellen	-	-	X
Aanpassen van beschikbare modellen	-	-	X
Weergeven plantcompositie	-	-	X
Lijst van beschikbare plantcomposities	-	-	X
Aanpassen van beschikbare plantcomposities	-	-	X
Weergeven waterproperties	-	-	X
Lijst van beschikbare watertypes	-	-	X
Aanpassen van beschikbare watertypes	-	-	X
Modellen plants en waters verwijderen	-	-	X
Gebruikers verwijderen	X	X	-
Gebruikers autorisatie	X	X	-
Inleidende tekst homepage	-	X	X

### COULD HAVE

Online aanpassen van de homepage	X	X	-
Online aanpassen plant model en waterpage	X	X	-
Verwijderen van modellen plants en waters	-	-	X
Historie verwijderen door beheerder	-	-	X
Onderhoud aan de site	X	X	-

Weergeven gebruikershistorie	-	-	X
Weergeven van historie resultaten	-	-	X

#### WOULD HAVE

Foutcontroles	X	X	X
Veranderen van modelparameters	-	-	X
Verwijderen van historische gegevens door gebruiker	-	-	X
Link van login naar registreren	X	X	-
Encrypted password	-	-	X
Melding als registreren gelukt is	X	X	X
Gebruikersvoorwaarden	-	-	X
Model plant en water introtekst	-	X	X



### Bijlage 3: Database attributes

**Table: Users**

keys	Name	Data Type	Size (precision,Scale)	Allow Nulls	Default
pk	UserId	Int	4		
	FirstName	nvarchar	50		
	LastName	nvarchar	50		
	Street	nvarchar	50	√	
	City	nvarchar	50	√	
	Region	nvarchar	50	√	
	PostalCode	nvarchar	50	√	
	Country	nvarchar	50	√	
	Password	nvarchar	50		
	Email	nvarchar	100		
	Unit	nvarchar	50	√	
	IsSuperUser	Bit	1		0
	Telephone	nvarchar	50	√	
	UserName	nvarchar	100		

**Table: Treatmentplants**

keys	Name	Data Type	Size (precision,Scale)	Allow Nulls	Default
pk	UserId	Int	4		
	UserId	Int	4		
	Name	nvarchar	50		
	Description	nvarchar	50	√	
	DateAndTime	Datetime	8		

**Table: PlantProcesses**

keys	Name	Data Type	Size (precision,Scale)	Allow Nulls	Default
pk	IdPlantProces	Int	4		
pk	IdTreatmentplant	Int	4		
	IdModel	int	4		
	ChangedInput	nvarchar	100	√	
	DateAndTimeChanged	Datetime	8		

**Table: Models**

keys	Name	Data Type	Size (precision,Scale)	Allow Nulls	Default
pk	IdModel	Int	4		
	IdModelType	nvarchar	100		
	IdProces	nvarchar	100		
	IdProcesType	nvarchar	100		
	ModelName	nvarchar	50		
	Description	nvarchar	50	√	
	TemplatInput	nvarchar	100		
	Hyperlink	nvarchar	50		
	Logo	nvarchar	50	√	

**Table: ModelType**

keys	Name	Data Type	Size (precision,Scale)	Allow Nulls	Default
pk	IdModelType	Nvarchar	100		
	Description	text	16	√	

**Table: Proces**

keys	Name	Data Type	Size (precision,Scale)	Allow Nulls	Default
pk	IdProces	Nvarchar	100		
pk	IdProcesType	Nvarchar	100		
	Description	text	16	√	

**Table: ProcesType**

keys	Name	Data Type	Size (precision,Scale)	Allow Nulls	Default
pk	IdProcesType	Nvarchar	100		
	Description	text	16	√	

**Table: Waters**

keys	Name	Data Type	Size (precision,Scale)	Allow Nulls	Default
pk	IdWater	Int	4		
	UserId	Int	4		
	Name	Nvarchar	50		
	Description	Text	16	√	
	DateTime	DateTime	8		

**Table: WatersData**

keys	Name	Data Type	Size (precision,Scale)	Allow Nulls	Default
pk	IdWaterData	Int	4		
	IdWater	Int	4		
	Volume	int	4		
	T	Datetime	8		
	Properties (250)	real	8	√	

**Table: WatersLibrary**

keys	Name	Data Type	Size (precision,Scale)	Allow Nulls	Default
pk	PropertyKey	Nvarchar	50		
	Name	nvarchar	50		
	Description	nvarchar	50	✓	
	Group	nvarchar	50		
	Method_WLB	nvarchar	50	✓	
	Trueness_WLB	nvarchar	50	✓	
	Unit	nvarchar	50	✓	
	Drinkwater1	Int	4	✓	
	Drinkwater2 (default)	Int	4	✓	
	Remarks	nvarchar	50	✓	

**Table: History**

keys	Name	Data Type	Size (precision,Scale)	Allow Nulls	Default
pk	IdHistory	Int	4		
	UserId	Int	4		
	PlantName	Nvarchar	50		
	WaterName	Nvarchar	50		
	DateAndTime	DateTime	8		

**Table: HistoryDetails**

keys	Name ↓	Data Type	Size (precision,Scale)	Allow Nulls	Default
pk	IdHistoryWater	Int	4		
	IdHistory	Int	4		
	Volume	int	4		
	T	Datetime	8		
	ModelInput	narchar	100		
	ModelResult	narchar	100		
	Properties (250)	real	8	✓	

Hogeschool  
van Utrecht

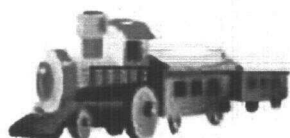


Advies- en Ingenieursbureau

Technisch ontwerp

# KOOP

Koppeling Onderzoek, Onderwijs en Praktijk



## Bijlage 3

Student:  
Studentnr:  
Eerste examiner:  
Stagebedrijf:

E.N.J. Hendriks  
1106061  
H. Plessius  
DHV Water



## **Samenvatting**

In dit document wordt er uitgelegd waarom voor sommige technieken worden gekozen. Voor het maken van de applicatie KOOP is er gekozen voor Windows 2000 server met IIS v 5.0, SQL server, programmeertaal ASP.Net met Visual basic als server-scripting taal.

De applicatie zal worden gemaakt met de ontwikkeltool Visual studio.net.

Als applicatie ondersteuning is gekozen voor de portal DotNetNuke, de reden hiertoe was: Veel basis functionaliteiten als waren voorgeprogrammeerd; Mogelijkheid tot uitbreiden met eigen modules waren aanwezig; mooie en dynamische user-interfase.



## Inhoudsopgave

<b>HOOFDSTUK 1. INTRODUCTIE .....</b>	<b>86</b>
1.1 INLEIDING .....	86
<b>HOOFDSTUK 2. GEKOZEN TECHNIEKEN .....</b>	<b>87</b>
<b>HOOFDSTUK 3. PRESENTATIE .....</b>	<b>91</b>
3.1 ALGEMEEN .....	91
3.2 DE PRESENTATIE VAN "KOOP" .....	91
3.2.1 <i>De basis - Portal</i> .....	91
3.3 DE OPZET .....	92
3.4 PLANT COMPOSER MODULE.....	93
<b>HOOFDSTUK 4. DATABASE .....</b>	<b>96</b>
<b>HOOFDSTUK 5. AANPASSINGEN .....</b>	<b>97</b>
<b>HOOFDSTUK 6. LITERATUURLIJST .....</b>	<b>98</b>
<b>Bijlage 1: Databaseopzet.....</b>	<b>99</b>

## **Hoofdstuk 1. Introductie**

### **1.1 Inleiding**

In dit hoofdstuk is beschreven hoe de applicatie er technisch uitziet.

Hierbij zal duidelijk worden beschreven wat voor gegevens van de gebruiker gevraagd worden, wat voor gegevens er onderling verstuurd worden, en hoe de uitkomst gepresenteerd wordt. Daarnaast wordt er beschreven wat voor technieken er gebruikt worden voor het creëren van de applicatie.

Er zal aandacht worden besteed waarom er voor de desbetreffende technieken gekozen is en wat voor opties er overwogen zijn.

De paragrafen van het technisch ontwerp zijn als volgt ingedeeld:

Technieken  
Presentatie  
Veranderingen na evaluatie  
Literatuur

Er zijn een aantal technieken beschikbaar voor het maken van applicaties. In dit hoofdstuk is beschreven welke technieken er gebruikt worden voor het maken van de applicatie KOOP.

De eisen die gesteld zijn aan de toe te passen techniek zijn:

De applicatie moet cliënt-site uit te voeren zijn. Hierbij zijn geen eisen gesteld aan het type browser waarin de applicatie moet worden weergegeven.  
Een logische opstelling van verschillende modules, waarbij deze de data kunnen opslaan, verwerken en aan de gebruiker presenteren.  
De onderdelen van de applicatie moet simpel uitbreidbaar zijn naar een ander type database en / of rekenmodule.

De "verplichte" technieken:

Windows 2000 server, IIS v.5.0  
Visual Studio.Net – voor het gebruik van .NET scripting  
SQL database

## Hoofdstuk 2. Gekozen technieken

### Keuze van de Programmeertaal

- mogelijkheden

Bij de keuze van een programmeertaal wordt met verschillende aspecten rekening gehouden. Deze zijn o.a. de kennis van de stagiair en/of programmeur; de ondersteuning van de cliënt omgeving (browsers); Ondersteunende hulpmiddelen door b.v. forum, helpfiles, ontwerpomgeving, e.d.; psychologisch aspect van de programmeur (de eigen mening van de programmeur over de programmeertaal); Keuze van de opdrachtgever; Intentie van de website (database ondersteund); Keuze van het operation system.

Ondersteuning door het OS	Intentie website	Psychologisch aspect	Hulpmiddelen bij stagebedrijf	Ondersteunende cliënt omgeving	Kennis stagiair	Kennis begeleider	Programmeertaal	
Windows 2000 + IIS 5.0	Communicatie SQL i.c.m. XML	+	VS.Net MSDN	+			ASP.Net I.c.m. XML	Keuze opdrachtgever
+	+/-	+	onbekend	+	7	8	Asp	
							Asp.Net	
+	+	+	VS.Net+MSDN	+	6	6	- VB	
+	+	-	"	+	5	?	- J#	
+	+	+	"	+	1	?	- C#	
+	+	-	"	+	1	?	- C++	
+/-	+/-	-	onbekend	+	5	6	Php	
+/-	+/-	-	onbekend	+/-	6	?	Java	

- besluit

De keuze van de programmeertaal stond aan het begin van het project al vast. De keuze van de opdrachtgever was dat de applicatie gemaakt moest worden in een dot net omgeving. De reden hiervoor was het geïntegreerde XML mogelijkheden. Daarbij kan de gebruiker gebruik maken van zijn eigen keuze programmeertaal, zoals visual basic, Java en C.

Het nadeel van deze programmeertaal is dat het prototype gemaakt is in Asp – Visual basic scripting. Indien de keuze hierop was gevallen, konden er veel onderdelen hergebruikt worden. Een tweede nadeel is dat de kennis van ASP.Net minder is dan de kennis van ASP. Hierdoor kan de doorlooptijd voor de realisatie vergroten.

### Keuze van de database

- Mogelijkheden

In het prototype is gebruik gemaakt van een MS Access database, de reden daartoe was de simpliciteit en aanpasbaarheid. Aangezien deze database niet gemaakt is voor multi-user applicaties met meer dan 25 gebruikers en 1 van de minst snelle is raad ik af deze in gebruik te nemen.

Voor een Linux operation system wordt aangeraden gebruik te maken van mySQL, die bekend staat om zijn multi-user mogelijkheden (+/- 32.000 users) en zeer hoge opslagmogelijkheid (terabyte).

Voor een Windows operation system wordt aangeraden gebruik te maken van SQLServer, het voordeel hiervan is gelijk aan die van mySQL alleen heeft de opslagcapaciteit een maximale van 2 Gigabyte

- Besluit

De opdrachtgever heeft besloten de applicatie te draaien op een windows operation system. Hierdoor is het besluit gevallen op SQLServer. De maximale opslagcapaciteit van 2 Gigabyte moet theoretisch geen problemen geven.

- Aanbeveling

Indien er toch buiten verwachting om de opslagcapaciteit een probleem zal vormen, moet dmv een eenvoudige verandering aan het script een verwijzing naar een database op een andere locatie benaderbaar worden. (verwijzing is te vinden in Web.config in de root van DotNetNuke)

## **Keuze platform**

- Mogelijkheden

Voor een platform kan men kiezen uit een aantal verschillende mogelijkheden.

Zoals:

- Windows gebaseerde platformen (zoals Windows XP; Windows 2000; Windows 98; etc.)
- Linux gebaseerde platformen (Redhead, Suse, etc.)
- Mac gebaseerde platformen (OS-X, OS-9)
- Gecombineerde platformen d.m.v. een emulator
- Gecombineerde platformen d.m.v. meerdere servers

- Besluit

Het besluit om Windows 2000 te gebruiken is grotendeels gekomen door het gebruik van Windows 2000 op DHV intern. Ook het gebruik van Visual studio.Net is een reden voor het gebruik van een Windows platform.

Een Mac gebaseerde omgeving wordt niet aangeraden. Door de geringe keuze aan databases (ASCII-based en Filemaker) en het gebrek aan een goede omgeving, maakt dit het programmeerwerk niet makkelijker.

Gecombineerde platformen d.m.v. een emulator is mogelijk, hoewel de performance zo'n hoedanigheid vermindert dat het niet aan te raden is.

- Aanbeveling

Op dit moment is er ook gekozen voor een Windows gebaseerde database. Hierbij is het mogelijk om op dezelfde server de database te plaatsen. Indien er wordt gekozen voor een stabielere omgeving voor de database wordt Linux (Redhead) aangeraden (zie Test Computer totaal September 2003).

Gecombineerde platformen d.m.v. meerdere servers is op zich geen slecht idee. Het voordeel hiervan is dat elk onderdeel zijn eigen processor heeft. Hierdoor kan er veel tijd gewonnen worden. Een nadeel is dat er meer machines en koppelingen nodig zijn, en dus meer kans op gebreken.



## **Keuze portal**

- Mogelijkheden

Voordat er een keuze is gemaakt welke portal er het meest geschikt is dit project, is hieronder een lijst weergegeven van de onderzochte portals:

- Portal solidario
- AspApp
- APG
- 4dPortal
- FullXML
- Crystal Space 3d SDK
- Liferay
- Yacoom
- Rainbow
- GFS 3.2
- Dotnetnuke
- Dotnetnuke XXL

- Besluit

Als beste is de portal (dnn) Dotnetnuke uit de test gekomen (zie verslag Portals). Hierbij zijn de plaatsen verdeeld tussen dnn en dnn xxl.

## Hoofdstuk 3. Presentatie

### 3.1 Algemeen

In het functioneel ontwerp is er een sitemap gemaakt waarop de verschillende site staat aangegeven waarbinnen de gebruiker kan navigeren. In dit hoofdstuk worden deze sites op het technische aspect nader bekeken. Dit houdt in dat er een grafische voorstelling wordt gemaakt hoe het eruit komt te zien.

### 3.2 De presentatie van "KOOP"

De website is opgezet vanuit de portale basis DotNetNuke. In deze basis bevinden zich al standaard functionaliteiten, zoals sitebeheer en autorisatiebeheer. Tevens bezit de portal verschillende onderdelen die te plaatsen zijn in het portalframe, afhankelijk van de wensen van de beheerder.

#### 3.2.1 De basis - Portal

De portalbasis is opgedeeld in drie hoofdonderdelen, namelijk het kop, hoofd en het voetgedeelte. Het kopgedeelte is het belangrijkste van de portal. Hierin bevinden zich alle tabmogelijkheden zoals beheer en registratie. Ook is hierin de login functie verwerkt. Het hoofdgedeelte is voor de geautoriseerde aanpasbaar naar eigens inziens. Hieruit kan de gebruiker kiezen uit verschillende modules. Een opsomming van de modules die zichtbaar worden d.m.v. een dropdown box zijn:

- Announcements (toevoegingen van mededelingen)
- Banners (toevoegen logo's)
- Contacts (bekijken van contacten)
- Discussions (discussieforums)
- Documents (voor het opslaan en afhalen van documenten)
- Events (Melding wanneer iets staat te gebeuren)
- FAQ ( Frequently asked Questions)
- Feedback (door de gebruiker te geven feedback aan systeembeheer)
- HTML (html script)
- Iframe (kan een andere site worden geopend in de portal)
- Images (toevoeging van plaatjes)
- Links (maken van links)
- Maps (Weergeven van een bepaald adres)
- News(ontvangen van nieuwsberichten)
- Search (zoeken van informatie binnen de portal)
- Userdefinedtable (maken van tabellen)
- Vendors (weergeven van vernoten)
- Weather (het weerbericht)
- XML (mogelijke toevoeging van een xml model)

In het voetgedeelte staat een vermelding van de gebruiker. Dit mag volgens het copyright naar wens worden aangepast.

### **3.3 De opzet**

Volgens bijlage 1 zijn de product specifieke eigenschappen zoals het maken van een plant en het opzet van een type water nog niet mogelijk zonder aanpassingen in de programmatuur.

Er wordt een viertal modules toegevoegd ter behoeve van het doel van de applicatie. De vier modules zijn:

- Model Composer Module (voor het opzetten van modules)
- Plant Composer Module (voor het opzetten van plants)
- Water Composer Module (voor het opzetten van waters)
- Execute Module (Uitvoeren van een simulatie)

In een later stadium worden er onderhoudsmodules toegevoegd. Deze worden in een later stadium behandeld.

### 3.4 Plant Composer Module

De Plant composer is opgedeeld in verschillende panels, hiervoor is gekozen om het overzicht te houden en gemakkelijk stukken onzichtbaar te maken.

- Basic information
- New Plant
- Alter Plant
- Models view
- Add Models
- Edit XML

De plant composer is opgedeeld in 3 lagen, dit zijn:

Presentatie laag	Business laag	Data laag
PlantComposer.ascx	PlantComposer.ascx.vb	PlantComposerDB.vb
Met de Presentatie laag wordt de GUI bedoeld. De GUI is de manier waarop het systeem zich aan de gebruiker presenteert.	Bij de business-laag moet de applicatie-logica (c.q rekenregels en business gegevens) voorgesteld worden.	De data-laag zorgt voor communicatie met de database.

#### Basic information view:

Hierin staat de actieve Plant van de Gebruiker

#### New Plant:

Dit panel is voor het aanmaken van een nieuwe Plant:

Hierbij wordt in de database een nieuw record aangemaakt bij tabel TreatmentPlants. Met als variabele waarde de plant naam en de plant omschrijving.

DateAndTime worden automatisch meegegeven. UserId is de waarde van de User na inloggen.

SQL Statement: <procedure dbo.CreatePlant>

#### Alter Plant

Dit panel is voor het veranderen van de Plant omschrijving zonder de modellen te verwijderen. Hierbij wordt de plant aangepast met een nieuwe naam en omschrijving. De waarde van DateAndTime blijft op de oude waarde staan.

SQL Statement: < Procedure AlterPlant>

#### Models view

In dit panel wordt een weergave gemaakt van de modellen gekoppeld aan de plant. Dit wordt mogelijk gemaakt door de functie ASP:Grid. In de grid wordt de ModelNaam, omschrijving, DateAndTimeChanged en de ChangedInput weergegeven.

SQL Statement: <Procedure GetModelsGrid>



## Add Models

In dit panel worden de, voor de gebruiker beschikbare, modellen weergegeven. Als eerste kan men kiezen uit een moeilijkheidsgraad (ModelType) waarna de gebruiker kan kiezen uit een type proces. Na het selecteren van deze waarde verschijnt de ProcesTypes. De combinatie van deze selecties geeft het model weer. Na het kiezen van het model is er de mogelijkheid om de bijbehorende XML waarde te veranderen.

SQL Statement: <Procedure AddModel>

## Edit XML

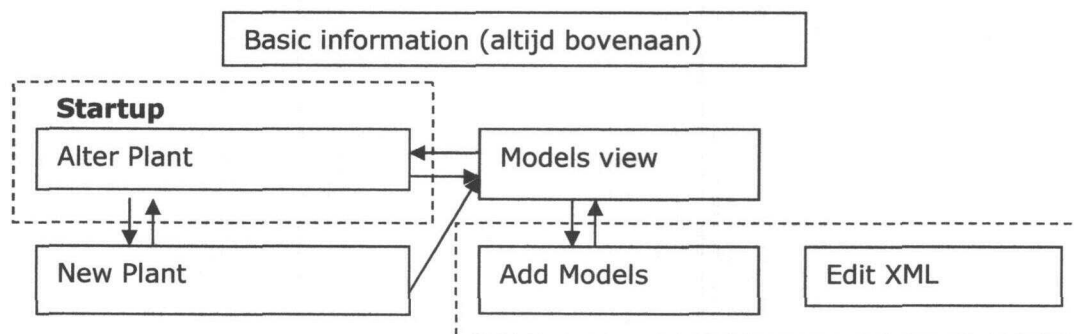
Hierin wordt alleen een tekstveld weergegeven, deze wordt uitgelezen bij procedure AddModel.

## Classoverzicht PlantComposerDB

Het onderstaande overzicht geeft weer welke operaties door de module PlantComposer worden uitgevoerd.

<b>Class:</b> PlantComposerDB
<b>Attributes:</b> Alle invoervelden van de GUI
<b>Operations:</b> CreatePlant (UserId, Name, Description, DateAndTime, IdTreatmentplant) GetPlantList (UserName) GetPlantProperties (UserId, PlantDropDownList) GetModelsGrid (IdTreatmentplant) GetModelType () GetProces (ModelType) GetProcesType (ModelType, Proces) GetXML (ModelType, Proces, ProcesType) AddModel (IdTreatment, ModelType, Proces, ProcesType, XMLMem)

## Panel opzet



### Andere modules

De opzet van de andere modules zijn nog niet bekend, hiervoor wordt eerst de reactie afgewacht van de opdrachtgever. Indien deze hierop aanmerkingen heeft kan deze worden doorverwerkt in de andere module.

## **Hoofdstuk 4. Database**

De performance van de applicatie is bijna net zo belangrijk als het consistent opslaan van gegevens. Hierbij is het belangrijk dat de gebruiker op de hoogte wordt gesteld dat er database - activiteiten zijn of niet.

Tevens is de duur van het uitvoeren van een query van belang. Getracht wordt het tijdspad van een query gemiddeld niet langer dan 500 miliseconden te laten duren. Deze wordt gecontroleerd door de door SQL meegeleverde SQL Profiler. Hiermee kan de tijd per query worden gepresenteerd.

De communicatie van en naar de Stimela-server zal niet worden vastgelegd. Dit omtrent de veranderende bandbreedte binnen het bedrijf DHV en de performance van Stimela-server.

SQL server beschikt zelf over een record locking systeem (alleen met schrijven, blijft wel altijd leesbaar). Hierdoor kunnen meerdere gebruikers tegelijkertijd in de zelfde tabel actief zijn. Wel wordt getracht alle transacties uit te voeren met een rollback. Hierdoor worden alle acties terug getrokken binnen de transactie.

## Hoofdstuk 5. Aanpassingen

Na de presentatie van KOOP van de eerste increment, zijn er veel dingen veranderd op het gebied van de GUI. Deze heeft ook veel invloed op techniek achter de applicatie. Nu is namelijk een globale module stimela ontworpen die de module Plant, Water en execute in zich heeft. (voor presentatie, zie functioneel ontwerp). De opzet van de Model module wordt in dit stadium nog buiten scope gelaten.

### Execute

Class ExecuteDB
Attributes \$userId -WaterId -PlantId
ExeGetWatersList(UserId) ExeGetPlantsList(UserId) Simulate(WaterId,PlantId) Create XML Extract XML

### Water

Class WaterDB
Attributes \$userId -WaterId
WtrGetWaterDescription(WaterId) WtrGetWaterDetails(WaterId) Add(WaterName,WaterDescriptions) Update(WaterName, WaterDescriptions, WaterId) Delete(WaterId) AddWater(WaterId)

### Plant

Class PlantDB
Attributes \$userId -PlantId
PLTGetPlantDescription(PlantId) PLTGetPlantDetails(PlantId) PLTGetModelType(ProcesType, Proces) PLTGetProces(ProcesType) PLTGetProcesType() Add(PlantName, PlantDescription) Update(PlantName, PlantDescription, WaterId) Delete(PlantId) AddProces(PlantId)



## Results

Class ResultsDB
Attributes
\$userId
-PlantId
-WaterId
-IdHistory
-IdHistoryDetail
-IdModel
ResShowResults(IdHistory)
ResShowProces(IdHistory, IdModel)

## Database opzet

De vernieuwde databaseopzet staat weergegeven in "Hoofdstuk 7: Aanpassingen" van het functioneel ontwerp.

## Hoofdstuk 6. Literatuurlijst

Boeknaam: ASP.NET  
Schrijver: J. Martin & B. Tomson  
ISBN#: 90-430-0539-8

Boeknaam: XML  
Schrijver: C. Ashbacher  
ISBN#: 90-430-0420-0

Boeknaam: het SQL Leerboek 5<sup>e</sup> uitgave  
Schrijver: R.F. van der Lans  
ISBN#: 90-395-0755-4

## Websites

Meerwaarde DSDM  
<http://www.computable.nl/artikels/archief2/d51ra2lo.htm>

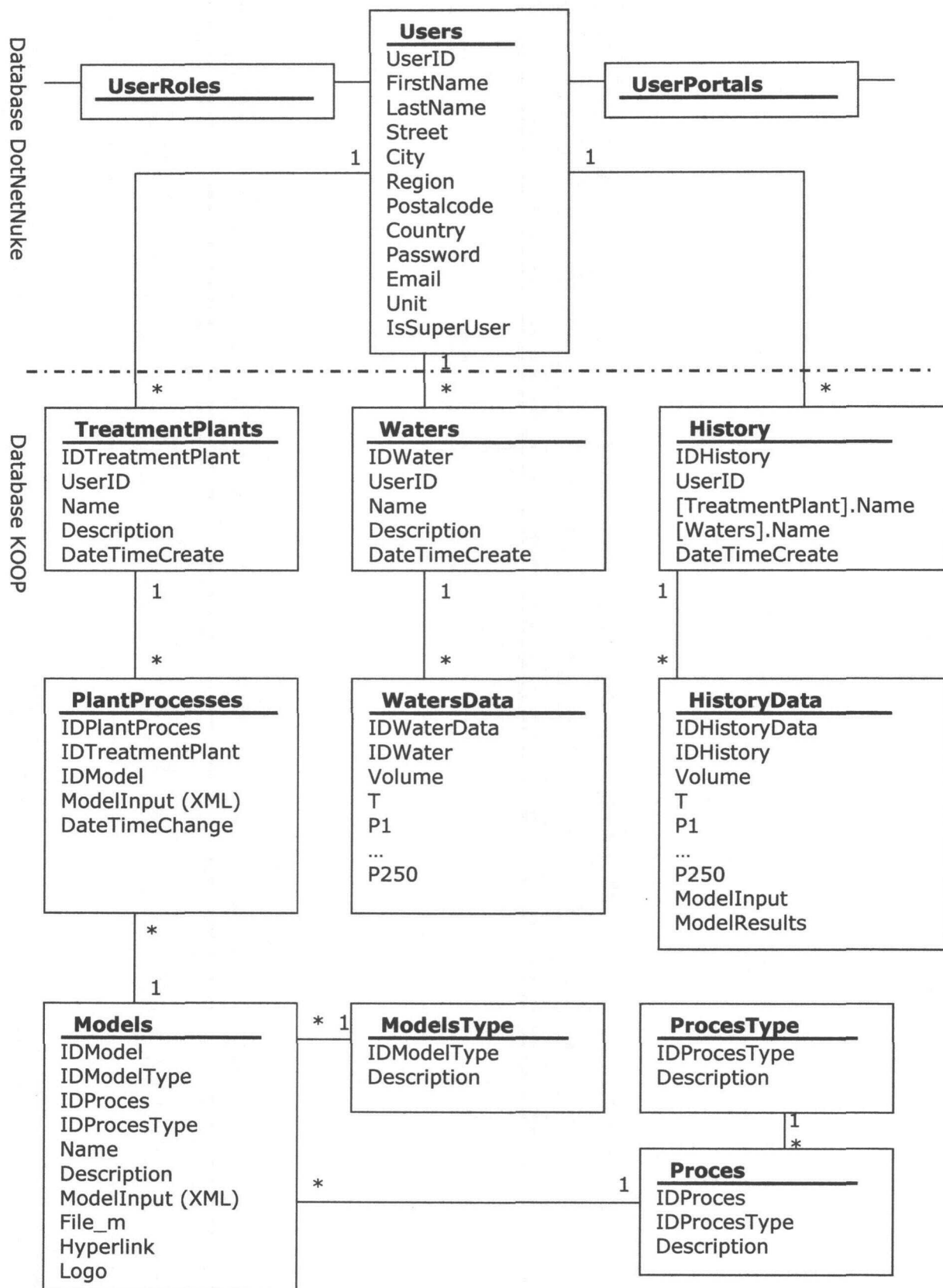
Prince 2 i.v.m. DSDM  
[http://www.array.nl/release/artikelen/2002/2\\_Leeuw\\_Prince2.pdf](http://www.array.nl/release/artikelen/2002/2_Leeuw_Prince2.pdf)

Tutorial website  
<http://www.w3schools.com/>

Modules voor website dotnetnuke  
<http://sourceforge.net/search/>  
<http://www.lucaslabs.net/DesktopDefault.aspx?tabid=95>

Dotnetnuke database opzet  
<http://www.dynst.com/DesktopDefault.aspx?tabid=78>

Bijlage 1: Databaseopzet



Hogeschool  
van Utrecht

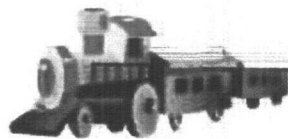


Advies- en Ingenieursbureau

Portals

# KOOP

Koppeling Onderzoek, Onderwijs en Praktijk



## Bijlage 4

Student:  
Studentnr:  
Eerste examiner:  
Stagebedrijf:

E.N.J. Hendriks  
1106061  
H. Plessius  
DHV Water

## Inhoudsopgave

<b>Hoofdstuk 1. Wat is een portal</b>	<b>102</b>
1.1 In het kort.....	102
1.2 Uitleg portals.....	102
<b>Hoofdstuk 2. Onderzoek portals</b>	<b>103</b>
2.1 Algemeen.....	103
2.2 Portal Uitleg.....	103
2.3 Portal specificaties.....	105
<b>Hoofdstuk 3. Conclusie</b>	<b>107</b>
3.1 Inleiding.....	107
3.2 De beste portal .....	107
Bijlage 1 .....	108



## **Hoofdstuk 1. Wat is een portal**

### **1.1 In het kort**

In het algemeen worden de volgende eigenschappen toegeschreven aan een portaal:  
Het betreft een startpagina, in de vorm van een unieke URL voor elke gebruiker binnen een organisatie, waarin kunnen zijn opgenomen:

- Links naar interne of externe www-sites (sites die informatie aanbieden), en
- Links naar bestaande applicaties (transactieverwerkende systemen)
- De sites en applicaties zijn met een muisklik oproepbaar
- De gebruiker hoeft maar 1 keer in te loggen (single sign on).
- Personalisatie: De gebruiker kan zijn of haar startpagina naar eigen smaak inrichten (items toevoegen of verwijderen), echter wel binnen de in het portaal vastgelegde toegangsregels (autorisatie).

### **1.2 Uitleg portals**

De term 'portal' leek aanvankelijk weer zo'n hypewoord uit Amerika. Aangezien dit steeds vaker terug komt op het internet is het interessant de tot stand uit te zoeken wat het nu betekent.

We beginnen ons onderzoek bij whatis.com. Een site die uitleg biedt over de meeste typische internet termen. Volgens whatis.com is een portal een website, waarvan het de bedoeling is dat je die als startpunt voor je verdere surftochten. Een portal biedt minimaal links naar andere websites. Het liefst aangevuld met de mogelijkheid om ook te zoeken op andere sites. Portals moeten daarnaast nieuws bieden en andere actuele informatie, zoals weerberichten en beurskoersen. Verder kan er vaak gezocht worden naar telefoonnummers en postcodes. Helemaal perfect is het als je routes kan bekijken op een online wegenkaart. Ten slotte moet de gebruiker met een portal ook kunnen communiceren met andere gebruikers. Via webmail voor persoonlijke berichten. En via online forums en chatboxen voor openbare discussie. De laatste ontwikkeling is dat je een portal aan je eigen smaak kunt aanpassen. Bij een volgend bezoek herkent de server je browser en filtert de informatie waar je toch geen interesse in hebt eruit. Tot zover whatis.com.

Een voorbeeld van een Nederlandse portal is:

<http://mijn.telegraafnet.nl/> - Mijn Telegraafnet

Bij sommige portals is het mogelijk deze na eigen inziens aan te passen, en alles wat je niet zint weg te klikken, zoals

<http://www.startpagina.nl> - Startpagina

Tenslotte zijn er de sites van een aantal Nederlandse zoekmachines en een aantal mediabedrijven, die zich in de loop der tijd meer richting portal ontwikkeld hebben.

<http://www.ilse.nl> - Ilse

<http://info.omroep.nl/portal/> - Gezamenlijke Omroepen

<http://www.tmfweb.nl/> - TMF

<http://www.track.nl> - Track van Wegener

<http://www.vindex.nl> - Webwereld IDG

<http://www.veronica.nl/> - Veronica

## Hoofdstuk 2. Onderzoek portals

### 2.1 Algemeen

Voor het onderzoek naar een geschikte portal is er geprobeerd zo veel mogelijk diversiteit tussen de portals te hebben. Dit houdt in dat er gekeken is naar een verscheidenheid aan programmeertalen, freeware / betaling, type databaseondersteuning en indien mogelijk de systeemeisen.

Het onderzoek is gehouden onder 12 portals. Deze zijn opgezocht via <http://www.google.com>, <http://www.altavista.com>, <http://www.downloads.com> en <http://sourceforge.net>. Bij elke portal is getracht deze te installeren en te testen op verschillende onderdelen. Ook is de mogelijkheid tot uitbreiding als belangrijk aspect meegenomen in de testing, omdat de portal de basis zal worden van het project "KOOP".

### 2.2 Portal Uitleg

In dit hoofdstuk wordt bij elke portel een korte uitleg gegeven wat de mogelijkheden zijn. De eerste hiervan was in het Spaans en kon ik niet vertalen.

Naam portal	Uitleg van de portal
portal solidario	In het Spaans (die is bij mij niet zo goed)
aspapp	PortalApp is een web portal systeem bestaande uit verschillende ASP modules en een gewone database. Hierin is verwerkt: manage users, membership, Content, articles, events, news, Links discussions forums, files, adevertising, FAQ, downloads, surveys shopping cart, classified ads, en meer. Het design kan makkelijk worden aangepast dmv style sheets en aanpasbare html. Access en SQL database ondersteund.
amazing portal generator (APG)	In APG is het mogelijk in korte tijd HTML pagina's te creeren. Hierin kunnen gemakkelijk links worden verwerkt en de layout worden aangepast.
p4drtal	Met 4D Portal, heb je de beschikking om gemakkelijk online toegang te krijgen tot verschillende applicaties, communities en documenten.
fullxml	Een portal met de volgende functionaliteiten: <ul style="list-style-type: none"><li>- Chat</li><li>- Event calendar</li><li>- Forum patch</li><li>- FullXML Editor replaced by HTMLArea</li><li>- <b>Internal</b> messaging</li></ul>



	<ul style="list-style-type: none"> <li>- Knowledgebase</li> <li>- Login not case sensitive</li> <li>- Member manager</li> <li>- Newsletter manager (supports HTML-Mail and attachments)</li> <li>- Photo gallery with category and folder function</li> <li>- Security patch</li> <li>- Single login in intranet</li> <li>- upload to subfolders (with ability to create and drop directories)</li> <li>- Version manager</li> </ul>
Crystal Space 3D SDK	Crystal Spase is een open source 3D omgeving. Welke gebruik maakt van Open GL omgeving
liferay	<p>Liferay is een Java / J2EE gebaseerde portal welke ondersteund:</p> <ul style="list-style-type: none"> <li>- personalization, web email, document library, Message boards, shopping, en andere webtools.</li> </ul>
yacoom	<p>Yacoom is een PHP/SQL gebaseerde Portal, met de focus op:</p> <ul style="list-style-type: none"> <li>- Makkelijke uitbreidbaarheid</li> <li>- Gebruik van XML</li> <li>- Object georiënteerde</li> <li>- PHP en ADODB</li> </ul>
Rainbow	Rainbow is een content management systeem gebaseerd op de Ibuyspy portal. Dit is uitgebreid met multi-language, multiportals en verschillende thema's.
Gluecode Portal Foundation Server™ 3.2 (PFS)	<p>PFS is een java gebaseerde, open source framework voor het maken van portals. PFS is een gratis product, maar wel met een licentie. Hierin is verwerkt:</p> <p>Native user management framework with support for users, groups, roles, and permissions</p> <p>Portlet creation tools for HTML, URL, and RSS portlets</p> <p>Content publishing of unstructured documents</p> <p>Customization of look-and-feel</p> <p>Support for open standards</p>
DotNetNuke XXL	Multi-language en verbeterde versie van het DotNetNuke project, met module toevoeg mogelijkheid
DotNetNuke	DotnetNuke is een open source, content management systeem gebaseerd op het DotNet platvorm. Werkt samen met een SQL database.

### 2.3 Portal specificaties

Elke portal heeft zijn eigen specificaties, zoals het gebruik van een database, programmeertaal en modules. Hieronder staat weergegeven welke specificaties de portals hebben en hoe deze overkomt op de gebruiker. Hieraan zijn bepaalde gewichten aan toegekend.

Naam	Taal (1)	Program-meertaal	Systeem-eisjes	Database	Modules	Uitbreid-baarheid (2)	Online aan-pasbaarheid	Overzichtelijheid	Presentatie	Bedrag Euro
portal solidario	SP	Php	IE5 + cookies	-	-	NM	8	8	8	72,00
aspapp	EN	Asp	-	Access of SQL database	Zie website (16/47)	M	8	7	6	Freeware Voledige versie 389,00
amazing portal generator	EN	Html	Geen	Geen	Geen	M	-	6	2	Pop up
p4drtal	EN	C	IE 3.02 of hoger	Stand-alone (4D)	Zie website	NM	6	6	6	Freeware
fullxml	NL & EN	XSL & ASP	IE 5.0	Xml files	Zie website	M	6	7	7	Freeware
Crystal Space 3D SDK	EN	C++, assembly Perl, Python	IE en Netscape	-	-	M	6	8	8	Freeware
iferay	EN	Java	IE en Netscape	DB2, Firebird, Hypersonic, MySQL, Oracle, PostgreSQL, SAP, SQL Server	-	M	-	7	5	Freeware
yacoom	EN	Php	PHP 4.0.4	SQL	-	M	6	7	3	Freeware
Rainbow	EN	Asp.net	IE of netscape 6	SQL	25	M	7	7	7	Freeware
Gluecode Portal Foundation Server™ 3.2	EN	java	J2SE 1.4	Hyper-sonic SQL	-	M	7	8	8	Freeware
DotNetNuke XXL	Multi	Dot net	IIS 5.0	SQL	25	M	8	8	8	Freeware
DotNetNuke	EN	Dot net	IIS 5.0	SQL	19	M	8	8	8	Freeware

Footnotes (1) en (2) op volgende pagina

Footnote

- Geen melding over  
gemaakt. / niet bekend

**(1)**

EN Engelse  
NL Nederlandse  
SP Spaanse

**(2)**

M Mogelijk  
NM Niet Mogelijk



## Hoofdstuk 3. Conclusie

### 3.1 Inleiding

In dit hoofdstuk is er op een overzichtelijke manier de belangrijkste portals (onderstaande tabel, van boven naar onder) onder elkaar gezet. Aan elke portal is een totale waarde toegekend met een schaal van 0 tot 100. Indien er geen duidelijke vermelding is gemaakt van bepaalde specificaties zoals het gebruik van een database ed, is hier de vast waarde van 3 toegekend.

Gewicht	10 %	15 %	15 %	25 %	10 %	15 %	10 %	
Portal Naam	gebruik SQL Database	belangrijke Modules (1)	Uitbreidbaarheid	aanpasbaarheid	overzichtelijkheid	presentatie	bedrag	totaal index
DotNetNuke	10	10	10	10	8	8	8	88,2
DotNetNuke XXL	10	10	10	10	8	8	8	88,2
Gluecode	10	3	10	10	7	8	8	86,9
Crystal Space 3d	3	3	10	10	6	8	8	85,95
Portal Solidario	3	3	1	10	8	8	8	84,2
Aspapp	10	10	10	10	8	7	6	77,9
Rainbow	10	10	10	10	7	7	7	77,8
fullxml	1	10	10	10	6	7	7	76,65
Liferay	10	3	10	10	3	7	3	75,15
Yacoom	10	3	10	10	6	7	3	75,9
p4drtal	1	10	1	10	6	6	6	65,15
amazing portal generator	1	1	10	10	3	6	2	63,8

(1)

Belangrijke modules zijn Announcements, links en documents.

### 3.2 De beste portal

Als beste is de portal DotNetNuke uit de test gekomen. Hierbij zijn de plaatsen verdeelt tussen DotNetNuke (normal), en DotNetNuke XXL (v.1.0.9). Het verschil is dat DotNetNuke minder fouten bevat dan DotNetNuke XXL. DotNetNuke heeft wel weer een mogelijkheid om modules aan de applicatie toe te voegen. (De volgende versie van DotNetNuke XXL (v.1.0.10) is zit nog in de testfase maar zal 1 deze dagen uitkomen)

# Bijlage 1

Portal Locations			
Naam	Locatie	voorbeeldsite	Comments
portal solidario	<a href="http://www.portalsolidario.net/">http://www.portalsolidario.net/</a>	<a href="http://www.portalsolidario.net/diario/index.php?no=no">http://www.portalsolidario.net/diario/index.php?no=no</a>	alleen hosting
aspapp	<a href="http://www.aspapp.com/apps/products.asp?catid=66&amp;prodid=portalapp">http://www.aspapp.com/apps/products.asp?catid=66&amp;prodid=portalapp</a>		
amazing portal generator	<a href="http://www.rtgsoftware.com/apg/index.htm">http://www.rtgsoftware.com/apg/index.htm</a>		alleen html
p4drtal	<a href="http://www.my4dportal.com">http://www.my4dportal.com</a>		stand-alone db . Niet aanpasbaar
fullxml	<a href="http://www.fullxml.com">www.fullxml.com</a>		db is xml
Crystal Space 3D SDK	<a href="http://sourceforge.net/projects/crystal/">http://sourceforge.net/projects/crystal/</a>	<a href="http://crystal.sourceforge.net/drupal/node.php?id=4">http://crystal.sourceforge.net/drupal/node.php?id=4</a>	niet werkend (slecht helpfile)
liferay	<a href="http://sourceforge.net/project/showfiles.php?group_id=49260">http://sourceforge.net/project/showfiles.php?group_id=49260</a>	<a href="http://www.liferay.com">www.liferay.com</a>	niet werkend (slecht helpfile)
Yacoom	<a href="http://sourceforge.net/project/showfiles.php?group_id=73519">http://sourceforge.net/project/showfiles.php?group_id=73519</a>	<a href="http://yacoom.sourceforge.net/index.php?x">http://yacoom.sourceforge.net/index.php?x</a>	werkt alleen op php 4
Rainbow	<a href="http://sourceforge.net/project/showfiles.php?group_id=66837">http://sourceforge.net/project/showfiles.php?group_id=66837</a>	<a href="http://www.rainbowportal.net/Rainbow/DesktopDefault.aspx">http://www.rainbowportal.net/Rainbow/DesktopDefault.aspx</a>	Ibuyspy based
Gluecode Portal Foundation Server™ 3.2	<a href="http://www.gluecode.com/website/html/download.html">http://www.gluecode.com/website/html/download.html</a>	<a href="http://www.gluecode.com">http://www.gluecode.com</a>	Moet in het bezit zijn van Java J2SE 1.4
DotNetNuke XXL	<a href="http://sourceforge.net">http://sourceforge.net</a>	<a href="http://dotnetnuke.de">http://dotnetnuke.de</a>	Mogelijkheid tot extra module invoegen
DotNetNuke	<a href="http://dotnetnuke.com">http://dotnetnuke.com</a> officiële site:	<a href="http://dotnetnuke.com">http://dotnetnuke.com</a> <a href="http://www.anokamasons.org">http://www.anokamasons.org</a> <a href="http://www.pcrelief.net/">http://www.pcrelief.net/</a>	