

Rapport project

**Ontwerp van een online registratieinterface voor
museumobjecten op basis van ADLIB-technologie**

Fase I (2004-2005)



Centrum Agrarische Geschiedenis vzw

Vlamingenstraat 39
3000 Leuven

tel.: +32 (0)16 32.35.25
fax: +32 (0)16 32.35.26
e-mail: contact@cagnet.be

www.cagnet.be

Inhoudstafel

1	INLEIDING	4
2	PROJECTOMSCHRIJVING EN METHODE	5
2.1	Uitgangspunt.....	5
2.1.1	Collectieregistratie bij niet-professionele musea	5
2.1.2	Schrijfmogelijkheden ADLIB Internet Server	5
2.2	Doelstellingen van dit project	6
2.3	Motivering van het project	6
2.4	Werkwijze	7
2.4.1	Methodiek van het project	7
2.4.2	Verloop werkzaamheden.....	8
2.4.3	Uitvoerder en Partners.....	8
3	UITWERKING VAN HET PROJECT.....	10
3.1	Vaststellen van de vereisten	10
3.1.1	Werkwijze	10
3.1.2	Vereisten.....	10
3.2	Logisch design	14
3.2.1	Logisch design database	14
3.2.2	Logisch design toepassing	15
3.2.3	Logisch design interface	16
3.3	Systeemarchitectuur	19
3.3.1	Cliënt-side systeemarchitectuur	19
3.3.2	Server-side systeemarchitectuur	21
3.3.3	Verantwoording van de gemaakte keuzes	26
3.4	Fysisch design.....	28
3.4.1	Fysisch design database	28
3.4.2	Fysisch design toepassing	33
3.4.3	Fysisch design interface	44
3.5	Testfase	59
3.5.1	Heemmuseum Het Molenijzer Putte.....	59
3.5.2	Heemmuseum De Zuiderkempem Westerlo	60
3.6	Publicatie en verspreiding van de resultaten	61
3.6.1	Publicatie geregistreerde collectieinformatie	61
3.6.2	Verspreiding resultaten onderzoek.....	62
4	CONCLUSIE EN MOGELIJKE VERDERE STAPPEN	63
4.1	Conclusie.....	63

4.2	Mogelijke verdere stappen	63
5	BIJLAGEN	65
5.1	Bijlage 1: Problemen ADLIB Internet Server	65
5.2	Bijlage 2: Deelnemers studiedag rond Agrarisch Erfgoed 14/09/2005	66

1 Inleiding

In dit rapport wordt verslag uitgebracht over de eerste fase van het project 'Ontwerp van een online registratieinterface voor museumobjecten op basis van ADLIB-technologie'. In de loop van dit project werd een applicatie ontwikkeld die toelaat om museumobjecten via het Internet te registreren in een ADLIB databank. Deze toepassing werd uitgetest door twee heemmusea, Het Molenijzer uit Putte en Het Museum van de Zuiderkempen uit Oevel (Westerlo).

Dit verslag valt uiteen in drie delen. Het eerste deel gaat in op de uitgangspunten, doelstellingen, motivatie en methodiek van dit project. Het tweede deel bevat de antwoorden op de onderzoeksvragen die in dit project centraal stonden. Hierbij worden achtereenvolgens de vereisten, het logisch design, de systeemarchitectuur, het fysisch design, de test van de ontwikkelde toepassing en de verspreiding van de resultaten besproken. Een derde deel tenslotte bevat enkele conclusies en suggesties voor verdere ontwikkeling.

De doelstelling van dit verslag is enerzijds rapporteren over het verloop van het project. Anderzijds wil het ook nuttige informatie bieden aan iedereen die er aan denkt om een gelijkaardig project te starten met de ADLIB Internet Server.

Uiteraard is de belangrijkste output van dit project de applicatie zelf. Op vraag kan een proefversie van deze toepassing worden uitgetest. Wij bezorgen u dan URL van de webpagina waar de applicatie zich bevindt en een gebruikersnaam met bijbehorend paswoord. Als u hierin geïnteresseerd bent, kan u contact opnemen met Jo Bekaert (CAG - Vlamingenstraat 39 - 3000 Leuven - 016/32.35.42 - jo.bekaert@cagnet.be).

Tot slot wil ik iedereen bedanken die dit project heeft mogelijk gemaakt. In de eerste plaats de Vlaamse Gemeenschap die dit project financieel heeft ondersteund. In de tweede plaats alle mensen die vrijwillig hun steun verleenden aan dit project. Zowel de mensen die deelnamen aan de vergaderingen van de werkgroep (Het Heemmuseum van Putte, Het Druivenmuseum in Overijse en de Midzeelhoeve) als de leden van de stuurgroep (Peter Heyrman (KADOC), Jan Cools (Culturele Biografie Vlaanderen), Ode De zutter (Project Move Provincie Oost-Vlaanderen), Marc Jacobs (VCV), Robert Nouwen (Openluchtmuseum Bokrijk) en Hans Nissens (Cultuurnet Vlaanderen)). In het bijzonder wil ik Staf de Winter (Heemmuseum Het Molenijzer), Remi Heylen en Guy Thys (Heemmuseum De Zuiderkempen) bedanken voor het uittesten van de toepassing.

Jo Bekaert



Met de stein van de Vlaamse overheid

2 Projectomschrijving en methode

2.1 Uitgangspunt

Aan de basis van dit project liggen twee uitgangspunten: de graad van collectieregistratie bij niet-professionele musea en de nieuwe mogelijkheden van de ADLIB Internet Server.

2.1.1 Collectieregistratie bij niet-professionele musea

Een van de problemen waar vele kleine, dikwijls niet – professionele, musea mee kampen, is het ontbreken van een degelijke collectieregistratie, volgens de standaarden die in dit domein gelden. Dat werd nogmaals vastgesteld door Bert Woestenborghs die binnen het kader van het project Veldwerk-Denkwerk van het Centrum Agrarische Geschiedenis een heel aantal van deze musea, met een agrarische collectie, bezocht. De onderstaande grafiek toont aan dat van de bezochte collecties bijna twee derde niet of slechts gedeeltelijk geïnventariseerd is. Vele van deze instellingen beschikken niet over de nodige kennis en middelen om een dergelijke registratie te realiseren. Dikwijls maken in deze musea vrijwilligers de dienst uit. Ondanks het feit dat vele van deze mensen goed werk verrichten, ontbreekt het hen in veel gevallen aan de nodige kennis om een professionele registratie te realiseren. Dit heeft tot gevolg dat de registratie ofwel niet gebeurt ofwel niet afgestemd is op de standaarden die in de museumsector gelden. Hierdoor wordt de uitwisseling van collectiegegevens en het mogelijke gebruik ervan, door een brede groep mensen, beperkt. Daarnaast beschikken deze instellingen niet over de nodige financiële middelen om de technische hulpmiddelen aan te kopen, die in de professionele museumsector gebruikt worden. In het bijzonder geldt dit voor de registratiesoftware, die rekening houdend met het beschikbare budget van deze instellingen zeer duur is.

Inventarisatie collecties Veldwerk

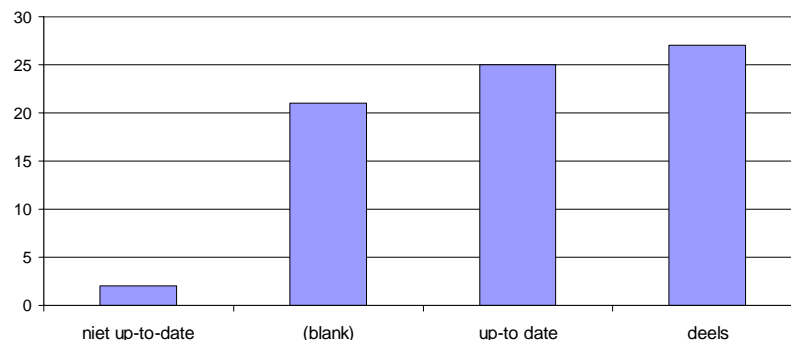


Diagram 1: Grafiek inventarisatiegraad landbouwmusea (Bron: Project Veldwerk-Denkwerk CAG)¹

2.1.2 Schrijfmogelijkheden ADLIB Internet Server

ADLIB Museum is een softwarepakket voor collectieregistratie dat een aantal jaren geleden door de Vlaamse overheid ter beschikking werd gesteld aan een grote groep

¹ Woestenborghs, B., *Veldwerk Denkwerk. Agrarisch erfgoed: stand van zaken*. Centrum Agrarische Geschiedenis. 2005.

musea in Vlaanderen. Dit programma wordt ontwikkeld door de Nederlandse firma ADLIB Information Systems en biedt een groot aantal mogelijkheden voor de registratie van museumobjecten. Daarnaast biedt dit systeem de mogelijkheid om een geregistreerde collectie op het internet te publiceren via de ADLIB Internet Server. Dat is een afzonderlijk programma dat door ADLIB speciaal hiervoor werd ontwikkeld. Sinds versie 5.0 van de software ter beschikking kwam, is het mogelijk om deze module te gebruiken om via het internet data weg te schrijven in een ADLIB databank. ADLIB biedt tot op heden echter nog geen interface aan, ontwikkeld met standaard webtechnologie, om via deze module aan collectieregistratie te doen. Nochtans biedt deze techniek (ADLIB Internet Server + standaard webtechnologie) interessante mogelijkheden voor ontwikkelaars en gebruikers. Deze werden tot hier toe echter onvoldoende uitgetest. In het onderdeel systeemarchitectuur van dit rapport worden ze op een rijtje gezet.

2.2 Doelstellingen van dit project

Om in te spelen op deze problematiek startte het Centrum Agrarische Geschiedenis een project om een webapplicatie te ontwerpen, die niet-professionele musea moet toelaten om gegevens over hun eigen museumobjecten, op eenvoudige wijze, via het internet in één centrale databank op te slaan. De bedoeling was musea de mogelijkheid te bieden om hun collectie kosteloos, en via de geldende standaarden te registreren. Als technisch platform werd gekozen voor de technologie van ADLIB Information Systems, om op die manier de schrijfmogelijkheden van de ADLIB Internet Server te kunnen uittesten.

Het verslag dat hier voor ligt, heeft betrekking op de eerste fase van dit project, die liep van juni 2004 tot juni 2005. Tijdens deze eerste fase werd gezocht naar een antwoord op de volgende vragen:

- Aan welke eisen moet de applicatie voldoen?
- Is het mogelijk om op basis van de databanktechnologie van ADLIB Information Systems een proefapplicatie te bouwen?
- Is een dergelijke applicatie ook werkbaar in de praktijk voor de beoogde doelgroep van niet-professionele musea?
- Bestaat er interesse voor een dergelijke toepassing?

2.3 Motivering van het project

De resultaten van dit onderzoek zijn relevant voor twee verschillende doelgroepen. In de eerste plaats voor de niet-professionele musea met beperkte middelen en ten tweede voor alle gebruikers van de software van ADLIB Information Systems.

Een van de doelstellingen van het Centrum Agrarische Geschiedenis is het bieden van ondersteuning aan erfgoedinitiatieven, die werken in het themaveld landbouw-voeding. De ondersteuning strekt zich uit over verschillende domeinen, waaronder ook collectieregistratie. De uitkomsten van dit project zijn zeer relevant voor deze ondersteunende functie. Een groot deel van de **niet-professionele musea** beschikt immers over een **deelcollectie landbouw-voeding**, die in vele gevallen niet of zeer beperkt geregistreerd is. Dit project levert mogelijk een instrument op dat voor deze musea op termijn bruikbaar is om een deel van hun collectie op gestandaardiseerde manier te registreren, zonder dat zij daar zelf een grote investering voor hoeven te doen.

Daarnaast bevordert dit project bij musea die mee in dit project stappen de kennis op vlak van standaarden en registratieprocedures.

Ten slotte is het maar een kleine moeite om collectiegegevens, eens deze geregistreerd zijn, aan een breed publiek ter beschikking te stellen via een website. Bijvoorbeeld via www.hetvirtueleland.be die door het CAG wordt beheerd. Op deze site staan het agrarisch erfgoed en de instellingen die dit erfgoed bewaren centraal. In theorie kan elke collectie agrarisch erfgoed op deze site een plaats krijgen.

Daarnaast is het CAG lid van de ADLIB-trekkersgroep in Vlaanderen, die bijeenkomsten van de Vlaamse **ADLIB gebruikersgroep** organiseert en gebruikers op vraag ondersteunt in hun gebruik van het programma. Onderzoek rond toepassing van de ADLIB technologie, en de verspreiding van de resultaten kan musea er toe aanzetten om hun software actiever te gaan gebruiken en de mogelijkheden ervan optimaal te benutten. Het CAG gebruikt de software van ADLIB als basis voor de uitbouw van haar website rond de geschiedenis van landbouw en voeding (www.hetvirtueleland.be). In de voorbije jaren bouwde het CAG heel wat expertise op in het werken met deze technologie, wat goed van pas kwam bij de uitwerking van het project. Dankzij dit project kon deze kennis nog verder worden uitgebouwd met als doel deze ter beschikking te stellen van andere ADLIB-gebruikers.

De mogelijkheid om gegevens via webtechnologie in de databank weg te schrijven biedt interessante perspectieven om op een relatief eenvoudige manier (want gebruik makend van veelgebruikte, meer standaard technologieën als HTML, Javascript etc.) eigen applicaties rond de ADLIB databank te bouwen. Een van de directe voordelen hiervan is dat licentiekosten kunnen worden gereduceerd indien men een eigen applicatie ontwikkelt. Men koopt immers één licentie aan van de databank en één licentie van de ADLIB Internet Server met een onbeperkt aantal gebruikers. Op die manier kan men met een onbeperkt aantal gebruikers via de Internet Server op deze databank werken.

2.4 Werkwijze

2.4.1 Methodiek van het project

Als overkoepelende methode voor de uitvoering van dit project werd gekozen voor een benadering waarbij er, op basis van een aantal verzamelde vereisten en een analyse hiervan, relatief snel een prototype werd ontwikkeld dat dan met de betrokken partijen kon worden getest en besproken. De opmerkingen werden vervolgens verwerkt in een nieuw prototype dat dan opnieuw geëvalueerd en verbeterd kon worden. Het komt er op neer dat een aantal fasen van de ontwikkelingscyclus van het programma verschillende keren doorlopen werden en niet één keer zoals dat met een meer lineaire benadering het geval is.

Deze methode heeft het voordeel dat men sneller kan werken en snel kan inspelen op eventuele problemen die zich voordoen. Ook verhoogt deze manier van werken de betrokkenheid van de eindgebruiker van de applicatie bij de ontwikkeling, omdat er regelmatig momenten van overleg plaatsvinden, waarbij er opmerkingen en suggesties tot verbetering kunnen worden gegeven. De interactie tussen “gebruiker” en “ontwerper” is met andere woorden zeer belangrijk. De eindgebruiker (in dit geval het niet-professionele museum) heeft zo meer het gevoel dat hij meewerkt aan het project in plaats van dat er iets boven zijn hoofd wordt uitgewerkt waar hij of zij geen invloed op heeft.

2.4.2 Verloop werkzaamheden

juni - augustus 2004

- Verkenning van de schrijfmogelijkheden van de ADLIB Internet Server
- Ontwikkeling proeftoepassing om de schrijfmogelijkheden uit te testen

september - oktober 2004

- In kaart brengen vereisten van de applicatie
- Bijeenkomst stuurgroep
- Bijeenkomst werkgroep

oktober - januari 2004

- Ontwikkeling werkende applicatie

januari - juni 2005

- Test invoer met het Molenijzer in Putte
- Verwerking opmerkingen - aanpassing applicatie

februari - juni 2005

- Test invoer met het Museum van de Zuiderkempen
- Verwerking opmerkingen – aanpassing applicatie

juli 2005

- Verbetering records
- Verwerking opmerkingen en aanpassing applicatie

augustus-september 2005

- Opmaken rapport
- Aanpassing applicatie
- 14/09 studiedag

2.4.3 Uitvoerder en Partners

Het project werd uitgevoerd door Jo Bekaert, medewerker van het Centrum Agrarische Geschiedenis in samenwerking met het museum Het Molenijzer uit Putte en het Museum van de Zuiderkempen uit Westerlo.

Tijdens dit project werd ook een vergadering belegd met een aantal experts om advies in te winnen over het project. In deze adviesgroep zaten de volgende mensen: Jan Cools (Culturele Biografie Vlaanderen), Marc Jacobs (VCV), Peter Heyrman (KADOC),

Ode De Zutter (Project Move – Provincie Oost-Vlaanderen), Robert Nouwen (Bokrijk) en Hans Nissens (Cultuurnet Vlaanderen).

3 Uitwerking van het project

3.1 Vaststellen van de vereisten

3.1.1 Werkwijze

Om de eisen waaraan de applicatie moest voldoen in kaart te brengen werd op 22 september 2004 een bijeenkomst belegd met een aantal experts. Tijdens deze bijeenkomst werd het project voorgesteld en besproken. Aanwezig tijdens deze vergadering waren Peter Heyrman (KADOC), Jan Cools (Culturele Biografie Vlaanderen), Ode De zutter (Project Move Oost-Vlaanderen), Marc Jacobs (VCV) en Robert Nouwen (Bokrijk). Hans Nissens (Cultuurnet Vlaanderen) kon jammer genoeg niet aanwezig zijn. De voornaamste opmerkingen die uit deze vergadering naar voor kwamen hadden betrekking op de te registreren informatie en op copyright.

Daarna werden een klein aantal musea uitgenodigd die in aanmerking konden komen als gebruiker van de toepassing. De bijeenkomst werd gehouden op 14 oktober 2004. Op deze bijeenkomst, waren drie musea vertegenwoordigd: Het Molenijzer (Putte), Het Druivenmuseum (Overijse) en De Midzeelhoeve (St.-Katelijne-Waver). Tijdens deze bijeenkomst werd het project voorgesteld en met de aanwezigen besproken.

De voornaamste conclusie die uit deze bijeenkomst getrokken werd, was dat de aanwezige musea erg geïnteresseerd waren in het project en bereid tot medewerking, maar zelf nog geen uitgesproken verwachtingen hadden van de applicatie. Eerst was gepland om onder de musea met een collectie agrarisch erfgoed in Vlaanderen een enquête te houden, met de bedoeling om hun eisen en verwachtingen van een dergelijke applicatie in kaart te brengen. Op basis van de uitkomst van de bijeenkomst werd er voor geopteerd om af te zien van een schriftelijke enquête. Dit zou immers weinig concrete resultaten opgeleverd hebben.

Na de bijeenkomst werd besloten om een andere manier van werken te volgen. Er werd voor geopteerd om eerst op basis van literatuur zelf een lijst van vereisten op te stellen, om dan op basis van deze bevindingen binnen relatief korte termijn een eerste demoversie van de applicatie te kunnen ontwerpen. Deze demo zou dan door enkele betrokkenen worden geëvalueerd. De opmerkingen die hieruit voort kwamen, konden dan verwerkt worden in de proefapplicatie die dan opnieuw kon worden geëvalueerd enz.

Eind oktober 2004 was er een eerste lijst met vereisten beschikbaar en werd met de ontwikkeling van de eigenlijke applicatie gestart. Tijdens de duur van het project werd de lijst met vereisten voortdurend uitgebreid met opmerkingen en suggesties die door de gebruikers van de toepassingen naar voor werden geschoven.

3.1.2 Vereisten

In wat volgt zullen de vereisten waaraan de applicatie moest voldoen worden opgesomd. De vereisten worden gegroepeerd onder de rubrieken systeemarchitectuur, te registreren gegevens, functionaliteit, standaarden en interface design.

3.1.2.1 Systemarchitectuur

- Er dient gebruik gemaakt te worden van een cliënt-server architectuur.
- De applicatie moet bruikbaar zijn, louter door middel van een standaard webbrowser (Internet Explorer, Firefox) die werkt op een doorsnee computer met een internetverbinding.
- Als DBMS (Database Management System) moet er gebruik gemaakt worden van de software van ADLIB Information Systems.
- Als Serverplatform dient gebruik gemaakt te worden van Windows 2000 Server. Om te werken met de ADLIB Internet Server is dit noodzakelijk.

3.1.2.2 Te registreren gegevens

De applicatie heeft de bedoeling om collectieregistratie mogelijk te maken. Over de manier waarop collectieregistratie het best wordt aangepakt, bestaat redelijk wat literatuur². Op basis hiervan werd bepaald welke informatie er zeker met de proefapplicatie moest kunnen worden geregistreerd.

Dit resulteerde in de selectie van de volgende velden die zeker aanwezig moesten zijn:

- Instellingsnaam
- Objectnummer / Inventarisnummer
- Objectcategorie
- Objectnaam
- Titel
- Beschrijving
- Vervaardig(st)er
- Datering (begin en einde vervaardiging en de precisie)
- Materiaal
- Toestand
- Afmetingen
- Aard Verwerving
- Persoon of instelling waarvan verworven werd
- Bedrag verwerving
- Datum verwerving
- Standplaats
- Trefwoord
- Opmerkingen
- Afbeelding

Daarnaast werden nog een aantal extra velden voorzien:

- Opschrift/merk
- Techniek vervaardiging
- Plaats vervaardiging

² In het Nederlands verschenen onder meer de volgende publicaties: Hogenboom, Jeanne. *Basisregistratie voor collecties, voorwerpen en beeldmateriaal*. Rotterdam, 1988, 114 p. en *Basiscursus: Registratie en documentatie*. Stichting LCM. Tilburg, 2002.

- Copyright
- Toestemming publicatie afbeelding
- Toestemming publicatie objectbeschrijving

3.1.2.3 Functionaliteit

- De applicatie mag enkel toegankelijk zijn na het ingeven van een gebruikersnaam en een paswoord
- Er moet voorzien worden in een handleiding en hulp voor het algemeen gebruik van de applicatie
- Voor elk veld moet er voorzien worden in een helpfunctie die duidelijk aangeeft welke informatie er verwacht wordt in een bepaald veld, aangevuld met informatie over de schrijfwijze en voorbeelden
- Er moeten controles kunnen worden uitgevoerd op de gegevens die worden ingevoerd (uniciteit van bepaalde gegevens, controleren of een veld al dan niet is ingevuld, vorm van de gegevens controleren etc.)
- Het moet mogelijk zijn om velden te herhalen
- Er moet kunnen gewerkt worden met lijsten met daarin gecontroleerde termen
- Het moet mogelijk zijn om de ingevoerde collectiegegevens opnieuw op te zoeken op een eenvoudige en op een meer complexe manier (met Booleaanse operatoren)
- De zoekresultaten moeten op een efficiënte manier kunnen worden doorbladerd en gerangschikt
- Collectiebeschrijvingen moeten kunnen aangepast en eventueel verwijderd worden.
- De gegevens moeten op verschillende manieren kunnen worden uitgevoerd, op het scherm en op papier
- Er moet kunnen aangegeven worden waar de rechten liggen voor wat de afbeelding en de objectbeschrijving betreft
- Afbeeldingen tot 4 MB moeten door de gebruiker zelf kunnen worden doorgestuurd
- Velden moeten gegroepeerd kunnen worden weergegeven
- Het moet mogelijk zijn om zoekresultaten tijdelijk te bewaren

3.1.2.4 Standaarden

De volgende standaarden worden gebruikt:

- De afbeeldingen worden bewaard in TIFF of JPG³
- Benamingen die door de participerende instellingen worden gebruikt om hun afbeeldingen te benoemen, worden zoveel mogelijk gerelateerd aan de Nederlandse versie van de AAT (Arts & Architecture Thesaurus)⁴
- Voor de webpagina's wordt zoveel mogelijk gewerkt naar de standaard HTML 4.01 specificatie

³ Voor meer informatie over de standaarden kan men o.m. terecht in Drake, Karl-Magnus e.a. *Good Practice Handbook version 1.2*. Minerva Working Group 6. 2003, 105 p. en op <http://www.den.nl/>.

⁴ Voor meer informatie over de Nederlandstalige AAT zie <http://www.aat-ned.nl/>

3.1.2.5 Interface design

Het ontwikkelen van een duidelijke en bruikbare interface is niet eenvoudig. Er werd naar gestreefd om een duidelijke, heldere, sobere interface te ontwikkelen zonder al te veel toeters en bellen. Er werd gebruik gemaakt van algemene richtlijnen in dit verband die op talrijke websites terug te vinden zijn⁵.

⁵ Zie onder meer <http://www.webstyleguide.com/interface/> , <http://www.useit.com/> , <http://soc.kuleuven.be/com/mediac/otc/lv/ud/usabilitylinks.htm> ...

3.2.2 Logisch design toepassing

De volgende diagrammen geven een beter inzicht in de toepassingslogica van de applicatie. De bedoeling van deze diagrammen is een beter begrip te krijgen van de functionaliteit van de applicatie op hoog niveau. De diagrammen treden met andere woorden niet in detail. De volgende diagrammen zijn gebaseerd op de eisen inzake functionaliteit.

Diagram 1 geeft de drie belangrijke elementen weer in de applicatie: de gebruiker die aan collectiebeheer wilt doen, het systeem dat in de nodige functionaliteiten voorziet en de databank waarin alle informatie wordt bewaard.

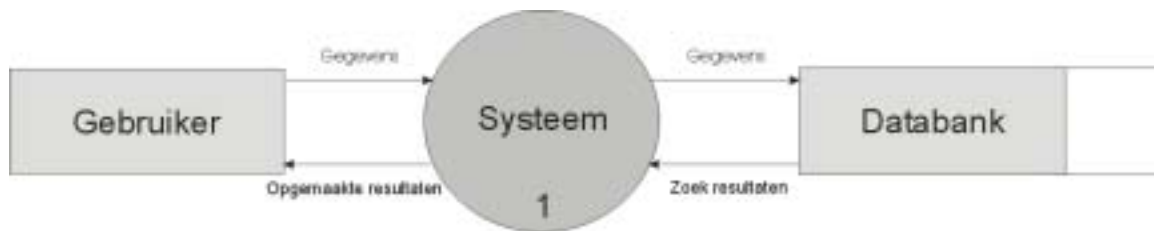


Diagram 3: DFD Niveau 1

De volgende diagrammen geven de bewegingen van data binnen de applicatie weer op een lager niveau. Diagram 2 illustreert dat er binnen het systeem 3 groepen van processen zijn. Ten eerste zijn er processen die verband houden met het opzoeken van informatie. Ten tweede zijn er processen die verband houden met het beheren van de informatie. Ten derde zijn er processen die de identiteit van de gebruiker controleren.

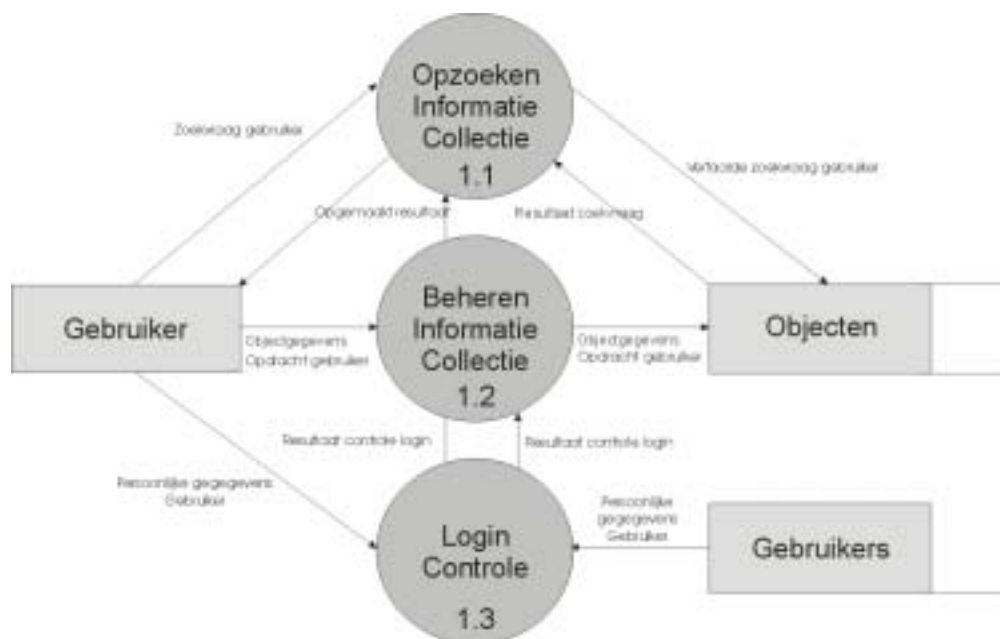


Diagram 4: DFD Niveau 2

Diagram 3 splitst het proces beheren van informatie uit in drie subprocessen: Toevoegen, aanpassen en verwijderen van collectieinformatie. Op dit niveau wordt ook het lock-controle proces vermeld dat verhindert dat records gelijktijdig in gebruik kunnen zijn.

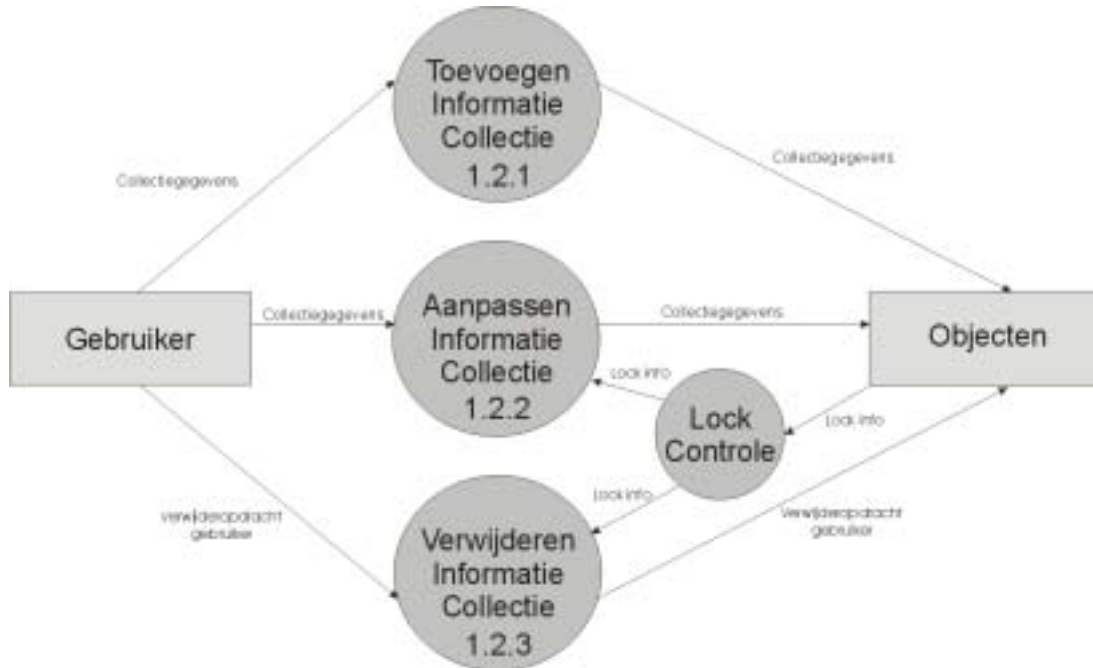


Diagram 5: DFD Niveau 3 – Substelsiem Informatiebeheer

3.2.3 Logisch design interface

Het volgende diagram geeft inzicht in de opbouw van de interface. Het geeft een overzicht van de opeenvolgende stappen die ondernomen moeten worden om de taken, die bij de bespreking van de toepassingslogica werden aangehaald, uit te voeren. Het diagram wordt verduidelijkt aan de hand van een aantal woordelijke beschrijvingen van deze stappen. Zowel het diagram als de beschrijvingen treden niet in detail. Het is de bedoeling om een globaal overzicht van de interface te geven. Vervolgens zullen de belangrijkste mogelijkheden van de applicatie besproken worden in samenhang met de interface. Hierbij wordt niet in detail getreden.

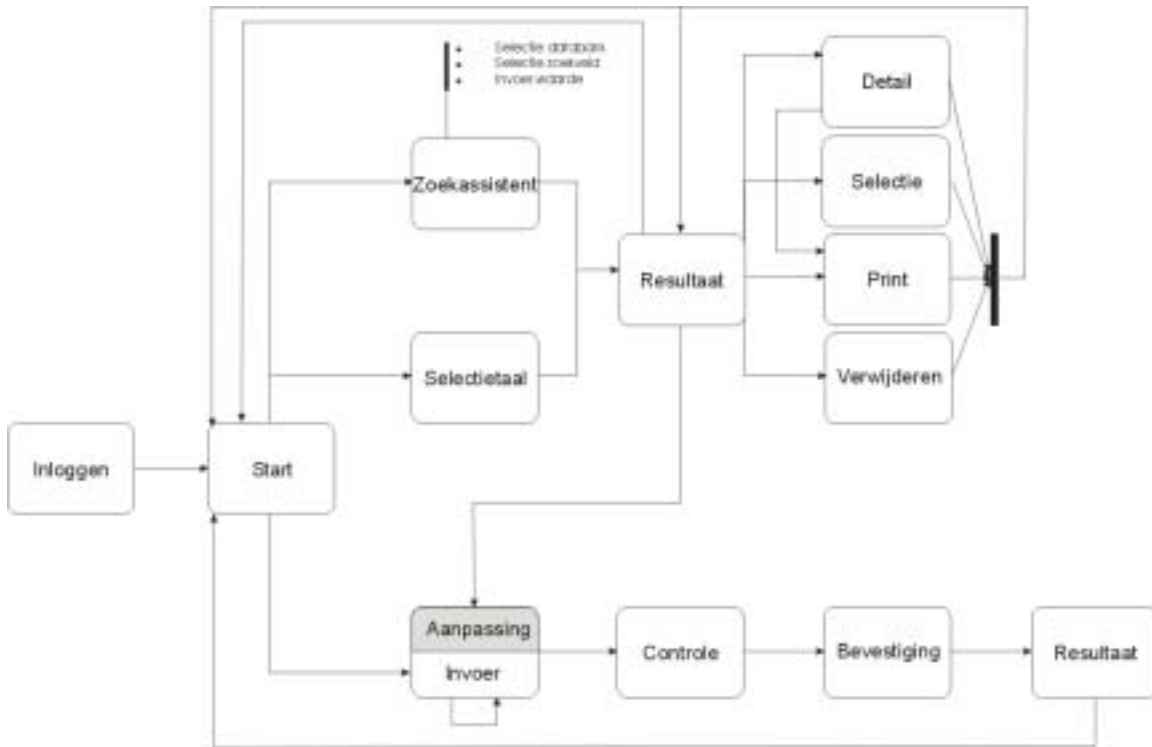


Diagram 6: Flow Interface

3.2.3.1 Inlogprocedure

Vooraleer gebruik te kunnen maken van de applicatie dient een gebruiker eerst in te loggen met een gebruikersnaam en een paswoord. Dit eerst en vooral om er voor te zorgen dat niet iedereen de gegevens in de databank kan benaderen. Daarnaast is deze procedure ook noodzakelijk om te kunnen bepalen welke gegevens voor een bepaalde gebruiker toegankelijk moeten zijn.

3.2.3.2 Zoeken van informatie

De gebruiker wenst informatie op te zoeken in de databank met collectiegegevens. Hiervoor kan hij gebruik maken van twee verschillende zoekmodi. De eerste modus is de zoekassistent. Deze biedt de gebruiker hulp bij het zoekproces, door een stap-voor-stap methode aan te bieden. Een tweede modus maakt gebruik van een selectietaal om complexere zoekopdrachten mogelijk te maken. Beide modi leveren een zoekresultaat op. De gebruiker kan door dit zoekresultaat navigeren om het gewenste resultaat op te zoeken. Vervolgens kan de gebruiker kiezen uit een aantal opties. Hij kan detailinformatie opzoeken over een bepaald object, hij kan een aantal objectbeschrijvingen tijdelijk selecteren, hij kan de gegevens van een object afprinten, hij kan er voor kiezen om een objectbeschrijving te verwijderen of hij kan er voor kiezen een object te bewerken of aan te passen. Terugkeren naar de startpagina is uiteraard ook een optie. Dat is overigens ook een optie na het uitvoeren van eender welke bewerking. Dat geldt ook voor de mogelijkheid om terug te keren naar de resultatenpagina.

3.2.3.3 Toevoegen van informatie

Als een gebruiker een nieuwe objectbeschrijving aan de databank wenst toe te voegen, kan hij hier op het startscherm onmiddellijk voor kiezen. Er wordt een leeg invoerscherm gepresenteerd dat toelaat om alle velden die relevant zijn voor de beschrijving van het object in te vullen. Sommige velden zijn vrij in te vullen, aan andere velden is een trefwoordenlijst gekoppeld waaruit de gebruiker een term kan selecteren. Als de beschrijving volledig is, vraagt de gebruiker aan het systeem om deze inhoud te controleren. Het systeem geeft vervolgens aan of er al dan niet aanpassingen dienen te gebeuren. Indien dit het geval is krijgt de gebruiker de gelegenheid om de nodige aanpassing door te voeren. Vervolgens worden de gegevens opnieuw gecontroleerd. Indien alle controles positief zijn, krijgt de gebruiker de kans om de beschrijving definitief te bevestigen, waarna de gegevens worden opgeslagen. Nadat de gegevens opgeslagen zijn, krijgt de gebruiker te zien of de toevoegactie geslaagd is en de mogelijkheid om een nieuwe optie te selecteren.

3.2.3.4 Aanpassen van informatie

Het aanpassen van gegevens kan de gebruiker doen door de objectbeschrijving die hij wenst aan te passen eerst op te zoeken. Er wordt hem de mogelijkheid geboden om de aanpasfunctie te activeren. Vervolgens krijgt de gebruiker de beschrijving in een aanpasmodus aangeboden. Deze modus komt overeen met de modus waarin een gebruiker informatie kan toevoegen, met als verschil dat de objectgegevens al in de velden ingevuld zijn. Deze gegevens kunnen vervolgens aangepast worden. Als de gebruiker de gegevens volledig acht kan de controlefunctie worden geactiveerd, waarna dezelfde stappen volgen als bij het toevoegen van gegevens (controle, bevestiging, resultaat).

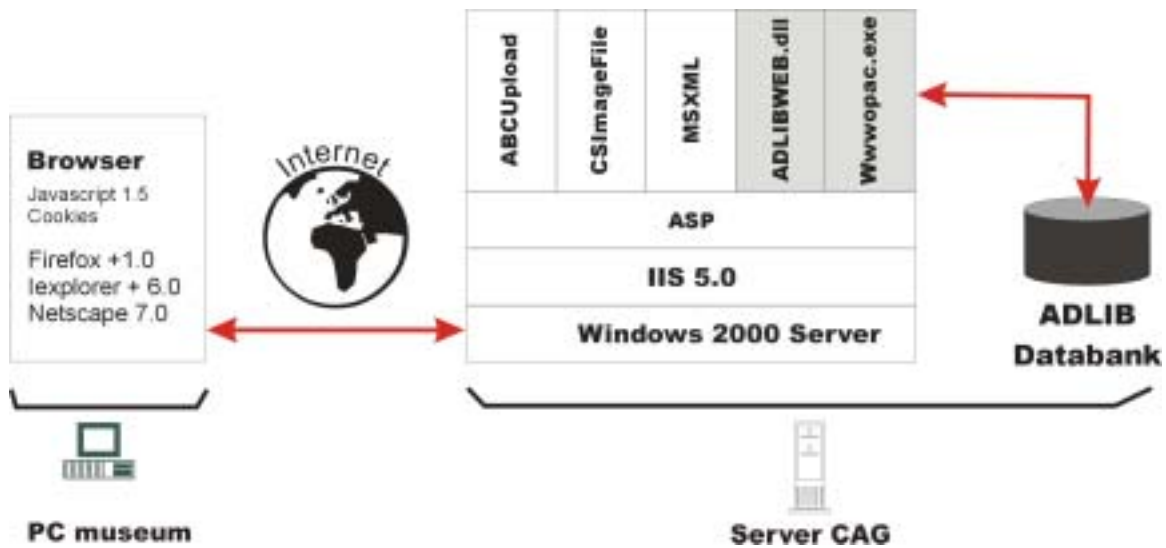
3.2.3.5 Verwijderen van informatie

Een gebruiker kan gegevens uit de databank verwijderen door deze eerst op te zoeken. Vervolgens kan hij de verwijderfunctie activeren en krijgt hij het resultaat van de bewerking aangeboden. Vervolgens krijgt hij de mogelijkheid om terug te keren naar de startpagina.

3.3 Systemarchitectuur

De vereisten die eerder in dit verslag werden beschreven, lieten geen fundamentele keuzemogelijkheden meer toe op vlak van systeemarchitectuur. Op voorhand stond immers vast dat er gewerkt zou worden met een cliënt-server architectuur met webtechnologie en dat de applicatie rond een ADLIB databank zou worden opgebouwd. Gezien ADLIB zelf sterk op Microsoft producten georiënteerd is, lag de keuze voor deze producten op voorhand ook min of meer vast.

De volgende afbeelding geeft een overzicht van de gebruikte architectuur:



In wat volgt zal de gebruikte technologie meer uitgebreid besproken worden. De beschrijving is opgesplitst in een onderdeel cliënt-architectuur en een onderdeel server-architectuur. Gezien dit project de bedoeling had om het gebruik van ADLIB-technologie voor het ontwikkelen van webapplicaties uit te testen, zal hier extra aandacht aan besteed worden. Meer bepaald zal de werking van de ADLIB Internet Server uitvoerig worden besproken.

In de slotparagraaf zal het gebruik van de gekozen architectuur verantwoord worden door voor- en nadelen op een rijtje te zetten.

3.3.1 Cliënt-side systeemarchitectuur

Om de applicatie zo ruim mogelijk toegankelijk te maken, werd er voor geopteerd om de applicatie zo te ontwikkelen dat de cliënt enkel over een computer met internetverbinding dient te beschikken waarop een (gratis) webbrowser draait.

3.3.1.1 Hardware en besturingssysteem

De gebruiker hoeft niet over een zware, zeer recente computer te beschikken om de toepassing te gebruiken. In principe moeten er enkel webpagina's kunnen worden getoond die relatief eenvoudig zijn van structuur, zonder al te veel franjes. Er moeten

dus geen zware processen worden gestart en uitgevoerd langs de kant van de cliënt. Er wordt geen gebruik gemaakt van ActiveX of Java Applets.

De applicatie werd ontwikkeld om vlot te kunnen draaien op een systeem waarop een webbrowser vlot draait. De minimumeisen komen dan ook hier mee overeen. Als indicator worden de systeemeisen die verbonden zijn met het gebruik van Mozilla Firefox weergegeven voor een winkel-omgeving:

Besturingssystemen

- Windows 98
- Windows 98SE
- Windows ME
- Windows NT 4.0
- Windows 2000
- Windows XP (*Aanbevolen*)
- Windows Server 2003

Minimum Hardware

- Pentium 233 MHz (*Aanbevolen: Pentium 500MHz of hoger*)
- 64 MB RAM (*Aanbevolen: 128 MB RAM of hoger*)
- Harde schijf: Geen specifieke eisen, gezien er geen extra zaken moeten geïnstalleerd worden. Als men afbeeldingen wil gaan toevoegen is een harde schijf van enige omvang uiteraard interessant, gezien het uploaden van een vaste schrijf sneller verloopt dan uploaden van een CD- of DVD-lezer.

3.3.1.2 Browser

De applicatie zou op een doorsnee recente browser moeten werken, zoals Internet Explorer 6.0, Firefox 1.0 en Netscape 7.0 . Voorwaarde is dat de browsers HTML 4.01 en Javascript 1.5 ondersteunen en het plaatsen van cookies kunnen toelaten. Dit is het geval met de genoemde browsers.

3.3.1.3 Internetverbinding

Voor wat de internetverbinding betreft, geldt uiteraard hoe sneller hoe beter. Een breedbandverbinding is aanbevolen, zeker als men afbeeldingen over het internet aan de objectbeschrijving wenst toe te voegen. Een belangrijk aandachtspunt in dit verband is de snelheid waarmee gegevens kunnen geupload worden. Voor gewone tekst hoeven er geen speciale eisen te worden gesteld, gezien de omvang hiervan vrij beperkt is. De omvang van afbeeldingen kan daarentegen al snel oplopen. Dit heeft tot gevolg dat een grotere uploadcapaciteit noodzakelijk is, als men vlot grote afbeeldingen wil doorsturen.

De uploadcapaciteit waarover een doorsnee internet gebruiker beschikt is doorgaans aan de lage kant als men dit met de downloadcapaciteit vergelijkt. De onderstaande

tabel illustreert dit aan de hand van enkele producten voor internettoegang van de twee bekendste aanbieders van Internetdiensten in Vlaanderen.

Aanbieder	Product	Max. download	Max. upload
Telenet	ComfortNet	0,5 Mbps	128 Kbps
Telenet	ExpressNet	5 Mbps	192 Kbps
Belgacom	ADSL Plus	4 Mbps	256 Kbps
Belgacom	ADSL Light	512 Kbps	128 Kbps

(situatie op 08/08/2005)

Uiteraard kan niet van elke gebruiker van de applicatie verwacht worden dat hij extra investeringen in zijn internetverbinding zou doen om het uploaden van grote afbeeldingen mogelijk te maken. Daarom wordt er ook voorzien in een mogelijkheid om afbeeldingen op CD of DVD aan te leveren aan de beheerder van de applicatie, zodat hij of zij de afbeeldingen kan invoeren. Het toevoegen van afbeeldingen gebeurt dan server side.

3.3.2 Server-side systeemarchitectuur

3.3.2.1 Hardware en besturingssysteem

Voor het ontwikkelen van de applicatie werd er gebruik gemaakt van een webserver met de volgende karakteristieken:

Besturingssysteem

- Windows 2000 Server SP3

Hardware

- Pentium IV 1.5 GHZ
- 1 GB RAM

3.3.2.2 Database Management System (DBMS)

Algemene voorstelling van de software

Het databanksysteem dat gebruikt wordt voor het ontwikkelen van deze applicatie is een product van ADLIB Information Systems (AIS), een Nederlands bedrijf dat bibliotheken, musea en archieven automatiseert, door gebruik te maken van zelf ontwikkelde databanksoftware.

Voor dit project wordt gebruik gemaakt van ADLIB Museum, een applicatie die speciaal ontwikkeld werd voor collectieregistratie in musea en die in Vlaanderen door een honderdtal musea wordt gebruikt.

ADLIB Museum is zowel bruikbaar als standalone applicatie op één pc als in een netwerkomgeving. De applicatie beschikt over een gebruiksvriendelijke grafische interface die toelaat om een groot aantal gegevens in verband met collectieregistratie in

de databank in en uit te voeren. Beeldintegratie behoort tot de standaardmogelijkheden evenals de koppeling naar externe bestanden zoals Word-documenten en Excel spreadsheets en internetpagina's.

ADLIB is geen relationeel systeem. De gegevens die behoren tot één record worden in een geheel opgeslagen en niet verdeeld over verschillende tabellen. Er is dus geen sprake van een doorgedreven normalisatie zoals dat in een relationeel systeem gebruikelijk is. Wel kunnen er links gelegd worden tussen verschillende databanken (bijvoorbeeld tussen de databank objecten en de databank personen) om redundantie te vermijden. Binnen een ADLIB applicatie wordt er niet gesproken over verschillende tabellen maar over verschillende databanken. Alle informatie zit in één databank gegroepeerd, bijvoorbeeld de databank personen en instellingen.

Wel is het sinds kort mogelijk om de ADLIB Interface te gebruiken op een relationeel systeem. De gegevens worden dan opgeslagen in een databank van een andere leverancier dan ADLIB. Recente versies van Microsoft SQL Server en ORACLE behoren hier tot de mogelijkheden. Ook in dit geval worden de gegevens echter niet opgeslagen zoals dat in een relationele databank doorgaans gebeurt. De bulk van de gegevens wordt in een XML-structuur in één veld van een record opgeslagen.

ADLIB gebruikt ook een eigen zoektaal om informatie in de databank te kunnen zoeken en maakt binnen de eigen interface geen gebruik van SQL. AIS ontwikkelde wel een ODBC-Driver, zodat het gebruik van SQL mogelijk wordt vanuit een andere toepassing. Deze driver beschikt echter over een beperkte functionaliteit. JOIN Queries waarbij data uit verschillende databanken uit ADLIB Museum dynamisch gecombineerd worden, zijn bijvoorbeeld niet mogelijk. Wel is het zo dat ADLIB voor de meeste operaties die doorgaans op een databank worden uitgevoerd eigen oplossingen heeft bedacht, zoals de ADLIB zoektaal en de programmeertaal ADAPL om de functionaliteit van ADLIB applicaties uit te breiden.

Het beheer van een ADLIB applicatie gebeurt door middel van de ADLIB Designer. De designer omvat een aantal verschillende tools om de applicatie aan te passen en om veel voorkomende taken uit te voeren. Voorbeelden daarvan zijn het toevoegen van nieuwe velden aan de databank, het creëren van indexen en het aanpassen van de user interface van de *stand alone* applicatie.

Webapplicaties met ADLIB

Om de ADLIB databanken toegankelijk te maken via een webapplicatie ontwikkelde AIS een eigen programma, dat als naam 'ADLIB Internet Server' meekreeg. Deze applicatie is beschikbaar in twee varianten. Als CGI-programma (wwwopac.exe) en als een ISAPI-programma (=Internet Server Application Programming Interface) (adlibweb.dll). Het ISAPI programma is van recentere datum en werkt sneller dan de CGI-variant. Deze laatste moet immers elke keer dat de Internet Server geraadpleegd wordt, worden opgestart en afgesloten. Het ISAPI-programma hoeft daarentegen slechts één keer te worden gestart.

Een andere optie om webapplicaties rond dit type databank te bouwen, is gebruik maken van de ADLIB ODBC Driver. Zoals gezegd zijn de mogelijkheden van deze driver vrij beperkt. Zo is het op dit ogenblik nog niet mogelijk om gegevens via deze driver in een ADLIB databank in te voeren. Het gebruik van ODBC was voor dit project dus al op voorhand uitgesloten, gezien het wegschrijven van gegevens in de databank nu éénmaal tot de doelstellingen behoort.

De mogelijkheden die de ADLIB Internet Server aanbiedt zijn behoorlijk uitgebreid. Een overzicht van de mogelijkheden is terug te vinden in de ADLIB Internet Server referentiegids die men van de website van ADLIB kan downloaden⁶.

De standaard zoekmogelijkheden waren uiteraard al vanaf de eerste versie van de Internet Server beschikbaar. De mogelijkheid om gegevens via de ADLIB Internet Server in een ADLIB databank in te voegen, is van recentere datum. Deze is pas met de release van ADLIB 5.0 ter beschikking gekomen. Bij het uittesten van de mogelijkheden van deze software binnen het kader van dit project, werd vastgesteld dat dit aspect van de software nog aan een aantal kinderziektes leed. Een lijst van de vastgestelde problemen is opgenomen in bijlage 1. Wel werd er van de kant van ADLIB goed meegewerkt om binnen relatief korte termijn een oplossing te vinden voor de problemen die aan hen vanuit het CAG werden doorgespeeld, waardoor uiteindelijk een werkende applicatie kon worden ontwikkeld. Dit bracht wel de nodige vertraging met zich mee.

Het beheer van de internetserver gebeurt niet via ADLIB Designer, maar via een xml-bestand (in oudere versies is dit het www-bestand) waarin de instellingen kunnen worden gedefinieerd. Hierin wordt onder meer bepaald waar de databank zich op de server bevindt en welke velden er na een zoekopdracht moeten worden geretourneerd vanuit een bepaalde databank.

Communicatie met de ADLIB Internet Server gebeurt steeds via een CGI-string. Deze string bestaat uit een URL die verwijst naar de Internet Server en de opdrachten die voor de internetserver bedoeld zijn. Een voorbeeld van een CGI-string gericht aan de Internet Server ziet er uit als volgt:

<http://www.mijnmuseum.com/webfiles/wwwopac.exe?DATABASE=mijnmuseumweb&FLD1=creator&VAL1=jansen&TRC1=on>

In deze string wordt aangegeven dat er moet gezocht worden in de databank mijnmuseumweb naar records waar in het veld creator de term 'jansen' voorkomt. De TRC aanduiding wijst er op dat ook termen als jansens, waarbij er na de term jansen nog andere tekens volgen mogen worden opgehaald.

De CGI string kan via een GET of een Post Request vanuit een browser rechtsreeks naar de ADLIB Internet Server verstuurd worden. Een andere optie is om de string vanuit de browser te versturen naar een middleware applicatie die dan de communicatie met de databank verzorgt.

Zowel het zoeken als het wegschrijven van informatie gebeurt op deze manier. Bij het invoeren van informatie moeten de in te voeren gegevens wel in xml opmaak aan de CGI-string worden toegevoegd.

3.3.2.3 Ontwikkelingsomgeving

De applicatie werd ontwikkeld door gebruik te maken van de volgende technologieën:

⁶ Zie <http://www.adlibsoft.nl> . In de rubriek support vindt men het onderdeel handleidingen. In dit onderdeel vindt u ook de referentiegids voor de Internet Server.

- **ASP/JSCRIPT**

ASP (Active Server Pages) is een krachtige technologie van Microsoft die ontworpen werd om dynamische webpagina's te creëren. ASP pagina's worden geschreven in een scripttaal als VBScript of JScript. ASP pagina's die door een gebruiker worden opgevraagd voeren een bepaalde actie uit zoals het ophalen van gegevens uit een databank en bieden de gebruiker dan het resultaat van de actie aan in een voor hem of haar zinvolle vorm, bijv. als een mooi opgemaakte HTML-pagina. De opvolger van ASP is ASP.NET.

In de applicatie wordt ASP 3.0 gebruikt als ontwikkelingsomgeving omdat de knowhow hiervan reeds beschikbaar was binnen het CAG. JScript werd gebruikt als scriptingtaal. De verschillende ASP pagina's vormen de motor van de applicatie die de verschillende taken uitvoert: data ophalen uit de databank via de ADLIB Internet Server, data wegschrijven naar de databank via de ADLIB Internet Server, afbeeldingen bewerken en opslaan, controles uitvoeren, XML en XSLT documenten aan elkaar koppelen etc.

- **XML**

XML (Extensible Markup Language) is een markuptaal die toelaat om data te structureren door middel van zelfgekozen tags (labels). Door gebruik te maken van die tags kan in de data een bepaalde hiërarchie worden aangebracht, waardoor de gegevens geschikt worden om op een bepaalde manier te manipuleren.

Het volgende voorbeeld illustreert deze structuur:

```
<?xml version='1.0' encoding='UTF-8' ?>

<bibliotheek>
  <boek>
    <auteur>Claus, Hugo</auteur>
    <titel>De geruchten</titel>
    <jaar>1996</jaar>
  </boek>
  <boek>
    <auteur>Reve, Gerard</auteur>
    <titel>De Avonden</titel>
    <jaar>1947</jaar>
  </boek>
</bibliotheek>
```

In deze applicatie wordt XML gebruikt als formaat om gegevens in op te slaan of in weer te geven. Enerzijds worden er statische gegevens bewaard in XML. Hierbij gaat het om zaken die zelden wijzigen, zoals menustructuren en dergelijke. Daarnaast levert de ADLIB Internet Server gegevens afkomstig uit de databank op in een XML formaat.

- **XSLT**

XSLT (Extensible Stylesheet Language for Transformations) is een taal die XML in elk tekst gebaseerd formaat kan omzetten, bijvoorbeeld in HTML. XSLT wordt gebruikt om de gegevens afkomstig uit statische XML documenten en uit de databank om te zetten in een mooi opgemaakt HTML document.

Logica, data en opmaak zitten met andere woorden strikt gescheiden. De gegevens worden bewaard in statische of dynamische XML, de opmaak gebeurt via XSLT en ASP levert de businesslogica die alles samen brengt.

Om bepaalde specifieke taken uit te voeren werd bijkomend een beroep gedaan op extra software:

- **MSXML**

MSXML is de XML parser en XSLT processor van Microsoft. Dit houdt in dat MSXML enerzijds controleert of XML documenten geldig zijn opgemaakt volgens de vereisten die aan geldige XML worden gesteld. Anderzijds zorgt MSXML er voor dat de XML via XSLT getransformeerd kan worden, bijvoorbeeld naar HTML.

Meer info over MSXML is te vinden op de website van Microsoft: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnanchor/html/anch_xmlprod.asp

- **CSImageFile**

CSImageFile is een COM object dat de bewerking van verschillende afbeeldingsformaten toelaat. Dit stukje software werd aangekocht om een dynamische bewerking van afbeeldingen mogelijk te maken.

Meer info: <http://www.chestysoft.com/imagefile/default.asp>

- **ABCUpload**

ABCUpload is een ActiveX component die toelaat om veilig bestanden up te loaden van een webbrowser naar een webserver. Een gratis versie van deze software is beschikbaar voor non-profit organisaties indien er op de site van de organisatie een link wordt gelegd naar de website van de producent.

Meer info: <http://www.websupergoo.com/abcupload-1.htm>

3.3.3 Verantwoording van de gemaakte keuzes

De keuze voor deze architectuur kan op een aantal manieren worden verantwoord. Uiteraard is geen enkel systeem zaligmakend en zijn er zowel voor- als nadelen. We zetten deze hieronder op een rijtje.

3.3.3.1 Voordelen

- Deze cliënt-server-oplossing is schaalbaar. Op dit moment werkt de applicatie met een tweelagige architectuur, waar er twee computersets gebruikt worden, één cliënt en één server. Zonder al te veel problemen zou er echter één of meerdere servers kunnen bijgeplaatst worden, waarbij elke server dan een specifieke taak heeft. De ene server zorgt dan bijvoorbeeld voor de uitvoering van de toepassingslogica, de andere voor de databasetoegang en een derde voor de opslag van de afbeeldingen. Hierdoor kan elke server afgestemd worden op een specifieke taak en daardoor deze taak ook beter uitvoeren.
- De toepassing is bruikbaar voor cliënten van verschillende types. Zowel browsers die draaien op een Windows-computer als op Apple Macintosh zouden in principe met de applicatie moeten kunnen werken zonder dat er aanpassingen hoeven te gebeuren.
- Het gebruik van de toepassing is niet gebonden aan één computer of één persoon. Zeker in een context van een vrijwilligersorganisatie is dit interessant, omdat het werk dan over verschillende personen kan worden verdeeld, die dat dan mogelijk thuis uitvoeren.
- De toepassing is 24 uur op 24 beschikbaar.
- Het gebruik van de internetstandaarden zoals HTML, XML, Javascript etc. maakt een duidelijke scheiding mogelijk tussen de presentatielogica, de toepassingslogica en de logica voor gegevenstoegang.
- Het gebruik van ADLIB als DBMS heeft het voordeel dat er gewerkt wordt met een systeem dat zijn deugdelijkheid heeft bewezen. In een vergelijkende test van systemen voor collectiemanagement die door het Canadian Heritage Information Network (<http://www.chin.gc.ca>) werd georganiseerd, scoorde ADLIB alleszins behoorlijk. ADLIB werkt met de standaarden die gelden in de museumwereld en wordt ook door een aanzienlijk aantal grotere, professionele musea gebruikt. Zeker in Vlaanderen, na de inspanningen die de Vlaamse Gemeenschap een aantal jaren geleden leverde om het systeem ter beschikking te stellen aan de geïnteresseerde musea.
- Een van de directe voordelen van een applicatie bouwen rond de ADLIB Internet Server is dat licentiekosten kunnen worden gereduceerd. Men koopt immers één licentie aan van de databank en één licentie van de ADLIB Internet Server met een onbeperkt aantal gebruikers. Op die manier kan men met een onbeperkt aantal gebruikers via de Internet Server op één databank werken

3.3.3.2 Nadelen

- Het gebruiksgemak van de applicatie wordt voor een stuk beïnvloed door de snelheid van de netwerkverbinding. Deze kan beïnvloed worden door factoren zoals het aantal mensen dat gelijktijdig gebruik maakt van het netwerk,

onderhoudswerken die door de leverancier van de diensten worden gedaan. Deze factoren zijn niet te beheersen.

- De cliënt-server architectuur zoals deze binnen deze opstelling wordt gebruikt, vraagt om een netwerkverbinding. De toepassing kan niet als stand alone applicatie werken.
- Ongeacht het feit dat browsers verondersteld worden om zoveel mogelijk gebruik te maken van de standaarden, kan het één en het ander wel eens verschillen tussen de verschillende producten waardoor de mogelijkheid bestaat dat sommige zaken binnen een bepaalde browser niet optimaal functioneren. Dit kan worden opgevangen door zoveel mogelijk tests uit te voeren en door het gebruik van browserspecifieke mogelijkheden te vermijden.
- De ADLIB software beschikt over een aantal eigenschappen die menig ontwikkelaar bij een eerste onderzoek de wenkbrauwen doet fronsen. In de eerste plaats het niet-relatieve karakter van de database. Het feit dat in de meest recente versie van ADLIB kan gewerkt worden met een standaard relationeel DBMS, verandert hier eigenlijk niet zo veel aan, gezien de structurering niet volgens de elementaire relationele regels verloopt. Het relationele model is natuurlijk niet heilig en het systeem bewijst dat het wel werkt. Een nadeel aan ontwikkelen met ADLIB is wel dat er weinig andere ontwikkelaars zijn die met het systeem werken. Meestal worden aanpassingen aan het systeem door ADLIB zelf uitgevoerd zodat er bij de meeste gebruikers van het systeem eigenlijk weinig knowhow voorhanden is over andere zaken dan het gebruik.
- De schrijfmogelijkheden van de ADLIB Internet Server, die voor dit project cruciaal zijn, werkten bij de aanvang van het project niet naar behoren. Enkele kinderziekten staken de kop op. Deze werden er na verloop van tijd wel uitgehaald, maar zorgden toch voor enige vertraging. Aan de snelheid waarmee schrijfacties door de Internet Server worden uitgevoerd kan ook nog worden gesleuteld.

De nadelen wegen echter niet op tegen de talrijke voordelen.

3.4 Fysisch design

3.4.1 Fysisch design database

3.4.1.1 ADLIB Museum Plus

De applicatie ADLIB Museum bestaat uit negen verschillende databanken waarin informatie wordt opgeslagen die relevant is voor objectbeschrijvingen. Deze databanken zijn de objectcatalogus, beeldreproducties, binnenkomende bruiklenen, uitgaande bruiklenen, archiefstukken, tentoonstellingen, personen en instellingen, thesaurus en valuta.

Voor het opslaan van de gegevens voor de applicatie die in dit project wordt ontwikkeld, zijn er vier van deze databanken relevant, namelijk objectcatalogus, beeldreproducties, personen en instellingen en thesaurus. De volgende tabel geeft weer welke gegevens er in welke databank worden opgeslagen.

Databank	Beschrijving gegevens
Objectcatalogus	Gegevens die specifiek zijn voor een bepaald object
Beeldreproducties	Gegevens over afbeeldingen die aan een object gekoppeld kunnen zijn
Personen en instellingen	Gegevens over personen en instellingen die aan een object gekoppeld kunnen zijn
Thesaurus	Termen die aan meerdere objectbeschrijvingen kunnen worden gekoppeld

De objectcatalogus kan in feite beschouwd worden als de hoofddatabank. De andere zijn randdatabanken die gebruikt worden om redundantie te vermijden.

De volgende diagrammen verduidelijken dit:

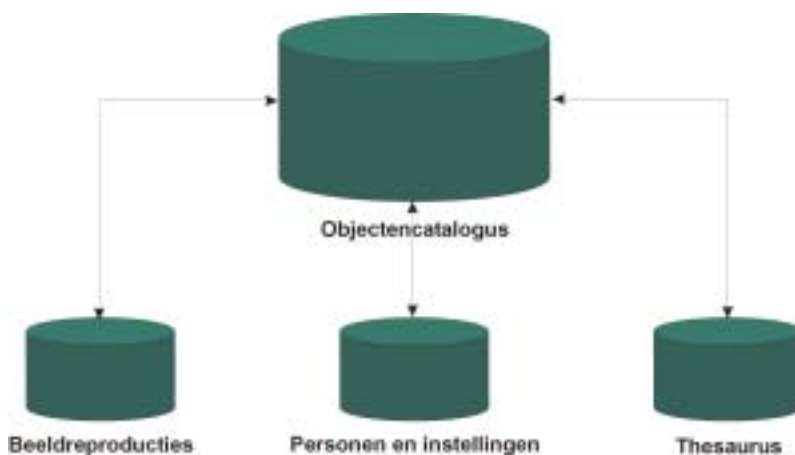


Diagram 7 : Voorstelling gebruikte databanken binnen ADLIB Museum

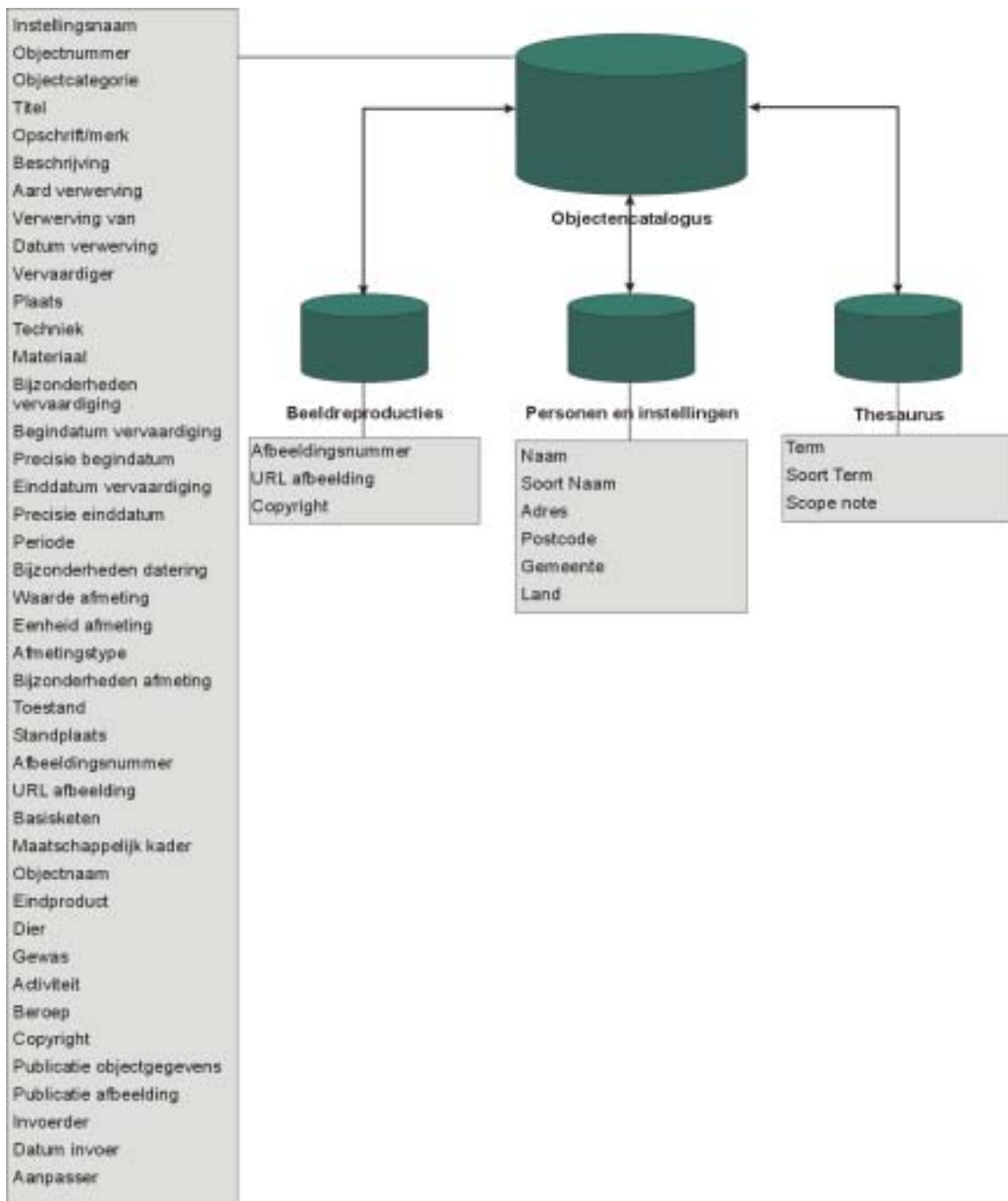


Diagram 8 : Voorstelling gebruikte databanken in ADLIB Museum met hun velden

De relaties tussen de afzonderlijke velden worden niet in het diagram weergegeven, omdat het diagram anders te complex zou worden. In de volgende tabel wordt verduidelijkt welke de eigenschappen zijn van de verschillende velden uit de objectencatalogus (J= Ja, N= Neen).

label	herhaalbaar	groep	link	controle
preref	N	N	N	N
instelling	N	N	Personen (naam)	verplicht
objectnummer	N	N	N	verplicht uniek
objectcategorie	N	N	Thesaurus (term + sort term: objectcategorie)	verplicht
titel	J	N	N	verplicht
opschrift/merk	N	N	N	N
beschrijving	N	N	N	N
aard verwerving	N	N	Thesaurus (term + soort term: verwervingsmethode)	verplicht
verwerving van	N	N	Personen (naam)	N
datum verwerving	N	N	N	geldigheid
vervaardiger	J	J	Personen (naam)	N
plaats	J	J	Thesaurus (term + soort term: plaats)	N
techniek	J	N	Thesaurus (term + soort term: techniek)	N
materiaal	J	N	Thesaurus (term + soort term: material)	N
bijzonderheden vervaardiging	N	N	N	N
begindatum vervaardiging	N	N	N	geldigheid
precisie begindatum	N	N	N	N
einddatum vervaardiging	N	N	N	geldigheid
precisie einddatum	N	N	N	N
periode	N	N	N	N
bijzonderheden datering	N	N	N	N
waarde afmeting	J	J	N	geldigheid
eenheid afmeting	J	J	Thesaurus (term + soort term: eenheid)	N
afmetingstype	J	J	Thesaurus (term + soort term: afmeting)	N
bijzonderheden afmeting	J	J	N	N
toestand	N	N	Thesaurus (term + soort term: toestand)	verplicht
standplaats	N	N	Thesaurus (term + soort term: standplaats)	N
afbeeldingsnummer	J	J	Beeldreproducties	N

			(Afbeeldingsnummer)	
URL afbeelding	N	J	Beeldreproducties (URL Afbeelding)	N
basisketen	N	N	Thesaurus (term + soort term: basisketen)	N
maatschappelijk kader	N	N	Thesaurus (term + soort term: basisinsteek)	N
objectnaam	N	N	Thesaurus (term + soort term: objectnaam)	verplicht
eindproduct	N	N	Thesaurus (term + soort term: eindproduct)	N
dier	J	N	Thesaurus (term + soort term: dier)	N
gewas	N	N	Thesaurus (term + soort term: gewas)	N
activiteit	N	N	Thesaurus (term + soort term: activiteit)	N
beroep	N	N	Thesaurus (term + soort term: beroep)	N
copyright	N	N	N	verplicht
publicatie objectgegevens	N	N	N	N
publicatie afbeelding	N	N	N	N
invoerder	N	N	N	N
datum invoer	N	N	N	N
aanpasser	J	N	N	N
datum aanpassing	J	N	N	N

De links die tussen de collectiedatabank en andere databanken worden aangebracht, worden gerealiseerd door het unieke nummer dat hoort bij elk record waarnaar verwezen wordt (de priref), te bewaren in een apart linkreferentieveld. Het systeem zorgt er voor dat elke keer een bepaald record geopend wordt, de gelinkte gegevens uit de andere databank worden opgehaald op basis van het unieke nummer. Er worden met andere woorden twee velden voorzien. Één voor het gelinkte gegeven en één voor de referentie.

In de volgende tabel wordt verduidelijkt welke de eigenschappen zijn van de verschillende velden uit de databank Beeldreproducties.

label	herhaalbaar	groep	link	controle
Afbeeldingsnummer	N	N	N	verplicht
URL Afbeelding	N	N	N	verplicht
Copyright	N	N	N	verplicht

In de volgende tabel wordt verduidelijkt welke de eigenschappen zijn van de verschillende velden uit de databank Personen en Instellingen.

label	herhaalbaar	groep	link	controle
Naam	N	N	N	verplicht
Soort naam	N	N	N	verplicht
Adres	N	J	N	N
Postcode	N	J	N	N
Gemeente	N	J	N	N
Land	N	J	N	N

In de volgende tabel wordt verduidelijkt welke de eigenschappen zijn van de verschillende velden uit de databank Thesaurus.

Label	herhaalbaar	groep	link	controle
Term	N	N	N	verplicht
Soort term	N	N	N	verplicht
Scope note	N	N	N	N

3.4.1.2 SQL SERVER

Omwille van performantie redenen werd aan het begin van het project beslist om de gegevens van de applicatiegebruikers tijdelijk onder te brengen in een afzonderlijke database. De reden hiervoor was dat de database met gebruikersgegevens op elke pagina wordt geraadpleegd en de verbinding met de server op dat moment overbelast was. Dat had tot gevolg dat de pagina's minder snel werden weergegeven. Daarom werd een SQL-server databank aangemaakt met hierin de gegevens van de gebruikers.

De gegevens die hierin werden ondergebracht zijn de volgende:

label	herhaalbaar	groep	link	controle
ID	N	N	N	verplicht
Paswoord	N	N	N	verplicht
Gebruikersnaam	N	N	N	verplicht
Email	N	N	N	verplicht

Bij een normale netwerkverbinding kunnen deze gegevens even goed in de databank "Personen en Instellingen" van ADLIB Museum worden ondergebracht.

3.4.2 Fysisch design toepassing

Om inzicht te geven hoe de belangrijkste processen van de applicatie werken, zullen de verschillende processen die in de DFD's werden aangehaald meer in detail en meer technisch besproken worden. Vooraleer de afzonderlijke processen meer in detail te bespreken, zal eerst de algemene opbouw van de toepassing worden toegelicht.

Het is niet de bedoeling om elke regel programmeercode waaruit de applicatie bestaat hier weer te geven. Het opzet is voornamelijk om de door een bespreking van enkele algemene technische principes de werking van de toepassing te verduidelijken.

3.4.2.1 Algemene opbouw

Zoals bij de bespreking van de systeemarchitectuur al werd aangehaald wordt er gebruik gemaakt van ASP-Jscript, XML en XSLT voor de ontwikkeling van de applicatie. De functies zoals het leggen van de connectie met de Internet Server, zoeken, aanpassen en verwijderen van gegevens kregen elk hun eigen ASP pagina. Naar deze pagina's kunnen dan bepaalde parameters worden doorgestuurd die nodig zijn voor het uitvoeren van de taak. Functies die door verschillende processen werden gebruikt werden gegroepeerd in aparte pagina's, zodat deze via een include hergebruikt konden worden. XML werd gebruikt om definities in op te slaan. Bijvoorbeeld van de formulieren of menu's. Dit heeft als voordeel dat deze snel kunnen worden aangepast.

De configuratie van de Internet Server zelf gebeurt via een apart XML-bestand. Hier moet gedefinieerd worden welke databanken er via de Internet Server beschikbaar zijn en welke operaties op deze databanken toegestaan zijn. Ook wordt in dit document bepaald welke velden een bepaalde zoekopdracht opleveren. Dit wordt met andere woorden niet geregeld door parameters in de CGI-zoekstring mee te geven. Schrijfoopdrachten kunnen worden toegestaan door de write_allowed parameter op true te zetten. Dat kan ingesteld worden voor elke databank afzonderlijk of voor alle databases tegelijkertijd.

Een voorbeeld van een dergelijk webconfiguratiebestand is het volgende:

```
<?xml version="1.0" encoding="UTF-8" ?>
<webConfiguration>

  <!-- Global settings -->
  <globalConfiguration>
    <databasepath>../mapdatabank/data</databasepath>
    <logfile>../mapdatabank/logfiles/weblog.csv</logfile>
    <debug>0</debug>
    <language>1</language>
  </globalConfiguration>

  <!-- Databases -->
  <databaseConfiguration database="objectcollectie">
    <database>collect</database>
    <brieffields>%0 BA IN B1 TI OC</brieffields>
    <write_allowed>>true</write_allowed>
  </databaseConfiguration>
</webConfiguration>
```

```
</databaseConfiguration>  
</webConfiguration>
```

In deze configuratie is het toegestaan om één databank via de ADLIB Internet Server te benaderen. Schrijven in de databank is toegestaan en een zoekactie op deze databank levert de velden op die in de databank de volgende indentificatietags kregen: %0, BA, IN, B1, TI en OC.

3.4.2.2 Opzoeken informatie collectie

In deze paragraaf zal besproken worden hoe het formuleren van een zoekvraag aan de databank, het ophalen van de informatie en de presentatie van het gevonden resultaat technisch in zijn werk gaat.

Om gegevens in een ADLIB databank op te zoeken, moet er een CGI-string naar de ADLIB Internet Server worden gestuurd, met daarin de URL van de ADLIB Internet Server en de zoekcriteria waaraan het zoekresultaat moet voldoen. Het versturen van een dergelijk CGI-string gebeurt via een formulier of via een link. Er werd voor geopteerd om de zoekvraag niet rechtstreeks naar de Internetserver te sturen, maar om een ASP pagina te voorzien die fungeert als een soort doorgeefluik met de volgende taken: een zoekvraag ontvangen vanuit een link of een formulier, de zoekvraag vertalen naar een voor de Internet Server begrijpbare zoekopdracht, het versturen van de zoekopdracht naar de ADLIB Internet Server en het verzorgen van de koppeling tussen het zoekresultaat in XML-formaat met het juiste XSLT-stylesheet.

Het voordeel van deze manier van werken zit hem eerst en vooral in de extra veiligheid die gecreëerd wordt. De Internet Server wordt immers niet rechtstreeks aangesproken, zodat de locatie van de Internet Server voor de gebruiker verborgen blijft. Ook geeft deze manier van werken extra flexibiliteit om de XML-resultaten uit de databank te bewerken en er makkelijk de juiste lay-out op toe te passen.

Een voorbeeld kan de manier van werken verduidelijken.

Stel dat een gebruiker op zoek wil gaan naar alle objecten waarbij er in de titel het woord 'koe' voorkomt. Hij kan dit doen door in het zoekvak van een zoekformulier de term 'koe' in te vullen. Als hij op de knop 'verzenden' klikt wordt de volgende CGI-String naar de ASP-pagina zoeken.asp verstuurd:

```
http://adresvandewebsite/zoeken.asp?TI=koe&stylesheet=gallerij.
```

Op basis van de parameters die in de string worden opgenomen, wordt bepaald in welke database wordt gezocht, in welk veld, naar welke term en welke lay-out het zoekresultaat moet krijgen.

Op basis van het parameter-paar TI=koe, wordt de zoekvraag samengesteld die voor de ADLIB Internet Server is bedoeld.

Deze ziet er uit als volgt:

```
http://adresadlibinternetserver/adlibweb.dll?Query?database=databasenaam&SEARCH=%28%20TI%20%3D%20%27koe%27%29&STARTFROM=1&LIMIT=54&SEARCHMODE=webor
```

In dit voorbeeld worden in de string de volgende parameters opgenomen:

DATABASE: deze parameter bevat de naam van de databank waarin gezocht moet worden;

SEARCH: bevat de eigenlijke zoekstring, die geëscaped werd⁷;

STARTFROM: bevat de postie in het zoekresultaat vanwaar er moet begonnen worden.;

LIMIT: bepaald het aantal records dat per keer uit de databank wordt opgehaald;

SEARCHMODE: bevat de zoekmodus die wordt gebruikt

Vervolgens wordt de zoekvraag verstuurd naar de Internet Server. Hiervoor wordt gebruikt gemaakt van het XML HTTP Request object. Dit object wordt gebruikt om een http request naar de ADLIB internet Server te sturen en het XML resultaat van dit request op te vangen.

In JScript ziet een GET-request voor de hoger vermelde CGI-string er uit als volgt:

```
var xmlhttp1, url, objXML;

// Aanmaak XML HTTP Request object
xmlhttp1 = Server.CreateObject ("Msxml2.XMLHTTP.4.0");

//Toekennen CGI-string aan de variabele url1
url =
"http://adresinternetserver/adlibweb.dll?Query?database=databasenaam&SEARCH=%28%20TI%20%3D%20%27koe%27%29&STARTFROM=1&LIMIT=54&SEARCHMODE=webor";

//Openen van de connectie met de Internet Server
xmlhttp1.Open("GET", url, false);

//Verzenden en ontvangen van de data
xmlhttp1.Send ();

// ResponseXML geeft het resultaat terug als een XML Document Object
var objXML = xmlhttp1.ResponseXML;
```

Het XML-resultaat dat de ADLIB Internet Server oplevert ziet er mogelijk uit als volgt:

```
<?xml version="1.0" encoding="UTF-8" ?>
<adlibXML xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://www.adlibsoft.com/adlibXML.xsd">
```

⁷ Het 'escapen' van een CGI-string houdt in dat van elk teken de hexadecimale ASCII-waarde moet worden genomen met een %-teken ervoor. In Jscript kan men hiervoor de functie escape() gebruiken.

```

<recordList>
  <record>
    <priref>10000022</priref>
    <institution.name>
      Centrum voor Historische Pedagogiek
    </institution.name>
    <object_name>wandplaat</object_name>
    <object_number>00000022</object_number>
    <title>
      Pedagogische wandplaat : ademhalingsstelsel
      van een koe
    </title>
    <object_category>schilderij</object_category>
  </record>
</recordList>
<diagnostic>
  <xmltype>UNSTRUCTURED</xmltype>
  <hits>1</hits>
  <search>TI = 'koe' startfrom 1 limit 54</search>
  <sort />
  <limit>1</limit>
  <hits_on_display>1</hits_on_display>
  <first_item>1</first_item>
  <backward>0</backward>
  <forward>0</forward>
  <databasepath>pad database</databasepath>
  <dbname>collect</dbname>
  <dsname />
  <cgistring url=" http://adresadlibinternetserver/adlibweb.dll?Query?database=databasenaam&SEARCH=%28%20TI%20%3D%20%27koe%27%29&STARTFROM=1&LIMIT=54&SEARCHMODE=webor
">
  <param name="database">collist</param>
  <param name="SEARCH">( TI = 'koe')</param>
  <param name="STARTFROM">1</param>
  <param name="LIMIT">1</param>
  <param name="SEARCHMODE">webor</param>
  </cgistring>
  <language>1</language>
  <epoch>1125656810</epoch>
  <error />
</diagnostic>
</adlibXML>

```

Dit resultaat wordt vervolgens gekoppeld aan een XSLT-stylesheet. Dit ziet er mogelijk uit als volgt:

```

<?xml version='1.0' encoding="UTF-8" ?>
<!DOCTYPE xsl:stylesheet [<!ENTITY nbsp "&#160;">]>

<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:msxsl="urn:schemas-microsoft-com:xslt"
  xmlns:user="http://www.cagnet.be/cagnet-ns">

<xsl:output method="html" />

<xsl:template match="/">
<html>
<head>
  <title>Objectbeschrijving</title>
</head>
<body>
  <xsl:for-each select="/adlibXML/recordList/record">
    <xsl:if test="priref != " ">
      Instellingsnaam: <xsl:value-of select="institution.name" /><br />
      Titel:<xsl:value-of select="title" /><br />
      Objectnaam:<xsl:value-of select="object_name" />
    </xsl:if>
  </xsl:for-each>
</body>
</html>
</xsl:template>
</xsl:stylesheet>

```

De koppeling van het XML document en het stylesheet kan binnen JScript mogelijk als volgt:

Eerst wordt het XSLT stylesheet opgehaald:

```

var xmlhttp2 = Server.CreateObject ("Msxml2.XMLHTTP.4.0");
var url2 = "http://pad_naar_het_xslt_document/test.xsl";
xmlhttp2.Open("GET", url2, false);
xmlhttp2.Send ();
var objXSL = xmlhttp2.ResponseXML;

```

Vervolgens kunnen XML-data en XSLT-stylesheet aan elkaar gekoppeld worden:

```

Response.Write(objXML.transformNode(objXSL));

```

Het resultaat levert HTML-code op die er uitziet als volgt:

```

<html xmlns:msxsl="urn:schemas-microsoft-com:xslt"
xmlns:user="http://www.cagnet.be/cagnet-ns">
<head>
  <META http-equiv="Content-Type" content="text/html">
  <title>Objectbeschrijving</title>

```

```
</head>
<body>
  Instellingsnaam: Centrum voor Historische Pedagogiek<br>
  Titel:Pedagogische wandplaat : ademhalingsstelsel van een koe<br>
  Objectnaam:wandplaat
</body>
</html>
```

In de browser wordt deze code als volgt weergegeven:

```
Instellingsnaam: Centrum voor Historische Pedagogiek
Titel:Pedagogische wandplaat : ademhalingsstelsel van een koe
Objectnaam: wandplaat
```

Deze techniek wordt doorheen de hele applicatie gebruikt om de gegevens afkomstig uit de databank weer te geven.

3.4.2.3 Beheren informatie collectie

3.4.2.3.1 *Toevoegen Nieuwe Informatie Collectie*

Het opzet voor het toevoegen van informatie aan de databank is gelijkaardig aan het opzoeken van informatie. Toch is dit proces wat complexer omdat er extra controles moeten worden toegepast. De verschillende stappen in het toevoeg-proces zullen achtereenvolgens besproken worden.

Indien een gebruiker er voor opteert om een nieuwe objectbeschrijving aan de databank toe te voegen, krijgt hij een lege objectfiche met alle mogelijke invulvelden ter beschikking. De definitie van deze fiche, met de velden en hun eigenschappen, zit opgeslagen in een xml-document. Op het moment dat een gebruiker een lege fiche opvraagt, wordt de definitie geladen en worden de juiste functionaliteiten per veld via een XSLT-stylesheet aangeboden in HTML, in combinatie met JavaScript aan de gebruiker. Deze manier van werken heeft als voordeel dat de velddefinities makkelijk kunnen worden aangepast en dat makkelijk nieuwe velden kunnen worden toegevoegd.

Vervolgens kan de gebruiker de fiche invullen. De waarden die de gebruiker in de velden invoert, worden niet dadelijk in de databank weggeschreven, maar in een tijdelijke buffer opgeslagen in een sessieobject dat voor elke gebruiker wordt aangemaakt. De reden hiervoor is, dat het wegschrijven van informatie in de databank een vrij tijdrovende activiteit is, waardoor het aantal schrijfacties op de databank best kan worden beperkt. De velden worden in de interface verdeeld over verschillende tabbladen en bij elke wissel van tabblad, worden de ingevoerde gegevens gebufferd.

Als de objectbeschrijving voltooid is, kan de gebruiker de gegevens voordragen ter controle. De controle gebeurt server side op basis van de gegevens in de buffer. Welke controle er gebeurt op een bepaald veld is afhankelijk van de velddefinitie. De controlefuncties bevinden zich in een afzonderlijke Active Server Page (ASP). Indien de controle voor één van de velden faalt, krijgt de gebruiker de mogelijkheid om dadelijk de gevraagde aanpassingen door te voeren en de controle opnieuw te starten. Indien de controles voor alle velden slagen, krijgt de bezoeker de mogelijkheid om alle ingevoerde informatie te overlopen en dan definitief op te slaan.

Het invoeren van informatie in ADLIB via de Internet Server gebeurt op ongeveer dezelfde manier als het zoeken van informatie. In de CGI-string die naar de Server wordt gestuurd, kan een extra parameter DATA worden opgenomen, waaraan men een CGI escaped_UTF-8 adlibXML document kan toekennen⁸. Bij het opslaan van de informatie wordt de goedgekeurde informatie uit de buffer gehaald en opgeslagen in de structuur die een ADLIB XML-document kenmerkt.

Concreet ziet deze structuur er uit als volgt. Stel dat men een objectbeschrijving wilt invoeren van een foto met als titel "Kempisch boerenpaard rond 1900", dan ziet de ADLIBXML-structuur van deze informatie er uit als volgt:

```
<adlibXML>
<recordList>
  <record>
    <object_category>foto</ object_category >
    <title> Kempisch boerenpaard rond 1900</title>
  </record>
</recordList>
</adlibXML>
```

In een CGI-string die geëscaped werd, heeft dit de volgende vorm:

```
http://adresadlibinternetserver/adlibweb.dll?Query?database=databasenaam&
%3CadlibXML%3E%3CrecordList%3E%3Crecord%3E%3Cobject_category
%3Efoto%3C/object_category%3E%3Ctitle%3EKempisch%20boerenpaard
%20rond%201900%3C/title%3E%3C/record%3E%3C/recordList%3E
%3C/adlibXML%3E
```

Om de connectie met de InternetServer tot stand te brengen, wordt op dezelfde manier gewerkt als beschreven staat bij het zoeken naar informatie, dus door middel van het XML HTTP Request object. Wel is het noodzakelijk dat men als methode bij het leggen van de connectie altijd POST gebruikt in plaats van GET. Het aantal karakters dat men aan een GET Request kan meegeven, is immers beperkt tot ongeveer 1024 tekens. Als men grote hoeveelheden tekst wenst in te voeren, kan dit bij het gebruik van een GET method problemen opleveren.

Na het schrijven van informatie geeft de Internet server een XML-resultaat terug met daarin de gegevens die in de databank werden ingevoerd. Op basis hiervan wordt een

⁸ Dit houdt in dat eerst de tekens met een accent, zoals ë, à etc moeten vervangen worden door de numerische referentie van het karakter, om ze UTF-8 compatibel te maken. Het karakter ë moet bijvoorbeeld vervangen worden door ë. Voor een lijst met numerische referenties kan u terecht op <http://www.cookwood.com/entities/>. Ook moeten de karakters &, <, > vervangen worden door hun respectievelijke karakterreferenties &, < en > of numerische referenties worden vervangen. Vervolgens moet de string ook voor CGI worden geëscaped.

boodschap voor de gebruiker gecreëerd die aangeeft of het invoeren al dan niet succesvol is verlopen.

3.4.2.3.2 *Aanpassen Informatie Collectie*

Het proces om objectgegevens aan te passen is een combinatie van het opzoeken en het invoeren van gegevens.

Een gebruiker zoekt eerst via het zoekmenu een bepaald record op. Vervolgens kan hij ervoor opteren om de objectgegevens die in het record opgeslagen zitten aan te passen.

De keuze voor deze mogelijkheid heeft tot gevolg dat binnen de applicatie de volgende operaties worden gestart:

1. Er wordt gecontroleerd of er zich op het record in kwestie een lock bevindt. Indien dit niet het geval is, wordt er een lock op het record geplaatst, anders wordt gemeld dat het record niet voor aanpassing beschikbaar is.
2. De selectie van deze optie heeft als resultaat dat alle gegevens van het geselecteerde record uit de databank worden opgehaald.
3. Vervolgens wordt de definitie van het invulformulier voor de objectbeschrijving geladen.
4. Daarna worden de opgehaalde gegevens gecombineerd met het lege invulformulier. Ze worden op de juiste plaats in het formulier ingevuld.
5. De ingevulde veldstructuur wordt gebufferd in het sessieobject.
6. Het ingevulde XML-document wordt vormgegeven en gepresenteerd aan de gebruiker.

Vervolgens kan de gebruiker de gegevens in het formulier aanpassen, laten controleren en opnieuw opslaan in de databank. Dit verloopt op dezelfde manier als beschreven werd bij het toevoegen van nieuwe objectgegevens. Een bijkomende actie is evenwel dat na het uitvoeren van de schrijfo opdracht, ook het aanwezige lock op het record wordt verwijderd.

Het enige verschil hier is dat er in het ADLIB XML bestand nu ook een veld wordt meegegeven waarin de unieke referentie van een objectbeschrijving zit (= priref). De ADLIB Internet Server kan op basis hiervan vaststellen dat het om een update van eerder ingevoerde collectieinformatie gaat en niet om een toevoeging van een nieuwe objectbeschrijving.

Een voorbeeld kan dit illustreren:

Stel dat men het record dat in de vorige paragraaf werd aangehaald als voorbeeld van het invoerproces, zou willen aanpassen. De datering die in het titelveld werd aangehaald, blijkt fout te zijn. Men wil de vermelding "Kempisch boerenpaard rond 1900" vervangen door "Kempisch boerenpaard rond 1950". Als het record wordt opgehaald, blijkt dat het de unieke referentie "00000001" heeft gekregen.

Het AdlibXML-document dat aan de de CGI-string wordt meegegeven in de DATA parameter ziet er dan uit als volgt:


```

<adlibXML>
<recordList>
  <record>
    <priref>00000001</priref>
    <object_category>foto</ object_category >
    <title> Kempisch boerenpaard rond 1950</title>
  </record>
</recordList>
</adlibXML>

```

3.4.2.3.3 Verwijderen Informatie Collectie

Het verwijderen van objectgegevens uit de databank gebeurt eveneens door het opzoeken van een specifiek record in de databank via de zoekinterface. Als het record gevonden is, kan een verwijderopdracht voor dat record worden uitgevoerd.

De opdracht die naar de ADLIB Internet Server wordt verstuurd, is gelijkaardig aan deze voor het updaten van een record. Ook hier wordt als extra veld de unieke referentie meegegeven. Alleen wordt bij een verwijderopdracht een min-teken voor de unieke referentie (=priref) geplaatst.

Stel dat men de objectbeschrijving die in de vorige paragraaf werd aangehaald, wenst te verwijderen, moet de volgende ADLIBXML aan de data-parameter van de CGI-string voor de Internet Server worden meegegeven.

```

<adlibXML>
<recordList>
  <record>
    <priref>-00000001</priref>
  </record>
</recordList>
</adlibXML>

```

3.4.2.3.4 Lock Controle

Als een record door een gebruiker met een standalone ADLIB applicatie wordt bewerkt, zal het systeem een lock op dit record plaatsen, om te verhinderen dat twee gebruikers tegelijkertijd een record kunnen aanpassen. Dat is noodzakelijk om binnen ADLIB de integriteit van de gegevens te bewaren. Als twee gebruikers gelijktijdig een record kunnen aanpassen, zou dit immers tot gevolg kunnen hebben dat de gegevens van de gebruiker die als eerste zijn gegevens opslaat, verloren zouden gaan. Locks vangen dit probleem op.

Om Lock controle ook binnen een webapplicatie mogelijk te maken, voorziet ADLIB in een protocol om locks op een record aan te brengen en te verwijderen via de ADLIB Internet Server. Dit protocol is niet gedocumenteerd in de ADLIB handleiding, maar is wel voorzien. In wat volgt wordt dit protocol besproken.

Deze gegevens werden ter beschikking gesteld door Hedzer Westra, ontwikkelaar bij ADLIBSoft.

Het locken van een record door middel van de ADLIB Internet Server gebeurt op dezelfde manier als elke andere actie, namelijk via het opgeven van specifieke parameters in de CGI-String die naar de Internet Server wordt verstuurd.

Benodigde parameters voor een lock-request:

database	Naam van de database
lockrec	Preref van het record waarop een lock moet worden geplaatst
locktype	Lock In deze parameter kan aangegeven worden of het gaat om een lock- of een unlock-operatie

Als resultaat geeft de internet server de volgende XML-output:

```
<locking>
  <action>lock</action>
  <lockId>id</lockId>
  <databaseName>db</databaseName>
  <databasePath>path</databasePath>
  <preref>recno</preref>
  <stat>code</stat>
</locking>
```

Als de statuscode anders is dan 0 is er een fout opgetreden.

Het formaat van de lockID ziet er als volgt uit: gebruikersnaam, '-', 5 willekeurige cijfers

Bijvoorbeeld:

```
<locking>
  <action>lock</action>
  <lockId>IUSR_CAG.98576</lockId>
  <databaseName>myndb</databaseName>
  <databasePath>\\CAGR\myndbA</databasePath>
  <preref>1</preref>
  <stat>0</stat>
</locking>
```

Het unlocken van een record gaat op dezelfde manier in zijn werk, alleen moeten de parameters iets anders worden ingevuld.

Benodigde parameters voor een lock-request:

database	Naam van de database
lockrec	Preref van het record waarop een lock moet worden geplaatst
locktype	unlock
lockid	ID die de Internet Server in de XML meegaf bij het plaatsen van een lock bijv. IUSR_CAG.98576 De lockid die in de XML output stond die de lockactie opleverde moet meegegeven worden om te kunnen schrijven en unlocken middels de lockid=.. CGI-parameter.

Een aantal opmerkingen:

- een geslaagde schrijfactie unlockt een record automatisch
- als een gebruiker een schrijfactie annuleert moet er altijd een unlock volgen!
- een unlock actie op een niet-gelockt record geeft code 0 (geen fout) als reactie
- een unlock actie met een verkeerde lockid geeft code 346

Meer informatie over dit onderwerp is te verkrijgen bij ADLIBsoft zelf.

Indien een unlock opdracht na het plaatsten van een lock niet wordt uitgevoerd, blijft het geplaatste lock bestaan. Daarom moest er een systeem ontwikkeld worden dat registreert welke locks door een gebruiker geplaatst worden, om deze later te kunnen verwijderen. In eerste instantie werd de registratie in het sessieobject bijgehouden. De lockregistratie werd aangemaakt op het moment dat er een record in de aanpasmode werd geplaatst en verwijderd als het record opnieuw werd opgeslagen. Ook kan de gebruiker een lock handmatig verwijderen door een record in updatemode vrij te geven, of door uit te loggen. Omdat de gebruikers dit niet altijd even strikt toepasten en de browser regelmatig sloten zonder af te loggen, leverde dit problemen op, omdat een record gelockt bleef en dan door de administrator van de applicatie handmatig moest worden verwijderd. Daarom werd een andere oplossing gezocht waarbij de geplaatste locks in een database werden bijgehouden. Deze zal in de nabije toekomst worden uitgewerkt.

3.4.2.4 Login Controle

De applicatie is enkel toegankelijk voor gebruikers met een gebruikersnaam en een correct paswoord. Als een gebruiker zijn paswoord en gebruikersnaam ingeeft, wordt gecontroleerd of dit in de SQL Server databank aanwezig is. Indien dit zo is, krijgt de gebruiker toegang tot de records in de databank die aan zijn instelling zijn toegewezen. De records van andere instellingen zijn voor hem niet toegankelijk. Sommige trefwoordenlijsten met niet-persoonlijke gegevens (zoals bijv. objectnaam) worden wel door meerdere instellingen gedeeld.

3.4.3 Fysisch design interface

Om het design van de interface te bespreken, wordt er gewerkt met screenshots van de verschillende webpagina's van de interface. Als structuur van de bespreking wordt het schema gebruikt dat bij de bespreking van het logisch design van de interface werd besproken.

3.4.3.1 Algemeen

Vooraleer de afzonderlijke schermen van de applicatie te bespreken, is het nuttig om even de globale opbouw van de interface te bekijken.

Aan de hand van het volgende scherm kan kort worden uitgelegd welke de voornaamste onderdelen zijn waaruit elk scherm bestaat:



De balk met de **algemene functies** is voor elke pagina gelijk. De functies waar deze balk naar verwijst staan los van de inhoud van een specifieke pagina.

In de balk met de algemene functies wordt gewerkt met klaplijsten. Niet alle mogelijkheden zijn onmiddellijk zichtbaar. Indien een gebruiker met de muiscursor over een functie beweegt, komen er extra mogelijkheden te voorschijn. Op die manier kan men de verschillende mogelijkheden bereiken. De functies zijn de volgende:

- **Home – Invoer:** Navigeren naar de eerste pagina van de invoerapplicatie;
- **Home – Presentatie:** Navigeren naar de eerste pagina van Het Virtuele Land; Dit is de website waarop de gegevens die men invoert, na interne controle zullen gepresenteerd worden;
- **Toevoegen – Object:** Toevoegen van een nieuwe objectbeschrijving aan de databank;
- **Toevoegen – Randgegevens - Afbelding:** Toevoegen van een nieuwe afbeelding aan de databank;
- **Toevoegen – Randgegevens – Instelling:** Toevoegen van een nieuwe instelling aan de databank;

- **Toevoegen – Randgegevens – Persoon:** Toevoegen van een nieuwe persoon aan de databank;
- **Toevoegen – Randgegevens – Thesaurus:** Toevoegen van een nieuw trefwoord;
- **Zoeken – Selectietaal:** Gegevens opzoeken door gebruik te maken van de selectietaal;
- **Zoeken – Zoekassistent:** Gegevens opzoeken door gebruik te maken van de zoekassistent;
- **Contact – Helpdesk:** Een bericht sturen naar de helpdesk;
- **Logout – Logout:** De applicatie reglementair verlaten;
- **Invulinstructie – Technisch:** De technische handleiding van deze site oproepen;
- **Invulinstructie – Inhoudelijk:** De inhoudelijke handleiding van deze site oproepen;
- **Andere – record vrijgeven :** Een record dat voor aanpassing in gebruik was terug vrijgeven.

Het is belangrijk dat men, als men de applicatie verlaat, eerst de functie logout aanklikt. Enkel zo kan men de applicatie reglementair verlaten en er voor zorgen dat de applicatie reglementair wordt afgesloten. De reden hiervoor is onder meer, dat een lock dat mogelijk werd aangebracht op een record, en dat nog niet eerder werd vrijgegeven, dan correct zal worden verwijderd.

3.4.3.2 Login

In dit scherm kan de gebruiker inloggen op basis van zijn gebruikersnaam en paswoord. Deze zijn hem of haar ter beschikking gesteld door het CAG. Nadat de gebruiker op de login knop heeft gedrukt worden de gegevens naar de server verstuurd en wordt er gecontroleerd of de combinatie gebruikersnaam-paswoord in de databank voorkomt. Indien dit het geval is wordt de gebruiker doorverwezen naar het startscherm. Indien de combinatie fout is, wordt dit aan de gebruiker gemeld en wordt hem de kans geboden om nogmaals te proberen.



3.4.3.3 Startscherm

Het startscherm biedt de gebruiker meerdere opties. Ofwel kan er met de zoekassistent of via de selectietaal gezocht worden naar gegevens die reeds eerder in de databank werden ingevoerd. Ofwel kunnen er nieuwe gegevens worden toegevoegd in vier verschillende databanken: de objectdatabank, de afbeeldingendatabank, de databank met personen en de databank met thesaurustermen. De voornaamste databank is deze met de objectbeschrijvingen. In wat volgt zal ook enkel de invoerinterface van deze databank worden besproken. De interface voor de andere databanken is immers sterk gelijkaardig. Nadat de gebruiker een keuze heeft gemaakt kan op één van de links worden geklikt.



3.4.3.4 Zoekassistent

Als er gekozen wordt om gegevens op te zoeken door middel van de zoekassistent, kan de gebruiker eerst selecteren in welke databank er gegevens moeten worden opgezocht. Nadat de databank geselecteerd is (in dit geval wordt de databank Objecten geselecteerd) kan op de knop 'volgende' worden gedrukt om naar het volgende keuzeschermb te gaan. Door op de help-knop te klikken, kan de gebruiker informatie opvragen over de bedoeling en de werking van deze pagina.



Vervolgens worden aan de gebruiker een aantal zoekvelden aangeboden, waarop de gebruiker kan gaan zoeken. Ook hier moet er weer gekozen worden uit de lijst. Per databank worden de belangrijkste velden aangeboden als zoek sleutel. De help-knop levert opnieuw informatie over de bedoeling van het scherm.



In dit geval werd er op het vorige scherm gekozen voor de zoek sleutel 'titel'. Voor deze sleutel dient een woord te worden ingevoerd, bijvoorbeeld 'koe'. Als er op de knop zoeken wordt geklikt, zullen alle records in de databank worden opgezocht, waarvan de 'titel' het woord 'koe' bevat.




3.4.3.5 Selectietaal

Een tweede zoekmogelijkheid is de selectietaal. Deze optie maakt het aan de bezoeker mogelijk om meer complexe zoekacties uit te voeren. In eerste instantie kan de bezoeker aangeven in welke databank er moet worden gezocht. Vervolgens kan hij de eigenlijke zoekvraag samenstellen, door eerst een veld aan te klikken, vervolgens een vergelijking en daarna eventueel een booleaanse operator. Op die manier wordt in het veld zoekvraag een syntactisch correcte zoekvraag samengesteld. De help-knop levert informatie op over het scherm. Door op de knop 'zoeken' te klikken wordt de zoekvraag naar de databank verstuurd.








3.4.3.6 Zoekresultaat

Het resultaat van een zoekactie is een overzicht van alle gevonden records waarin de gezochte term in het gevraagde veld of velden voorkomt. Uiteraard worden enkel die records getoond die ingevoerd werden door de leden van de instelling waar ook de gebruiker toe behoort. Records van andere instellingen worden afgeschermd. De resultaten worden standaard volgens een galerij-zicht getoond. Dit houdt in dat enkel de afbeeldingen getoond worden, zonder bijkomende informatie. Indien een andere weergave meer aangewezen is kan dit aangepast worden in de weergave-balk. Ook is het mogelijk om het aantal records dat tegelijkertijd wordt getoond aan te passen. Ook dit gebeurt bovenaan in de weergavebalk. De gebruiker heeft de beschikking over een heel aantal verschillende functies. Aan de linkerkant van het scherm staan een aantal functies opgesomd in de menubalk. Deze functies zijn de volgende:

Resultaat afdrukken	Met deze functie kan een gebruiker het zoekresultaat afdrukken. Een klik op deze functie roept een scherm op met een printassistent die toelaat om de resultaten in verschillende formaten af te drukken.
Selectie bekijken	Het is mogelijk om individuele zoekresultaten die tijdelijk werden bewaard gezamenlijk op te roepen door op het  -icoon te klikken.
Selectie verwijderen	Via deze link kunnen tijdelijk bewaarde zoekresultaten worden verwijderd.

Ook staan er onder elke afbeelding een aantal icoontjes die een bepaalde functie vertegenwoordigen. Deze zijn de volgende:

	Via dit icoon is het mogelijk om een aantal interessante zoekresultaten tijdelijk te bewaren.
	Dit icoon laat toe om de detailinformatie die bij een bepaald zoekresultaat hoort op te vragen.
	Het is mogelijk om via dit icoon een vergroting van een afbeelding en de detailinformatie van een record af te printen.

	Een bepaald record bewerken kan via dit icoon.
	Met dit icoon is het mogelijk om een bepaald record uit de databank te verwijderen.

Scrollen doorheen de gevonden zoekresultaten kan door op één van de paginanummers te klikken.



3.4.3.7 Detail

Een detailweergave van een record geeft een overzicht van de reeds ingevoerde informatie voor dat record.



3.4.3.8 Selectiescherm

Het selectiescherm geeft een overzicht van de records die door de gebruiker opgenomen zijn in zijn selectie. Het scherm is min of meer gelijkaardig aan het scherm waarin de zoekresultaten worden getoond. Ook hier zijn de iconen voorzien onder de afbeelding om bepaalde acties uit te voeren en kan de weergave worden aangepast. Vanuit dit scherm is het mogelijk om alle geselecteerde records in groep definitief uit de databank te verwijderen. Deze functie kan geactiveerd worden door op de link 'records verwijderen' te klikken die zich bevindt in de linkerkolom van het scherm.




3.4.3.9 Printassistent

De printassistent kan door de gebruiker worden opgeroepen om een zoekresultaat of een selectie van records af te printen. De gebruiker heeft de keuze tussen een aantal printformaten. Hij kan een lijst met enkel afbeeldingen afdrukken, een lijst met beknopte informatie en een lijst met alle recordinformatie.



3.4.3.10 Verwijderen

Het verwijderen van records uit de databank kan door op het -icoon te klikken dat zich onder de afbeelding bevindt. Om het record definitief te verwijderen, wordt om bevestiging van de actie gevraagd. Indien de gebruiker dan op de 'ja' knop klikt zal het record definitief uit de databank worden verwijderd.



3.4.3.11 Invoer

Als de gebruiker op het startscherm kiest voor de invoer van nieuwe gegevens krijgt hij een blanco registratieformulier ter beschikking met daarop de velden die nodig zijn om een basisregistratie van objecten mogelijk te maken.

De onderdelen van het invoerscherm

Het lege invoerscherm voor de objectgegevens ziet er uit als volgt:



Het invoerscherm bestaat uit twee delen. De linkerkolom, met als hoofding “menu object”, bevat de functies die specifiek verbonden zijn aan een invoerscherm. Op het rechterdeel van het scherm staat het eigenlijke invoerformulier waarin de gebruiker de objectbeschrijving kan invullen.

Men kan onmiddellijk zien dat het om het invoerscherm voor objectgegevens gaat aan het melkstoeltje in de linkerkolom van het scherm. Dit systeem wordt ook op de andere invoerschermen herhaald. Elk invoerscherm heeft zijn eigen symbol:



Onderdeel Afbeeldingen



Onderdeel Instellingen



Onderdeel Objecten



Onderdeel Thesaurus



Onderdeel Personen

Elk invoerformulier bestaat uit verschillende tabbladen. Elk tabblad is een groepering van invoervelden (of invoervakken) die bij elkaar horen. Zo bevat het blad identificatie een aantal velden die een bepaald voorwerp identificeren, zoals objectnummer, instellingsnaam, titel etc.


De namen van de verschillende tabbladen staan bovenaan de pagina gegroepeerd. Als de gebruiker naar één van de tabbladen wilt navigeren, kan hij op één van de namen van de tabbladen klikken. Een actief tabblad heeft bovenaan de tab een oranje streepje.

Identificatie	Verwerving	Vervaardiging	Datering	Afmetingen
Bewaring	Afbeelding	Trefwoorden	Beheer	

Verschillende types van invoervelden

Nadat een tabblad geactiveerd is, kan onmiddellijk begonnen worden met het invullen van de velden.

Op een tabblad zijn er drie verschillende types invoervelden te onderscheiden:


1. Een *gewoon invoerveld*, te herkennen aan de witte achtergrondkleur. In dit veld kan de gebruiker dadelijk de informatie intypen die hij er in kwijt wilt.
2. Een *invoerveld waaraan een interne klaplijst is gekoppeld*, waaruit een term kan worden gekozen. Men kan de lijst uitklappen door op het pijltje te klikken dat zich achteraan het veld bevindt en er vervolgens een term uit selecteren. Het is niet mogelijk om aan deze lijst termen toe te voegen.
3. Een *invoerveld waaraan een externe lijst gekoppeld is*. Deze velden zijn grijs van kleur en laten niet toe dat de gebruiker er rechtstreeks informatie invult. Men kan de externe lijst oproepen door op het lijsticoon te klikken dat zich achter het veld bevindt. Dit ziet er uit als volgt: . Als men op dit icoon klikt verschijnt er een lijst in een apart venster. Vervolgens kan de gebruiker één van de termen uit de lijst aanklikken. Hierdoor zal deze automatisch in het juiste veld worden ingevuld.

Indien de gebruiker een term zou willen invoeren die nog niet in de lijst voorkomt, kan hij, indien dit voor het invoerveld in kwestie is toegestaan, een term aan de lijst toevoegen. Dit kan door in het invoervak bovenaan de lijst een term in te vullen en op de knop “Voeg toe” te klikken. Er wordt dan eerst gecontroleerd of de term al in de databank voorkomt. Indien dit niet het geval is, kan de gebruiker op bevestigen klikken om de term, de naam of de afbeelding aan de lijst toe te voegen. Als men de lijst opnieuw oproept, zal de nieuwe term, persoon, instelling of afbeelding in de lijst zijn opgenomen en kan men deze selecteren.

De volgende afbeelding illustreert een lijstschermb voor het trefwoord dier.



Welke inhoud in welk veld?

In elk veld dient een specifiek aspect van de objectbeschrijving te worden ingevoerd. Welke informatie er in een bepaald veld gevraagd wordt, is voor elk veld vastgelegd. De gebruiker kan dit te weten komen door op het info-icoon te klikken dat zich achter elk veld bevindt. Dit info-icoon ziet er uit als volgt: . Als men dit aanklikt zal er een nieuw venster worden geopend met daarin allerlei informatie over de gegevens die worden verwacht in een veld. Deze informatie bestaat doorgaans uit drie onderdelen:

1. Een beschrijving van de verwachte veldinhoud
2. Een voorbeeld
3. Een aantal instructies betreffende het invullen van het veld.

Het grootste deel van de beschrijvingen is afkomstig van het invulboek van het project MOVE van de Provincie Oost-Vlaanderen⁹.



Verplichte velden

Velden waarbij achter de veldnaam een rode * voorkomt, zijn verplicht in te vullen. Indien men deze niet invult, zal nadat op de opdracht "record opslaan" werd geklikt, een controle plaatsvinden, waarna zal aangegeven worden, welke velden nog moeten worden ingevuld.


⁹ Voor meer info over MOVE zie http://www.oost-vlaanderen.be/public/cultuur_vrijetijd/cultuur/musea/move/index.cfm


Herhaalbare velden

Sommige velden bevatten een soort van informatie die herhaalbaar is. Een voorbeeld daarvan is het titelveld. Een kunstwerk kan immers in zijn levensloop verschillende titels hebben gekregen. Het kan een officiële titel hebben, die werd gegeven bij de creatie, maar het kan later, bijkomend, een meer populaire titel hebben gekregen. Daarom is het soms nodig dat een veld herhaalbaar is. Dit wil zeggen dat het meerdere keren kan voorkomen.


De velden die herhaalbaar zijn, kan men herkennen aan het -icoon. Als men hierop klikt, zal er onder een bepaald veld een nieuw leeg veld verschijnen. Deze velden gedragen zich op dezelfde manier als niet-herhaalbare velden. Indien men een eerder toegevoegd veld zou willen verwijderen, kan dit door op het min-icoon te klikken .

Gegroepeerde velden

Een speciaal type van herhaalde velden zijn de gegroepeerde velden. Dit zijn velden die men als één groep kan beschouwen. Als een gebruiker bij één van deze velden op het -teken klikt, zal er ook voor de andere leden van de groep een veld worden bijgemaakt.

Omgekeerd zullen alle bijgecreëerde velden voor de hele groep verdwijnen als men op het min-teken achter een veld klikt .

De veldinhoud wissen

De inhoud van een bepaald veld kan ook in één keer worden gewist. Dit kan door op het wis-icoon te klikken .

De inhoud van een heel tabblad wissen

Men kan de inhoud van een volledig tabblad in één keer wissen door in de linkerkolom van het invoerscherm op “Tabblad wissen te klikken”.

De inhoud van een heel record wissen

De inhoud van een volledig record kan men in één keer wissen door in de linkerkolom van het invoerscherm op de link “Record wissen te klikken”.

Een record opslaan

Als een gebruiker de ingevoerde informatie wilt bewaren in de daarvoor voorziene invulvelden, kan deze in de linkerkolom van het invoerscherm klikken op de knop “record opslaan”.

3.4.3.12 Controle

Nadat een gebruiker de opdracht gegeven heeft om een ingevoerde objectbeschrijving op te slaan, worden eerst een aantal controles op de gegevens uitgevoerd. Indien

bepaalde velden niet ingevuld werden, of niet op de juiste manier, zal dit onder de velden waarvoor de controle negatief evalueerde worden vermeld. De onderstaande afbeelding is hier een voorbeeld van:

The screenshot shows a web application interface for 'Geschiedenis van Landbouw en Voeding'. The page title is 'www.HetVirtueleLand.be'. The navigation menu includes 'Home', 'Toeweging', 'Zoeken', 'Contact', 'Logop', 'Invalideertactie', and 'Aanpak'. The main content area is divided into two sections: 'Nieuw Controle' and 'Preview Record'. The 'Preview Record' section contains several input fields with associated error messages. The 'objectnummer' field is highlighted in red and contains the value '001'. Below it, a red error message reads: 'Dit veld is verplicht in te vullen. Gelieve ook hier een waarde voor in te vullen!'. Other fields like 'objectcategorie', 'titel', 'jaar verwerving', 'toestand', and 'objectnaam' also have red error messages. The left sidebar shows a 'Nieuw Controle' button and a 'Functie' menu with options '- Record opslaan' and '- Terug'.

Zo is het veld objectnummer een verplicht veld. Indien dit niet is ingevuld, zal er onder dit veld de boodschap "Dit veld is verplicht in te vullen. Gelieve ook hier een waarde voor in te vullen!" verschijnen.

Een voorbeeld van een veld waar de vorm van de ingevulde gegevens wordt gecontroleerd is het datumveld. Indien een datum niet de vorm jjjj-mm-dd (bijv. 1999-12-05 heeft, zal er een foutboodschap worden gegenereerd.

Indien een dergelijke boodschap verschijnt, dient men de richtlijn te volgen en de aanpassingen door te voeren. Vervolgens kan men opnieuw, in de linkerkolom, op "record opslaan" klikken. De controles zullen dan opnieuw worden uitgevoerd.

3.4.3.13 Bevestiging

Indien de controle succesvol verlopen is, zal er geen foutboodschap meer verschijnen en zal de opdracht "record opslaan" in de linkerkolom vervangen worden door "bevestigen". Men kan dan de inhoud van het record nalezen en indien de gebruiker akkoord is kan hij op bevestigen klikken. Pas dan zal de informatie die u invoerde in de databank worden vastgelegd.

3.4.3.14 Resultaat

Na de invoer krijgt de gebruiker de onderstaande boodschap te zien:



Indien de invoer succesvol was wordt dit gemeld. Indien dit niet het geval is, krijgt de gebruiker onder de titel "Boodschap" een foutmelding te zien in een rode kleur. Indien er een fout optreedt wordt dit automatisch aan de helpdesk gemeld en neemt deze contact op met de gebruiker.

3.5 Testfase

De toepassing werd uitgetest in samenwerking met twee heemmusea: Het Molenijzer uit Putte en Het Museum van de Zuiderkempem uit Oevel (Westerlo). In wat volgt zal beschreven worden hoe de testfase met beide musea verliep.

3.5.1 Heemmuseum Het Molenijzer Putte

Instellingsnaam: Het Molenijzer
Adres: Heuvelstraat 41 B
Postcode: 2580
Gemeenste: Putte
Contactpersoon: Staf De Winter (dewinter.staf@skynet.be)

Het heemmuseum van Putte stapte vanaf oktober mee in het project. Zij waren ook aanwezig tijdens de eerste vergadering en waren onmiddellijk bereid mee te werken. Eind december was er een eerste proefversie van de applicatie beschikbaar en kon er met de test begonnen worden.

Aanspreekpunt van het project voor Het Molenijzer was Staf de Winter. Midden januari werd met hem afgesproken om het verloop van het project te bekijken. Als voorbereiding van deze bijeenkomst werden een aantal handleidingen voorzien. Een eerste handleiding had betrekking op de technische werking van de toepassing. Een andere op de manier waarop namen, data enz. moesten geschreven worden. Deze laatste handleiding werd gebaseerd op het werk dat reeds door het MOVE project van de provincie Oost-Vlaanderen in dit verband werd gedaan. Ook werden een groot deel van de afbeeldingen die door het museum van de objecten werden gemaakt op voorhand ingevoerd in de databank. Ook werd aan het museum een exemplaar bezorgd van de publicatie van Jeanne Hogenboom over basisregistratie, samen met enkele CIDOC-fiches met praktische informatie in verband met collectieregistratie¹⁰.

Tijdens de eerste bijeenkomst werd de proefapplicatie aan de hand van de handleidingen uitvoerig besproken en gedemonstreerd. Tegelijkertijd werden enkele praktische afspraken gemaakt. Van Putte werd gevraagd om een honderdtal objecten die verband hielden met de geschiedenis van landbouw en voeding te fotograferen en te beschrijven in de databank. Er werd gevraagd om bij de test de nadruk te leggen op het testen van de applicatie in plaats van op het invoeren van zoveel mogelijk voorwerpen. Ook werden enkele praktische aspecten van de collectieregistratie besproken. Vervolgens werd gestart met de test.

Het heemmuseum stuitte onmiddellijk op enkele praktische problemen. Gezien de museumruimtes in de winterperiode slecht verwarmd werden, was het moeilijk om lange tijd in de ruimtes te verblijven, waardoor de registratie slechts langzaam vorderde. Een ander probleem was dat niet alle medewerkers van het museum over de nodige ICT-kennis beschikten om aan het project mee te werken, zodat het werk voornamelijk door

¹⁰ Hogenboom, Jeanne. *Basisregistratie voor collecties, voorwerpen en beeldmateriaal*. Rotterdam, 1988, 113 p. Het etiketteren en merken van objecten, CIDOC. Registratie stap voor stap: als een object het museum binnenkomt, CIDOC.

de heer De Winter werd uitgevoerd. Toch kwamen er waardevolle suggesties voor aanpassing:

- Zo werd gevraagd om blanco invul fiches te voorzien met daarop een afbeelding en het afbeeldingsnummer, om zo manueel de fiches te kunnen invullen;
- Ook werd er gevraagd om extra functionaliteit toe te voegen om afbeeldingen op nummer te kunnen opzoeken vanuit de objectbeschrijvingen;
- Er werd ook gevraagd om op sommige plaatsen een extra lijstknop te voorzien, zoals bij de trefwoorden;
- Ook werd gesuggereerd om het lock-systeem anders te implementeren omdat het uitloggen of vrijgeven van een object tijdens de bewerking soms werd vergeten;
- Verder werden ook suggesties gedaan om de opmaak van bepaalde velden anders te laten verlopen;
- Ook kwamen door het gebruik een aantal fouten aan het licht die bij eerdere tests niet waren opgevallen.

Nadat de invoer van de records gebeurd was, werden de records door het CAG nagelezen op spellingsfouten en werden er hier en daar suggesties gedaan voor aanpassing van de inhoud. Ook werd bij deze test nagegaan welke trefwoorden aan de objecten werden gekoppeld en werden relaties met andere trefwoorden gelegd.

Uiteindelijk werden door het museum een 100-tal objecten ingevoerd. De conclusie was dat het programma voor het museum bruikbaar was voor gebruik. Het museum heeft ook gemeld dat het in de toekomst objectbeschrijvingen wilt blijven toevoegen aan de databank.

3.5.2 Heemmuseum De Zuiderkempen Westerlo

Instellingsnaam: De Zuiderkempen
Adres: Sint-Michielsstraat 2
Postcode: 2260
Gemeenste: Westerlo
Contactpersoon: Remi Heylen (rheylen@provant.be)

Het verloop van de test bij het Museum van de Zuiderkempen verliep gelijkaardig. De heemkring opteerde er voor om de objecten van één kamer van het museum die verband houden met zuivel te registreren. Verantwoordelijken hier waren Remi Heylen en Guy Thys. De suggesties voor aanpassing kwamen min of meer overeen met deze van Putte.

Na het invoeren van een groot deel van de records werden ook hier een aantal verbeteringen aangebracht en een aantal suggesties voor de aanpassing gedaan. Er werden in totaal een vijftigtal objecten geregistreerd. De conclusie was dat het programma ook voor dit museum bruikbaar was. Ook het museum van de Zuiderkempen heeft te kennen gegeven dat het in de toekomst van de toepassing gebruik wenst te maken.

3.6 Publicatie en verspreiding van de resultaten

3.6.1 Publicatie geregistreerde collectieinformatie

De objectbeschrijvingen die door de twee musea werden ingevoerd zijn in een laatste fase gepubliceerd op de website www.hetvirtueleland.be. Dit gebeurde op twee verschillende manieren.

Eerst en vooral werden de records via het zoekstelsel ter beschikking gesteld. Een zoekopdracht op de naam van de instelling ('Het Molenijzer' of 'De Zuiderkempem') levert de records op die door deze instellingen werden ingevoerd.

Ten tweede werd er een link gelegd tussen een aantal van de ingevoerde objectbeschrijvingen en een verhaal over zuivelgeschiedenis op www.hetvirtueleland.be. Op deze website worden naast een zoekmechanisme ook verschillende verhalen gepresenteerd over de geschiedenis van landbouw en voeding. De onderwerpen die aan bod komen zijn zeer divers. Het kan zowel gaan om een geschiedenis van het koken in 19^{de} en 20^{ste} eeuw als om een overzicht van de evolutie van het Europese landbouwbeleid. Gezien de objectbeschrijvingen voor een groot deel betrekking hebben op materiaal dat ingezet werd bij de bereiding van melkproducten, zoals boter en kaas, werd er voor geopteerd om de objecten te koppelen aan het verhaal over de Belgische zuivel in de 19^{de} en 20^{ste} eeuw. Op die manier wordt een bezoeker van de website bij het lezen van het verhaal in contact met de fysieke objecten afkomstig uit beide musea. Daarnaast kan de bezoeker zowel extra informatie over de objecten als over de de instelling oproepen. Op deze manier wordt het object zowel in zijn historische als in zijn omgevingscontext gesitueerd.

De volgende afbeelding is een screenshot van het verhaal over de geschiedenis van de zuivel in 19^{de} en 20^{ste} eeuw. De foto van de melkmachine is afkomstig van het museum Het Molenijzer uit Putte.



3.6.2 Verspreiding resultaten onderzoek

De resultaten van het onderzoek werden op verschillende manieren verspreid.

Eerst en vooral werd er op 14 september 2005 een studiedag georganiseerd met als onderwerp het Agrarisch Erfgoed. Tijdens deze studiedag werden er projecten voorgesteld van het Centrum Agrarische Geschiedenis, het Interfacultair Centrum Agrarische Geschiedenis en van de Stichting Levend Erfgoed. Op deze studiedag kwamen de resultaten van het project dat in dit rapport wordt voorgesteld aan bod. De studiedag was succesvol. 56 mensen schreven zich in voor het voormiddagprogramma, waaronder een groot deel medewerkers van de musea die speciaal met dit project werden beoogd. Een lijst van deelnemers aan de studiedag vindt men in de bijlagen (zie bijlage 2). De reacties van de deelnemers op de studiedag waren positief. Enkele van de reacties die tijdens deze studiedag werden opgetekend zijn de volgende:

- De heer Jan Bieleman van de Universiteit van Wageningen (Nederland) was sterk geïnteresseerd om gebruik te maken van de invoerinterface voor de registratie van de collectie van hun landbouwmuseum.
- De heer Remy Heylen van het Museum van de Zuiderkempes merkte op dat de aanpak van het project, waarbij gebruik wordt gemaakt van de huidige mogelijkheden op vlak van ICT, de interesse voor het agrarisch erfgoed bij jonge mensen zou kunnen stimuleren.
- De heer Leopold Maes informeerde naar de mogelijkheden van de applicatie om ook bibliotheekinformatie te registreren en zag dit als een waardevolle aanvulling op de mogelijkheden ervan.
- De heer Jaak Geuns van de dienst Toerisme van de stad Peer informeerde naar de mogelijkheden om de collectieinformatie van het stedelijk museum van Peer in te voeren in de database van het CAG en dit via deze weg te publiceren op het internet.

Uit de studiedag bleek dat de aanwezige musea geïnteresseerd waren in het project en zeer benieuwd naar de volgende stappen die genomen zullen worden.

Ten tweede werd het rapport waarin verslag wordt uitgebracht van het project in elektronische versie gepubliceerd. Het is te downloaden via de website van het Centrum Agrarische Geschiedenis (www.cagnet.be onder de rubriek publicaties). Via deze weg wil het CAG het project verder bekend maken en toekomstige ontwikkelingen aan de doelgroep communiceren.

Ten derde wordt het project in één van de volgende digitale nieuwsbrieven van het CAG in de kijker geplaatst. Via deze nieuwsbrief wil het CAG om de twee maanden iedereen in Vlaanderen en daarbuiten, die geïnteresseerd is in het erfgoed landbouw en voeding, bereiken. Via deze weg kunnen een groot aantal mensen geïnformeerd worden over het bestaan en de mogelijkheden van dit project.

4 Conclusie en mogelijke verdere stappen

4.1 Conclusie

Dit project had tot doel een online registratie-interface uit te bouwen op basis van de technologie van ADLIB Information Systems. Hiermee wilde het Centrum Agrarische Geschiedenis enerzijds nagaan of de technologie van ADLIB hiervoor geschikt was. Anderzijds wilde het nagaan of een dergelijk instrument kon worden ingezet voor de registratie van de collecties van kleine, niet-professionele musea.

Het antwoord op beide vragen is positief. Hoewel er bij het uittesten van de ADLIB Internet Server in de beginfase een aantal problemen opdoken (zie bijlage 1), bleek dit product op het einde van het project naar behoren te functioneren. De problemen werden steeds gesignaleerd aan ADLIB en deze zorgde binnen relatief korte tijd voor een update van de software die het probleem oploste. Toch zijn er nog een aantal opmerkingen die in dit verband moeten worden gemaakt. De snelheid waarmee een schrijfpdracht op een ADLIB Database wordt uitgevoerd, is relatief traag. Hier zou de firma in de toekomst nog aandacht aan kunnen besteden. Een tweede aandachtspunt is de gelijktijdige bruikbaarheid van de Internet Server voor deze doeleinden. De huidige test werd op kleine schaal uitgevoerd, maar de vraag is hoe de server zich zou gedragen als honderd gebruikers tegelijkertijd informatie in eenzelfde databank zouden willen invoeren. In theorie zou dit geen problemen mogen opleveren, gezien de ADLIB Internet Server hier speciaal voor ontworpen is, maar dit zou nog verder in de praktijk moeten worden uitgetest.

Ten tweede blijkt uit het project dat een online interface voor kleine, niet-professionele musea bruikbaar is als instrument voor collectieregistratie. De musea waarmee werd samengewerkt beschouwden de applicatie als een nuttig instrument om hun collectie mee te registreren. De bruikbaarheid werd ook aangetoond door de objectbeschrijvingen die door hen werden ingevoerd. Ook andere musea dan de twee testmusea bleken enthousiast. Dit was merkbaar tijdens de studiedag die doorging op 14 september 2005. Tijdens deze studiedag, met als onderwerp het agrarisch erfgoed werd de applicatie voorgesteld. Op deze studiedag waren verschillende kleine musea aanwezig, die te kennen gaven geïnteresseerd te zijn in het project. Ook was er interesse van het landbouwmuseum, verbonden aan de Universiteit van Wageningen. Ook deze instelling, die zelf met ADLIB werkt, zag het nut en de voordelen van een dergelijke toepassing in.

Een andere vaststelling is dat het project en de gevolgde manier van werken een belangrijke stimulus was voor de collectieregistratie in de twee testmusea. Het project was voor hen een aanleiding om na te denken over collectieregistratie en over de praktische uitvoering ervan. Zij gaven alleszins te kennen om in de toekomst verder werk te maken van de registratie van hun eigen collectie.

4.2 Mogelijke verdere stappen

Op vlak van techniek lijkt het aangewezen om de ADLIB Internet Server verder op zijn bruikbaarheid te testen door de applicatie in te zetten voor de registratie van grotere collecties. Het CAG heeft de bedoeling om in de toekomst projecten op te zetten met kleine musea om hun collectie agrarisch erfgoed te registreren met de ontwikkelde

software. Uit het rapport van het project Veldwerk-Denkwerk¹¹ dat door het Centrum Agrarische Geschiedenis werd uitgevoerd, blijkt immers de noodzaak van registratie van landbouwcollecties omwille van verschillende redenen. Eerst en vooral zijn verschillende nog niet-geregistreerde collecties fysiek bedreigd, door gebrek aan conservering of door het overlijden van de eigenaar. De kans is immers reëel dat na een overlijden de stukken te koop worden gesteld en verdwijnen naar het buitenland of zelfs simpelweg worden vernietigd. Ten tweede verdwijnt langzaam maar zeker de generatie mensen die de functie van bepaalde objecten kennen omdat ze deze zelf gebruikt hebben of hebben zien gebruiken. Daardoor wordt het steeds moeilijker om objecten te beschrijven en om de context te achterhalen waarin ze werden gebruikt. Ten derde is collectieregistratie één van de basisfuncties van een museum, om een collectie te kunnen beheren en open te stellen voor het publiek.

Een tweede fase van dit project zou zich op de registratie van een aantal van de meest bedreigde collecties kunnen richten.

Ook kan de applicatie nog verder uitgewerkt worden. Een aantal aspecten van de applicatie, zoals de manier waarop record-locks worden aangebracht en verwijderd zou herbekeken moeten worden. Verder kan de functionaliteit ook worden uitgebreid. Bijvoorbeeld zou de mogelijkheid kunnen voorzien worden om bibliotheekgegevens van musea via een gelijkaardige interface te registreren.

¹¹ Woestenborghs, B., *Veldwerk Denkwerk. Agrarisch erfgoed: stand van zaken*. Centrum Agrarische Geschiedenis. 2005.

5 Bijlagen

5.1 *Bijlage 1: Problemen ADLIB Internet Server*

- De inhoud van gelinkte velden, die niet via de Internet Server werden aangemaakt konden niet worden aangepast (ADLIB ref. nr. 2023)
- De inhoud van gelinkte velden verwijderen lukte niet
- Bij het schrijven van records via de Internet Server werden geen locks aangebracht (ADLIB ref. nr. 2024)
- In de documentatie was het (un)lock protocol niet gedocumenteerd
- Locks op afbeeldingen werden niet verwijderd als er een afbeelding werd aangemaakt
- Als er vanuit een record twee gelijke referenties werden gelegd naar een afbeelding werd het afbeeldingsrecord gelockt.
- ISAPI-crashes en trage prestaties (ADLIB ref.nr.: 2004)

5.2 Bijlage 2: Deelnemers studiedag rond Agrarisch Erfgoed 14/09/2005

Naam	Voornaam	Werkgever/instelling/woonplaats
Aerts	Erik	K.U.Leuven, dept. Geschiedenis
Annaert	Philippe	Federale bibliotheek landbouw, Brussel
Bieleman	Jan	Universiteit Wageningen, leerstoel agr.geschiedenis
Cool	Ivan	Torhout
Coppein	Bart	K.U.Leuven, afd. Romeins recht & rechtsgeschiedenis
Coppieters	Guy	Algemeen Rijksarchief, Brussel
Decat	Herman	t Smidshof, Attenhoven/Heemkunde Vlaanderen
De Cuber	Richard	Tildonk
De Herdt	René	MIAT, Gent
Demeyer	Mark	Boerenbond, Leuven
Desair	Lutgart	Heemkundige Kring Peer
D'Haeyere	Frans	Levend landbouwmuseum, Noorderwijk
De Winter	Staf	Heemkring 't Molenijzer, Putte
Egberts	Henk	Landbouwmuseum Wageningen
Feusels	Luc	Plattelandscentrum Meetjesland, Sint-Laureins
Geuns	Jaak	Stad Peer, Dienst Toerisme
Gherardts	Femke	Bakkerijmuseum, Veurne
Goossens	Willy	Worfthoeve, Geel
Goovaerts	Robby	Gemeente Putte
Gulinck-Janssens	Greet	Witloofmuseum, Kampenhout
Gullentops	André	Heemkring 't Molenijzer, Putte
Henckens	Carmen	Provincie West-Vlaanderen
Hermans	Paul	Heemkring Ansfried, Westerlo
Hermans	Roeland	CAG, Leuven
Heylen	Remi	Heemkring Ansfried, Westerlo
Heyrman	Peter	KADOC, Leuven
Hindryckx	Katrien	Provincie West-Vlaanderen
Huynen	Raymond	Academie Streekgebonden Gastronomie
Ingelbrecht	Laurent	Landbouwschool Torhout
Jengember	Carlo	Poperinge
Maes	Daphné	Heemkunde Vlaanderen, 's-Gravenwezel
Maes	Leopold	vzw Midzeelhoeve, Sint-Katelijne-Waver
Mertens	Louis	Heemkring Ansfried, Westerlo
Niesten	Eddie	CAG, Leuven
Otte	Els	Provinciaal Molencentrum, Wachtebeke
Peeters	Luc	Mechelse Veilingen, Sint-Katelijne-Waver
Schoefs	Hilde	Vlaams Centrum voor Volkscultuur, Brussel
Schrooten	Leo	Leuven

Segers	Yves	CAG/ICAG, Leuven
Standaert	Ovide	Korbeek-Dijle
Steen	Iris	Provincie Vlaams-Brabant, dienst Cultuur
Toelen	Toon	CAG, Leuven
Symons	Koen	VILT
Tijskens	Renaat	Herent
Trio	Patrick	Torhout
Vandenborne	Anne	KVLV, Wijgmaal
Van den Broeck	Hugo	Muizen
Van de Voorde	Erik	vzw Midzeelhoeve, Sint-Katelijne-Waver
Vandewynckel	Geert	VVV-Heuvelland
Van Hoof	Willy	vzw Midzeelhoeve, Sint-Katelijne-Waver
Van Liefferinge	Jules	Vlaamse Gemeenschap, afd. Land en tuinbouw
Van Molle	Leen	K.U.Leuven, dept. Geschiedenis
Vansteenkiste	Johan	Provinciaal Museum Bulskampveld, Beernem
Verdurmen	Elke	VCM-Contactforum voor Erfgoedverenigingen
Wauters	Ewald	Resource Analysis NV, Antwerpen
Zoons	Cindy	Vlaamse Gemeenschap, afd. Beeldende kunst en musea