

---

# Volledig gebruik van CSS op europe.fasson.com

Afstudeerverslag

---

## **Student**

Dennis van den Hoek  
99004676

## **Datum**

08-01-04

## **Opleiding**

Studie: Informatica en Informatiekunde  
Richting: VIA (Vormgeving en Ontwerp van Interactie)  
Locatie: Den Haag

## **Begeleiders**

Examinator: Dhr. P.J.G. Deters  
Examinator: J.P. van Leeuwen  
Bedrijfsmentor: Dhr. F. Steeman

## **Bedrijf**

Avery Dennison/RME – Afdeling ASD

## Referaat

In dit verslag wordt de afstudeeropdracht *Volledig gebruik van CSS op europe.fasson.com* besproken. De opdracht is uitgevoerd door Dennis van den Hoek, student aan de Haagse Hogeschool, voor de opleiding Vormgeving en Ontwerp van Interactie (VIA). Het project is uitgevoerd in opdracht van Avery Dennison, divisie Roll Materials Europe (RME). De activiteiten vonden plaats op het hoofdkantoor van RME, gevestigd te Leiden, op de afdeling IT Application Support and Development (ASD).

Descriptoren:

- Cascading Style Sheets
- HTML
- Lotus Notes
- Formula Language
- Lotus Script
- Agent
- IAD

## **Voorwoord**

Dit verslag heb ik geschreven in kader van mijn afstuderen voor de opleiding VIA aan de Haagse Hogeschool.

Dit project is de uitvoering geweest van de tweede opdrachtschrijving van dezelfde opdrachtgever, die ingeleverd is ter goedkeuring voor het afstudeertraject. Nadat een eerdere opdracht niet goedgekeurd werd, is deze opdracht wel aangenomen.

In dit eindverslag zal mijn afstudeerstage bij Avery Dennison besproken worden. Hierbij zal het gehele periode van 19 weken aan bod komen. De opdracht bestond uit de ontwikkeling van Cascading Style Sheets voor de website europe.fasson.com. Hierbij is ook een management systeem ontwikkeld, waarmee de nieuwe style sheets onderhouden kunnen worden. Dit verslag bespreekt het proces, dat hierbij doorlopen is.

Ik wil graag een woord van dank uiten richting mijn collega's van de afdeling ASD. Speciale dank gaat uit naar Adriaan Verkerk, Frank Steeman, Mee Lian Lee, Edward van Gelderen en Willem Issenger, voor hun begeleiding en technische ondersteuning. Ook wil ik graag mijn beide examinatoren bedanken, Dhr. P.J.G. Deters en Dhr. J.P. van Leeuwen, voor de begeleiding vanuit de opleiding en de feedback bij het schrijven van dit eindverslag.

Leiden, 7 januari 2004

Dennis van den Hoek

## Inhoudsopgave

<b>1 Inleiding.....</b>	<b>5</b>
<b>2 Organisatie en opdracht.....</b>	<b>6</b>
2.1 Avery Dennison .....	6
2.2 Volledig gebruik van CSS op de website.....	7
2.3 Verklaring van de methoden en technieken .....	11
2.4 Documenteren.....	13
<b>3 Definitiefase.....</b>	<b>14</b>
3.1 Inleiding .....	14
3.2 Opstellen van het Plan van Aanpak .....	15
3.3 Zelfstudie .....	21
3.3.1 Cascading Style Sheets .....	21
3.3.2 Lotus Notes .....	23
3.4 Uitvoeren van de systeemanalyse .....	24
3.4.1 Analyse van de website.....	27
3.4.2 Ontwikkelen van de document trees .....	31
3.4.3 Analyse van het management systeem voor het menu.....	35
3.5 Opstellen van de systeemeisen .....	36
3.5.1 Doelgroepen .....	36
3.5.2 Style sheets.....	37
3.5.3 Management systeem .....	39
3.6 Opstellen van het systeemconcept .....	42
3.6.1 Style sheets.....	42
3.6.2 Style sheets voor het printen .....	46
3.6.3 Management systeem .....	47
3.7 Opstellen van het pilotplan.....	51
3.7.1 Verdeling van pilots .....	51
3.7.2 Pilots en prioriteiten.....	53
<b>4. Ontwikkelfase .....</b>	<b>55</b>
4.1 Inleiding .....	55
4.2 Eerste style sheets.....	56
4.3 Tweede style sheets .....	63
4.4 Het management systeem .....	65
4.4.1 Technische functionaliteit .....	66
4.4.2 User interface .....	70
<b>5. Invoeringsfase .....</b>	<b>73</b>
5.1 Style sheets voor scherm en printen .....	73
5.2 Management systeem.....	73
<b>6. Evaluatie .....</b>	<b>74</b>
6.1 Procesevaluatie .....	74
6.2 Productevaluatie .....	78
<b>Literatuurlijst.....</b>	<b>80</b>
<b>Interne bijlage - Bijlage 1 - Theorie Lotus Notes.....</b>	<b>81</b>
<b><u>Externe bijlagen</u></b>	
<b>Bijlage A - Definitiestudie</b>	
<b>Bijlage B - Aanbevelingen en Script</b>	

## 1. Inleiding

In dit verslag wordt het procesgedeelte van mijn afstudeertraject bij Avery Dennison besproken. Het verslag behandelt het proces, waarbij style sheets voor een website zijn ontwikkeld, alsmede een management systeem waarmee deze style sheets onderhouden kunnen worden. Aan de hand van dit verslag zal ik beoordeeld worden voor het gevolgde traject.

Dit document is niet bedoeld als naslagwerk voor het stagebedrijf Avery Dennison. Er wordt echter wel een exemplaar aan de opdrachtgever aangeboden.

Allereerst zal in hoofdstuk 2 het stagebedrijf en de opdracht besproken worden. Hierbij zal een korte achtergrond van Avery Dennison en RME gegeven worden. Daarnaast zal de opdracht besproken worden, aan de hand van delen uit de originele opdrachtschrijving. Ook zal er een verklaring gegeven worden van de gebruikte methoden en technieken en wordt de wijze van documenteren voor dit project besproken.

In hoofdstuk 3 wordt de definitiefase van het project besproken. Hierbij zal ingegaan worden op het proces waarbij de definitiestudie tot stand gekomen is. Belangrijk hierbij was de systeemanalyse, die aan het begin van de definitiefase heeft plaatsgevonden. Ook zal de zelfstudie, die tijdens dit project gedaan is, nader besproken worden. Verder wordt hierin besproken, hoe de rest van de definitiestudie tot stand gekomen is. Daarbij komen zaken aan bod zoals het opstellen van het Plan van Aanpak, de systeemeisen en het systeemconcept en de invulling van het pilotplan.

Hoofdstuk 4 behandelt de ontwikkelfase van het project. Hierin zal besproken worden hoe de ontwikkeling van de style sheets voor de website verlopen is. Er zal ook aandacht besteed worden aan de ontwikkeling van het management systeem, dat bedoeld is voor het onderhouden van de nieuwe style sheets.

In hoofdstuk 5 zal een beschrijving gegeven worden van de invoering van de verschillende ontwikkelde producten.

Ten slotte wordt het laatste hoofdstuk gewijd aan een evaluatie van het project. In de vorm van een proces- en productevaluatie zal teruggeblikt worden op de belangrijkste onderdelen van het traject.

## 2 Organisatie en opdracht

### 2.1 Avery Dennison

#### Over Avery Dennison

Avery Dennison behoort tot de 500 grootste Amerikaanse bedrijven (*Fortune 500 company*) en beschikt over 200 vestigingen verspreid over 39 landen wereldwijd. Avery Dennison is wereldwijd leider op het gebied van drukgevoelige technologieën en zelfklevende materialen, kantoorbenodigdheden, etiketteersystemen en speciale chemische producten, met consumenten- en industriële markten overal ter wereld.

De stichter van het bedrijf, R. Stanton Avery, vond het originele concept van het zelfklevende etiket in 1935 uit – dit heeft geleid een globale industrie, die vele toepassingsgebieden heeft.

#### Over RME/ ASD

De activiteiten vinden plaats bij de divisie Roll Materials Europe (RME), onderdeel van de Roll Materials World Wide Division. De stage wordt volbracht op het hoofdkantoor van deze divisie, gevestigd in Leiden en de afstudeerder is werkzaam op de afdeling IT Application Support and Development. Binnen de divisie RME is de afdeling ASD verantwoordelijk voor onder andere het onderhoud van de website en de applicaties die gebruikt worden door de afdeling e-Business. Verder is deze afdeling verantwoordelijk voor de ontwikkeling van nieuwe applicaties, en wordt ondersteuning verleend bij het gebruik van de verschillende systemen die nodig zijn binnen het bedrijfsproces.

De divisie RME heeft zich toegelegd op de productie van het merk Fasson; wereldwijd marktleider op het gebied van zelfklevende labels.

- Op het hoofdkantoor zijn 140 mensen werkzaam.
- De totale IT-afdeling bestaat uit 50 mensen, waarvan er 40 op het hoofdkantoor werkzaam zijn.
- De afdeling IT Application Support and Development bestaat uit 20 mensen waarvan de meeste ook op het hoofdkantoor werkzaam zijn.
- Er wordt ontwikkeld met behulp van o.a. Lotus Notes en Visual Basic. Het ontwikkelteam bestaat uit 10 mensen.
- De meeste IT-toepassingen zijn gecentraliseerd en worden onderhouden door de 40 IT-specialisten in Leiden.

## 2.2 Volledig gebruik van CSS op de website

### Probleemstelling

De RME website (europe.fasson.com) is ontwikkeld in een Lotus Notes Domino omgeving. Dit is een gelimiteerde omgeving, als gevolg van het feit, dat Lotus Notes niet specifiek gericht is op het ontwikkelen van websites. De website is complex opgebouwd. Zo wordt de HTML van de pagina's voor een groot deel door software gegenereerd. De codes komen uit verschillende Notes databases met als gevolg, dat uiterlijke kenmerken op verschillende manieren gedefinieerd zijn; er wordt gebruik gemaakt van een style sheet, maar veelal worden stijlen in stukken HTML in de verschillende delen van de software geprogrammeerd. Dit gebeurt door verschillende ontwikkelaars, waarbij geen gebruik wordt gemaakt van standaarden of richtlijnen voor de stijlen. Deze inconsistente implementatie van verschillende oplossingen geeft grote problemen bij het onderhouden van de gebruikte stijlen op de website. Het is moeilijk aanpassingen te maken en in sommige gevallen is dit niet mogelijk.

Verder is het zo, dat er in de oude situatie geen printbare versies van de pagina's op de site aanwezig zijn, iets dat ook gewenst wordt.

Men heeft het belang van Cascading Style Sheets, in het vervolg CSS genoemd, als oplossing voor de verschillende problemen met betrekking tot de uiterlijke kenmerken ingezien. De implementatie van CSS is echter eveneens ongeorganiseerd en gebeurt meer ad-hoc, zonder onderliggende structuur, waardoor de genoemde problemen zijn blijven bestaan.

Bij de start van het project is er een in Lotus Notes ontwikkelde omgeving voor het bijhouden van de content op de website, zoals teksten en afbeeldingen; een content management system (CMS). Er is ook een soortgelijke omgeving ontwikkeld om het menu te onderhouden. Er is nog geen generieke manier om vormgevingsaspecten te onderhouden. Het is gewenst, dat de style sheets gemakkelijk en snel kunnen worden aangepast, met behulp van een vergelijkbaar management systeem.

### Doelstelling

Doelstelling van de opdracht is het opnieuw opzetten van de website, zodat deze optimaal gebruik maakt van CSS, waarbij in eerste instantie de huidige lay-out zoveel mogelijk gehandhaafd blijft. Hiervoor zal een style sheet ontwikkeld worden, waarin vastgelegd is hoe de verschillende HTML-elementen gepresenteerd dienen te worden op het scherm. Als oplossing voor de printbare pagina's zal een style sheet ontwikkeld worden waarin vastgelegd is hoe deze elementen gepresenteerd moeten worden op papier.

Er zal een systeemanalyse gedaan worden omtrent de huidige functionaliteit en structuur. Ook zal er een adviesrapport opgesteld worden, gericht op het toepassen van *best practices*<sup>1</sup> voor CSS. Uitgaande van dit adviesrapport zal een tweede versie van de style sheets ontwikkeld worden, die de *best practices* realiseren voor de website.

De tweede versie zal voor beide style sheets ontwikkeld worden; een style sheet voor de presentatie op het scherm en een style sheet voor de presentatie op papier.

Om te bereiken, dat de stijlen op de site gemakkelijk onderhouden kunnen worden, zal een management systeem voor de style sheets ontwikkeld worden, zodat aanpassingen in de style sheets gemakkelijk en snel kunnen worden doorgevoerd. Dit systeem zal gebruik maken van een eigen database, welke ook tijdens dit project ontwikkeld zal worden.

Hiervoor zullen style sheets volledig opnieuw ontwikkeld worden in CSS2, versie 2 van CSS. Hiervoor zal de HTML-code van de betreffende pagina's ook gewijzigd moeten worden. Dit zal gebeuren voor een aantal pagina's, waarvan vastgelegd is, dat deze gebruik maken van een standaard lay-outstructuur. Alle nodige wijzigingen zullen gedocumenteerd worden. Deze documentatie zal gebruikt worden bij de implementatie van de overige pagina's van de website. Deze implementatie zal binnen een ander project plaatsvinden. Ook zal het management systeem voor de style sheet en de database volledig binnen dit project ontwikkeld worden. Dit systeem zal wel in z'n geheel tijdens dit project geïmplementeerd worden. De ontwikkeling van het management systeem zal in Lotus Notes uitgevoerd worden.

De website dient compatible te zijn met de volgende browsers, op zowel het Windows als het MacOS platform;

- Internet Explorer 5
- Internet Explorer 6
- Netscape 7

---

<sup>1</sup> Best practices is een veel gebruikte term om technieken en methoden e.d. aan te duiden, waarvan uit praktijkervaring vaak genoeg gebleken is, dat deze goed werken. Het zijn in veel gevallen een soort algemeen geaccepteerde standaarden geworden.



De opdracht bestaat uit de volgende onderdelen;

- Systeemanalyse van de site in de huidige vorm. Samen met het ontwerp van de lay-outstructuur en bevindingen uit de definitiestudie zal dit resulteren in een adviesrapport voor het e-Business team met betrekking tot nodige aanpassingen op het gebied van lay-out en CSS-codering.
- De ontwikkeling van een heldere *document tree* van de huidige website, geldig voor alle pagina's, zowel de publieke als beveiligde transactiepagina's. (*Document trees* zullen verder toegelicht worden in hoofdstuk 3.4 *Uitvoeren van de systeemanalyse*)
- Het ontwerp van een lay-outstructuur in CSS2, die zoveel mogelijk identiek is aan de huidige lay-out.
- Het ontwerp van een nieuwe, verbeterde lay-outstructuur.
- De ontwikkeling van een style sheet die het mogelijk maakt deze lay-out te gebruiken voor alle eerder genoemde browser/OS combinaties.
- Het ontwerp en ontwikkeling van het management systeem met bijbehorende database, t.b.v. onderhoud van de style sheets.
- De ontwikkeling van een generieke manier om de nieuwe style sheets en structuur met bijbehorend management systeem te implementeren.
- Er zijn op dit moment geen printbare versies van de pagina's. Er dient een nieuwe printlay-out ontworpen te worden.
- De ontwikkeling van een tweede style sheet voor het printen van de pagina's in print lay-out.
- Implementatie van de aangepaste HTML-pagina's en de style sheets.
- Implementatie van het management systeem voor de style sheets.
- Opstellen documentatie t.b.v. implementatie overige pagina's; het gaat hier om de pagina's die niet tijdens dit project gewijzigd worden. Deze pagina's zullen binnen een ander project aangepast en geïmplementeerd worden.

### *Uitgangssituatie*

#### *-documentatie*

Er is nooit eerder onderzoek gedaan naar de eerder genoemde probleemstelling. Bij aanvang is er dan ook geen documentatie met betrekking tot dit onderwerp die bij de opdracht van dienst kan zijn.

Documentatie met betrekking tot de huidige programmering van de website e.d. kan wel gebruikt worden.

#### *-software*

Zoals naar voren is gekomen bij de bespreking van de probleemstelling in hoofdstuk 2.2, wordt er bij aanvang van het project al gebruik gemaakt van een CMS en een management systeem voor het onderhoud van het menu op de website. Deze systemen zullen samen met de huidige style sheet en de programmering van de website bij de opdracht gebruikt worden.

Verder is de volgende specifieke software beschikbaar;

- Lotus Notes 5
- Internet Explorer 5
- Internet Explorer 6
- Netscape 4.7
- Netscape 7.1
- AceHTML Pro 5
- Smart Draw Professional 6
- Paint Shop Pro 4

## 2.3 Verklaring van de methoden en technieken

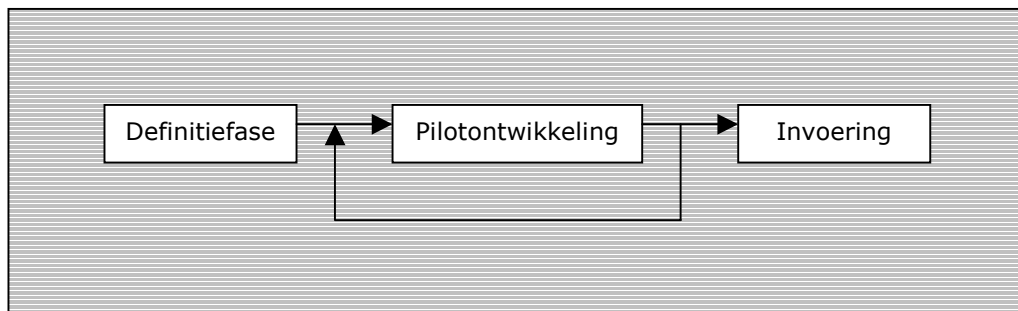
### Ontwikkelmethode IAD

Voor de opdracht wordt de ontwikkelmethode IAD (Iterative Application Development) gebruikt, toegepast in de vorm van RAD.

De IAD-ontwikkelmethodiek kent de volgende fases;

1. Definitiestudie – in deze fase wordt het nieuwe systeem in kaart gebracht. Mogelijkheden en beperkingen worden vastgelegd. Aan het einde van deze fase wordt het nieuwe systeem opgedeeld in *pilots*<sup>2</sup>.
2. Pilotontwikkeling – in deze fase worden de pilots ontwikkeld.
3. Invoering – het nieuwe systeem wordt ingevoerd.

Bij elk van deze fases kunnen iteraties plaatsvinden, de betreffende fase wordt dan dus meerdere keren doorlopen. IAD kent verschillende varianten; elke variant heeft een andere indeling van iteraties. Voor dit project is de variant *Rapid Application Development* (RAD) gebruikt. Hierbij wordt alleen de fase pilotontwikkeling geïtereerd.



---

<sup>2</sup> Pilot: Een deel van het uiteindelijke systeem, dat onafhankelijk van andere delen ontwikkeld en ingevoerd kan worden.

De volgende punten zijn de belangrijkste redenen geweest voor de keuze van RAD;

- De duidelijke probleemstelling en doelstelling van de opdracht hebben als gevolg, dat een eenmalige definitiestudie voldoende is voor het hele project.
- Wanneer de fase van implementatie meerdere malen uitgevoerd zou worden, zou dit betekenen, dat er *delen* van het systeem ingevoerd worden. Het heeft voor de opdrachtgever geen nut om een *deel* van het systeem ingevoerd te hebben, omdat op deze manier de eerder genoemde problemen uit de probleemstelling alsnog blijven bestaan. Op die manier zullen stijlkenmerken namelijk nog steeds op verschillende manieren gedefinieerd zijn. Ook de fase invoering zal dus maar eenmalig uitgevoerd worden. Hierbij zullen het management systeem en de style sheets met bijbehorende aangepaste HTML in één ingevoerd worden.
- Gekeken naar de uit te voeren activiteiten vergeleken met de beschikbare tijd is het wenselijk een minimum aan documentatie op te leveren. Alleen de noodzakelijke documenten zullen gemaakt worden. RAD is de IAD-variant met de minste documentatie.

### Ontwikkelomgeving

De ontwikkelomgeving van het hele project is Lotus Notes Domino. Het operating system waarop gewerkt zal worden is Microsoft Windows 2000 professional. Verder zal gebruikt gemaakt worden van Lotus Domino servers voor het ontwikkelen en testen van de nieuwe style sheets en het management systeem.

Verder wordt er gebruik gemaakt van de volgende technieken;

- HTML
- JavaScript
- CSS (CSS2.0)
- Lotus Script
- Lotus Notes Formula Language

## 2.4 Documenteren

Voor dit project wordt de gebruikelijke documentatie opgesteld. Er wordt een Plan van Aanpak geschreven en tijdens de definitie fase wordt een Definitiestudie geschreven. Ook wordt er een uitgebreide documentatie bijgehouden met betrekking tot de systeemanalyse, evenals systeemdokumentatie tijdens de ontwikkelfase van het traject. Tijdens de invoering wordt eveneens de nodige documentatie opgesteld.

Deze documentatie gebruik ik vrijwel allemaal zelf tijdens dit project, voor zaken als projectbeheersing en naslagwerk. Delen van deze documentatie worden echter ook aan de opdrachtgever opgeleverd. Als gevolg van het feit, dat Engels de voertaal is binnen dit bedrijf, moet documentatie in het Engels opgeleverd worden. Met de bedrijfsmentor zal per document steeds overlegd welke delen in het Engels opgeleverd moeten worden. Deze delen worden dan vertaald en in een daarvoor beschikbaar gestelde database opgeslagen.

### **3 Definitiefase**

#### *3.1 Inleiding*

Tijdens de definitiefase wordt de beginsituatie van het project in kaart gebracht. In de documentatie van de definitiestudie wordt deze situatie beschreven. Ook worden de eisen en wensen voor de nieuwe situatie in kaart gebracht in de vorm van een systeemconcept voor de te ontwikkelen CSS-documenten en het management systeem. Dit systeemconcept wordt opgedeeld in pilots. Na het voltooien van de definitiestudie kan begonnen worden met het ontwikkelen van de pilots.

Dit hoofdstuk geeft een beschrijving van de belangrijkste processen en keuzes die tijdens deze fase zijn gemaakt.

### 3.2 Opstellen van het Plan van Aanpak

Het Plan van Aanpak is het eerste document dat opgesteld wordt in kader van het project, en is hiermee de eerste documentatie die opgenomen wordt in de definitiestudie. Het Plan van Aanpak fungeert als basisplan voor het gehele ontwikkeltraject. Er wordt een korte achtergrond en probleemstelling gegeven, waarin de huidige situatie beschreven staat. Verder beschrijft het plan de doelstellingen van het traject met bijbehorende eindresultaten, op te leveren producten en randvoorwaarden.

Omdat het Plan van Aanpak als basisplan voor het ontwikkeltraject gebruikt wordt, wordt de eerste versie van het Plan van Aanpak in de eerste week voor het grootste deel afgerond. Het gaat hier om de *eerste versie*, aangezien de planning op dat moment nog aangepast kan worden. De meeste onderdelen van het Plan van Aanpak, liggen bij aanvang van het project reeds vast. Deze onderdelen zijn namelijk ook in de opdrachtomschrijving besproken. Het gaat hierbij om de volgende onderdelen;

- Achtergrondinformatie over het bedrijf
- Probleemstelling
- Doelstelling
- Uitgangssituatie
- Op te leveren producten
- Randvoorwaarden
- Te gebruiken methoden en technieken

Overige onderdelen kunnen na gesprekken met de bedrijfsmentor opgesteld worden. Tijdens deze gesprekken wordt besproken wat er tot dan toe op papier staat en de belangrijkste conclusie is, dat er geen maatregelen vastgelegd zijn met betrekking tot de bewaking van de voortgang van het project. Daarom zijn er nog twee hoofdstukken aan het Plan van Aanpak toegevoegd; *Beheersaspecten* en *Risicofactoren*. De redenen hiervoor zijn als volgt;

- Er moet duidelijk vastgelegd worden hoe ervoor gezorgd wordt, dat het project tot een goed einde gebracht wordt; dit wordt vastgelegd in het hoofdstuk *Beheersaspecten*.
- Er moet vooraf duidelijk afgesproken zijn wat de te nemen acties zullen zijn, wanneer blijkt, dat bepaalde onderdelen van het project niet volgens de planning uitgevoerd kunnen worden; dit wordt vastgelegd in het hoofdstuk *Risicofactoren*.

In het hoofdstuk *Beheersaspecten* worden de zogenaamde GOTIK-factoren behandeld. Dit zijn Geld, Organisatie, Tijd, Informatie en Kwaliteit. In het Plan van Aanpak heb ik naar eigen inzicht voor elk van deze factoren vastgelegd hoe deze te controleren zijn. Voor 'tijd' wordt bijvoorbeeld vastgelegd, hoeveel tijd er beschikbaar is en welke project documentatie gebruikt kan worden om het project te kunnen beheersen in het kader van tijd. Nadat dit document is opgesteld, wordt het door de bedrijfsmentor goedgekeurd.

In het hoofdstuk *Risicofactoren* wordt uiteengelegd wat de grootste 'valkuilen' zijn binnen dit project. Ook wordt beschreven wat de te nemen acties zullen zijn, wanneer blijkt, dat een bepaald onderdeel van het project niet volgens de normale planning uitgevoerd kan worden. De risico's worden in een gesprek met de bedrijfsmentor geïnventariseerd. Tijdens dit gesprek is de belangrijkste conclusie, dat de grootste risico's schuilen in de complexiteit van de website en de onderdelen van het project waarvoor technieken gebruikt gaan worden waarmee ik nog geen eerdere ervaring heb. Op basis van deze conclusie worden de risico's vastgelegd en met de bedrijfsmentor wordt besproken wat hierbij eventueel de te nemen acties zullen zijn. Het gaat hier om de volgende risico's;

- De ontwikkeling van het management systeem zou te zwaar kunnen zijn. Door het gebruik van technieken nieuw voor de stagiair, zou kunnen blijken dat de ontwikkeling van dit systeem zodanig ingewikkeld wordt, dat dit te veel tijd zal gaan kosten waardoor de voortgang van het project in het gedrang zou kunnen komen.
- Voor de ontwikkeling van de CSS moet er rekening gehouden worden met de complexiteit van de gehele website in de huidige vorm. De systeemanalyse heeft als primair doel duidelijke document trees op te stellen. Deze document trees samen beschrijven in principe de gehele lay-outstructuur van de website. Met het oog op de complexiteit van de website zou kunnen blijken dat het opstellen van algemeen geldende document trees veel tijd gaat kosten, wat ook weer de voortgang van het project zou kunnen storen.

Wanneer een van bovengenoemde punten voorkomen tijdens het project, zal in overleg met de bedrijfsmentor en de projectbegeleiders bepaald worden wat de verdere invulling van het project zal gaan worden. Er zullen dan (onderdelen van) pilots komen te vervallen. Onderdelen met een lage prioriteit zullen dan niet binnen dit project uitgevoerd worden. De opdracht zal dan dus zodanig herzien worden, zodat het project alsnog binnen de gestelde tijd afgerond kan worden.

Op eigen initiatief wordt ook nog het volgende risico voorgesteld, een valkuil waarmee eigenlijk in elk project rekening mee gehouden moet worden;

- Een laatste 'valkuil' waar rekening mee gehouden dient te worden is het verschijnsel dat ook wel bekend is als *scope creep*. In dit geval zullen er steeds meer wijzigingen gewenst worden in de pagina's van de website, die steeds verder van de opdracht afwijken.



Om deze laatste valkuil te voorkomen zal ik duidelijke grenzen moeten stellen en mij moeten houden aan de afgesproken systeemeisen. Op deze manier moet bereikt worden, dat alle uit te voeren activiteiten met het oog op de doelstelling van het project uitgevoerd worden. De bedrijfsmentor heeft hierbij ook een controlerende rol.

Nadat de bedrijfsmentor ook dit document heeft goedgekeurd, kon het hoofdstuk aan het Plan van Aanpak worden toegevoegd.

Een van de belangrijkste onderdelen van het Plan van Aanpak is de planning van het project. Deze wordt dus vroeg in het traject opgesteld en geeft een globaal overzicht van de uit te voeren activiteiten per week. De planning wordt zelfstandig opgesteld en met de bedrijfsmentor een aantal keren bijgesteld. In eerste instantie was het namelijk de bedoeling, dat de code van alle pagina's van de website door mij tijdens dit project aangepast zouden worden aan de nieuwe situatie. Met andere woorden; het was mijn taak om de gehele website geschikt te maken voor de style sheets die ook tijdens dit project ontwikkeld worden. In het begin van het project was ik mij nog niet bewust van de omvang en complexiteit van de website. Tijdens een informeel gesprek, dat ik de eerste week heb gehad met een van de ontwikkelaars die ook aan de website heeft gewerkt werd duidelijk, dat het wijzigen van alle HTML zodanig veel werk is, dat deze activiteit alleen al als project op zich gezien kan worden. Dit zou zoveel tijd gaan kosten, dat de voortgang van de rest van het project in gevaar zou komen.

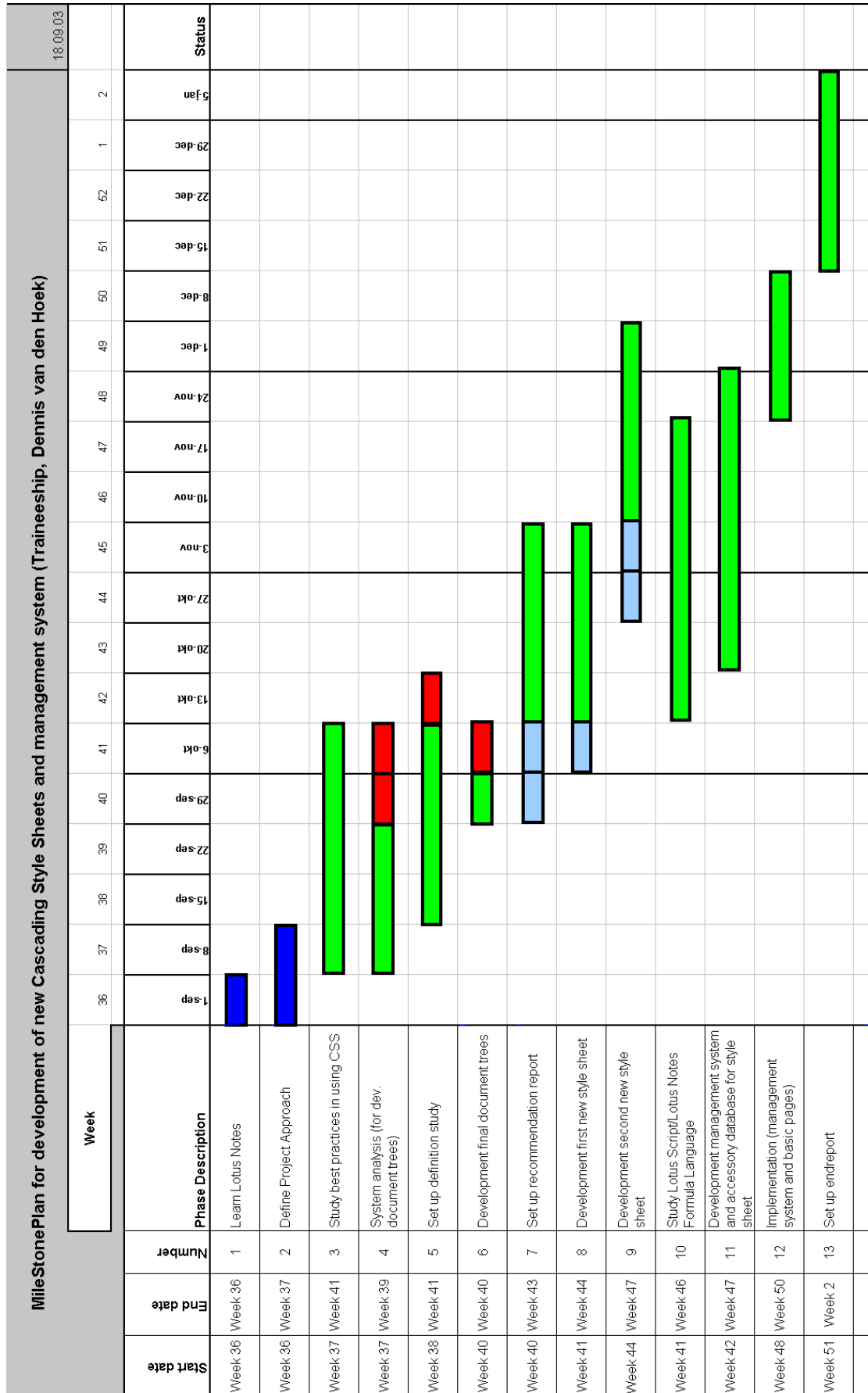
Dit probleem is besproken met de bedrijfsmentor en we hebben besloten, dat ik binnen dit project niet alle pagina's aanpas, maar een aantal pagina's, waarvan tijdens de systeemanalyse vastgelegd wordt, dat deze pagina's gebruik maken van een standaard lay-out. (bij het bespreken van de systeemanalyse zal dit verder toegelicht worden) De wijzigingen die nodig zijn voor deze pagina's, zullen door mij tijdens dit project uitgevoerd en gedocumenteerd worden. Nadat alle wijzigingen uitgevoerd zijn voor deze pagina's, zal op basis van die documentatie een script geschreven worden, waarin beschreven staat wat de te nemen stappen zijn bij het wijzigen van de HTML in de Notes databases. Omdat het hier standaard lay-outstructuren betreft, kan dit script gebruikt worden binnen een ander (stage)project, waarin de wijzigingen voor alle overige pagina's alsnog uitgevoerd zullen worden.

Als gevolg van bovenstaande beslissing is de planning aan het begin van de tweede week aangepast; het wijzigen van de HTML is opnieuw ingepland, omdat de wijzigingen nu niet betrekking hebben op alle pagina's, maar slechts een deel van de pagina's. Ook werd er een taak toegevoegd; het schrijven van het script voor het uitvoeren van de wijzigingen in de HTML.

Hieronder volgt een fragment van de planning zoals deze is opgenomen in het Plan van Aanpak.

Week	Activiteiten	Fase (IAD)
36	-Opstellen Plan van Aanpak (IAD documentatie) -Introducties -Leer Lotus Notes	Definitiefase
37	-Afronden Plan van Aanpak -Introducties -Systeem analyse (t.b.v. de ontwikkeling van document trees en ontwerp van de nieuwe lay-outstructuren) -Zelfstudie "best practices in using CSS2.0"	Definitiefase
38	-Systeem analyse (t.b.v. de ontwikkeling van document trees en ontwerp van de nieuwe lay-outstructuren) -Opstellen definitiestudie (IAD documentatie) -Zelfstudie "best practices in using CSS2.0"	Definitiefase
39	-Systeem analyse (t.b.v. de ontwikkeling van document trees en ontwerp van de nieuwe lay-outstructuren) -Opstellen definitiestudie (IAD documentatie) -Zelfstudie "best practices in using CSS2.0"	Definitiefase

In deze planning zijn alle belangrijke activiteiten per week opgenomen. Dit overzicht geeft echter geen helder beeld van de verdeling van de activiteiten over de tijd en ook is moeilijk te zien welke activiteiten *na elkaar* of juist *parallel* uitgevoerd worden. Om dit toch te visualiseren heb ik de planning ook in de vorm van een Gantt-chart opgezet. Dit overzicht is tijdens het traject steeds aangevuld om aan te geven welke activiteiten reeds uitgevoerd zijn en om het verschil aan te tonen tussen de initiële planning en de daadwerkelijke uitvoering van de activiteiten. Op deze manier werd de voortgang van het project overzichtelijk gemaakt. Een versie van de Gantt-chart is hier afgebeeld.



### *Toelichting bij de planning*

- Er wordt in de planning gesproken over *definitieve document trees* (*final document trees*). Er wordt hier gesproken over *definitief*, omdat er tijdens de systeemanalyse ook al document trees ontwikkeld zullen worden, als hulpmiddel bij de analyse en als voorbereiding op de ontwikkeling van de definitieve document trees in week 40. Deze document trees zullen gebruikt worden bij de ontwikkeling van de nieuwe style sheets.
- Er zullen 2 nieuwe lay-outs ontworpen worden en er zullen dus ook 2 (sets) nieuwe style sheets ontwikkeld worden (*first & second new style sheet* in de planning). De eerste lay-out zal zoveel mogelijk in de bestaande look & feel van de website zijn. De tweede lay-out zal een verbeterde lay-out zijn. Voor beide lay-outs zal uiteraard ook een printversie ontworpen worden. Beide lay-outs moeten bereikt worden in overeenstemming met de Webcontent Manager en e-Business Manager, dus dit is een iteratief proces op zichzelf.
- Het gehele traject wordt binnen 19 weken volbracht. (Normaal gesproken duurt de afstudeerstage 18 weken; in dit geval zijn er 19 weken beschikbaar, omdat er rekening gehouden wordt met een week vakantie, i.v.m. de kerstdagen). Als leidraad voor een fasering kan gekeken worden naar de activiteiten die de IAD methodiek beschrijft. De IAD-ontwikkelmethodiek kent de volgende fases;
  - ✓ Definitiestudie
  - ✓ Pilotontwikkeling
  - ✓ Invoering

In dit project wordt de IAD variant RAD toegepast. Hierbij wordt alleen de fase pilotontwikkeling geïtereerd.

Zoals al eerder vermeld is, moet alle documentatie, die aan het bedrijf wordt opgeleverd, in het Engels geschreven worden; met de bedrijfsmentor wordt overeengekomen, dat alleen delen van het Plan van Aanpak die expliciet betrekking hebben op de opdracht vertaald worden. Het gaat om onderdelen zoals de probleemstelling, doelstelling en de planning. Achtergrondinformatie zoals de informatie over het bedrijf en de afdeling is voor Avery Dennison niet interessant en wordt daarom niet opgenomen in de documentatie, die opgeleverd wordt aan de opdrachtgever. In week 2 wordt het Plan van Aanpak volledig afgerond.

### 3.3 Zelfstudie

Een groot onderdeel van de definitiestudie bestaat in dit project uit zelfstudie. Tijdens dit project is het nodig zelfstudie te doen op een tweetal gebieden;

- Cascading Style Sheets (CSS)
- Applicatieontwikkeling in Lotus Notes, voor het ontwikkelen van het management systeem voor de style sheets

#### 3.3.1 Cascading Style Sheets

Bij mij was de basiskennis van CSS reeds aanwezig aan het begin van het project. Voor de opdracht was het echter nodig om een grondig inzicht te ontwikkelen in de mogelijkheden en de beperkingen bij het gebruik van CSS 2, zodat CSS zo effectief en efficiënt mogelijk toegepast kan worden. Om dit te bereiken is zelfstudie naar dit onderwerp gedaan tijdens het project.

Voor deze zelfstudie is leermateriaal beschikbaar gesteld door Avery Dennison; voor de zelfstudie had ik een boek tot mijn beschikking ('Eric Meyer on CSS - Mastering the Language of Web Design' – geschreven door Eric Meyer). Dit boek behandelt 13 los van elkaar staande (fictieve) projecten, waarbij verschillende aspecten op het gebied van CSS aan bod komen. Elk hoofdstuk bestaat uit één zo'n project. Alle bestanden die nodig zijn om die projecten uit te voeren, zijn online beschikbaar, zodat alles wat ik in het boek leer, meteen in de praktijk gebracht kon worden. In de eerste week is een planning voor deze zelfstudie gemaakt, waarin in de eerste weken een aantal dagen per week aan de zelfstudie besteed wordt. Tijdens de zelfstudie worden aantekeningen bijgehouden van de geleerde technieken, die van belang zijn voor dit project. De zelfstudie voor CSS wordt in week 5 afgerond. Ik heb voor deze zelfstudie alle projecten uitgevoerd die in het boek behandeld worden.

Een aantal hoofdstukken in het bijzonder waren zeer nuttig voor het ontwikkelen van de style sheets binnen dit project, omdat in deze hoofdstukken onderwerpen aan bod komen, die binnen dit project van belang zijn; deze zijn hieronder kort toegelicht.

- *Converting an Existing Page* – In dit hoofdstuk wordt een bestaande pagina van een website besproken, waarvan alle uiterlijke kenmerken hard gecodeerd zijn in de HTML. In dit hoofdstuk wordt behandeld hoe de pagina aangepast wordt, zodat alle uiterlijke kenmerken gedefinieerd zijn in een style sheet.
- *Styling for print* – In dit hoofdstuk wordt behandeld hoe CSS gebruikt kan worden voor printbare versies van pagina's. Binnen dit project worden ook printbare pagina's ontwikkeld; een deel van het systeem concept is later dan ook gebaseerd op de technieken die binnen dit hoofdstuk aan bod komen.
- *Multicolumn Lay-out & Positioning a Better Design* – In deze hoofdstukken komt uitvoerig aan bod hoe elementen gepositioneerd kunnen worden op een pagina, zonder daarbij gebruik te hoeven maken van ingewikkelde tabelstructuren. Omdat ik binnen dit project ook te maken zou krijgen met zulke tabelstructuren waarvoor een andere oplossing bedacht moest worden, is dit hoofdstuk ook zeer nuttig geweest.

In alle hoofdstukken werd extra aandacht besteed aan de verschillende *bugs* en eigenaardigheden die voorkomen bij de verschillende moderne browsers. Aangezien ik binnen dit project ook voor verschillende browsers ontwikkel, was deze informatie zeer nuttig.

Naast het boek 'Eric Meyer on CSS' had ik ook de beschikking over een digitale CSS library. Dit is een web-based document, dat beschikbaar is op het netwerk van de afdeling. Het document behandelt de volledige specificatie van CSS 2.0 en is tijdens dit project regelmatig gebruikt als naslagwerk.

### 3.3.2 Lotus Notes

Uit de opdrachtomschrijving komt voort, dat het management systeem voor de style sheets in Lotus Notes ontwikkeld wordt. De zelfstudie voor applicatieontwikkeling in Lotus Notes was nodig, aangezien ik nog nooit eerder met het betreffende pakket gewerkt had.

In eerste instantie was gepland om de studie voor CSS en de studie voor ontwikkelen in Lotus Notes te combineren en beide af te ronden in de eerste weken van het project. In de eerste weken van het project is kennis op het gebied van ontwikkelen in Lotus Notes echter nog niet nodig. Verder zou het van belang zijn om tijdens de zelfstudie van beide onderwerpen praktijkervaring op te doen met de geleerde technieken. Aangezien er in het begin van het project nog geen helder beeld was van hoe het systeem er precies uit komt te zien, is ervoor gekozen om de studie voor ontwikkeling in Lotus Notes te verschuiven tot vlak voor en tijdens de ontwikkeling van het management systeem. Dit heeft het namelijk mogelijk gemaakt tijdens deze zelfstudie delen van het management systeem te ontwikkelen in conceptvorm, en zo de geleerde technieken meteen in praktijk te brengen. Nu was namelijk wel bekend hoe het systeem eruit kwam te zien. Deze indeling van de zelfstudie leverde nog een extra winst op, omdat de programmatuur van deze conceptdelen hergebruikt kunnen worden in het uiteindelijke systeem.

Wanneer er voor gekozen was meteen aan het begin van het project met deze zelfstudie te beginnen, was het natuurlijk ook mogelijk om praktijkervaring op te doen met de geleerde technieken, door zelf kleine systemen te ontwikkelen. Dit zou dan echter kleine stukjes programmatuur opleveren, die verder nergens voor gebruikt kunnen worden.

Een nadeel van deze herindeling van de zelfstudie, was het feit dat ik met deze beslissing een extra risico heb genomen. Door deze zelfstudie verder op te schuiven, zullen eventuele problemen, die tijdens deze zelfstudie kunnen optreden, ook later aan het licht komen. Ik heb deze beslissing uiteraard eerst met de bedrijfsmentor besproken. Uit dit gesprek bleek, dat ik eventueel extra ondersteuning van één van de andere medewerkers van de afdeling kan krijgen, mocht ik tijdens de zelfstudie van ontwikkelen in Lotus Notes in problemen komen.

Om bovenstaande redenen is de zelfstudie voor het ontwikkelen in Lotus Notes opgeschoven tot vlak voor de start van de ontwikkeling van het management systeem voor de style sheets. Deze zelfstudie zal verder aan bod komen bij de bespreking van het management systeem in de ontwikkelfase; hoofdstuk 4.4 *Het Management Systeem*.

### 3.4 Uitvoeren van de systeemanalyse

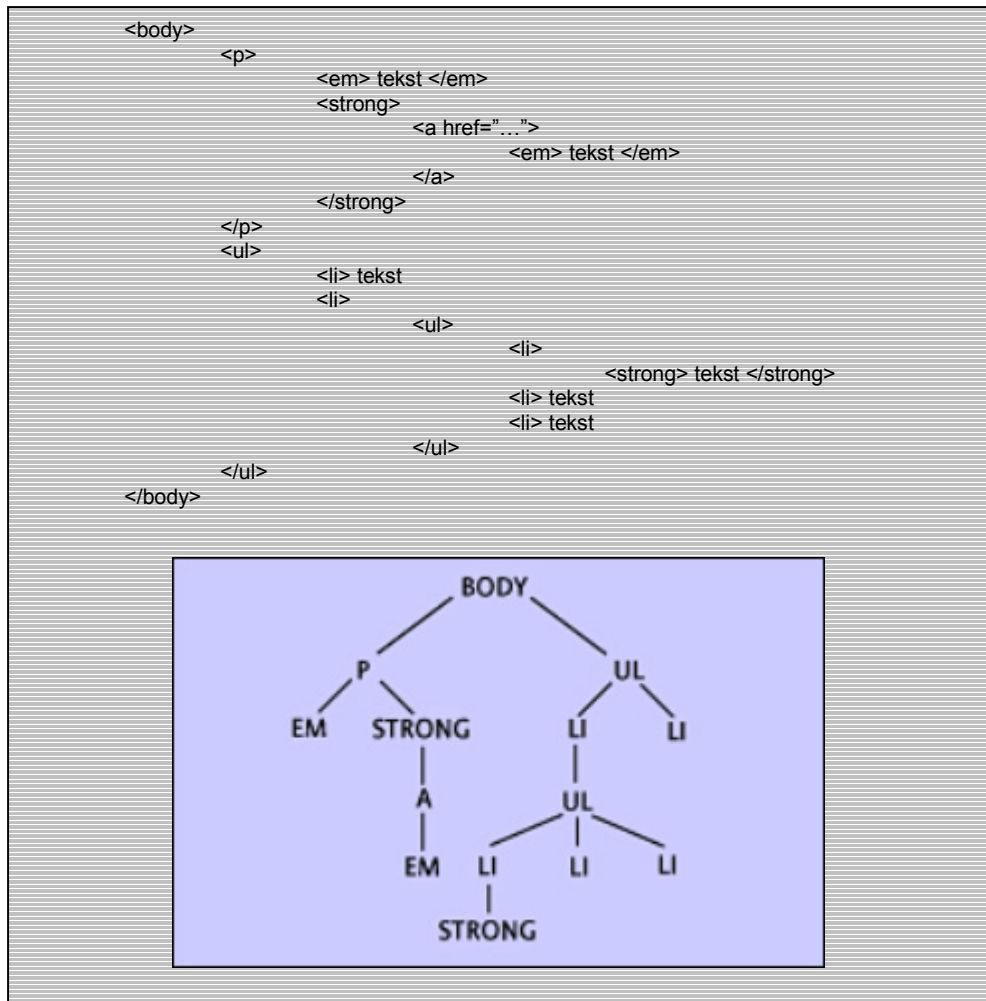
Een belangrijk onderdeel van de definitiestudie van dit project was de systeemanalyse. De analyse bestaat uit twee delen;

- De analyse van de website
- De analyse van het management systeem voor het menu van de website

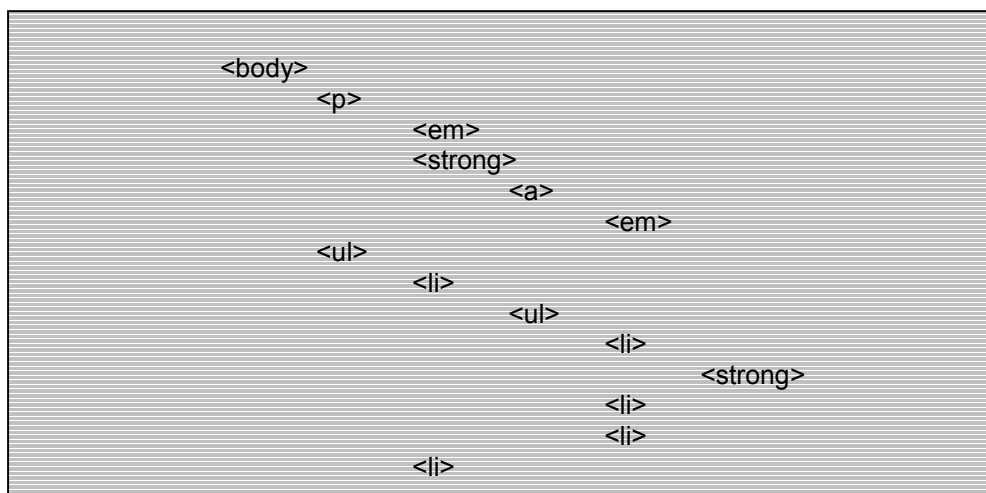
De analyse van de website heeft als belangrijkste doelen de beginsituatie in kaart te brengen en voldoende input te genereren voor de ontwikkeling van een aantal document trees voor de pagina's van de website.

Document trees zijn in feite schematische overzichten van de structuur van de betreffende pagina en beschrijven uit welke elementen een pagina is opgebouwd en hoe deze elementen met elkaar samenhangen. HTML-documenten zijn gebaseerd op een hiërarchie van elementen. Deze hiërarchie is visueel te maken door deze in een zogenaamde boomstructuur weer te geven. Binnen die structuur zijn alle HTML elementen van een webpagina afgebeeld, waarbij elk element een *child* en/of een *parent* is van een ander element. (De meeste elementen zijn zowel child als parent; het element, dat bovenaan de boomstructuur staat is alleen parent, elementen die helemaal onderaan de structuur staan, zijn alleen child) Hieronder is een voorbeeld getoond van een willekeurig fragment HTML code, met daarbij de bijbehorende document tree.





Vanwege de complexiteit en grote omvang van de document trees, die in dit project ontwikkeld worden, is een andere manier gekozen om de document trees weer te geven; een manier, die minder ruimte inneemt en toch dezelfde informatie toont. In onderstaand voorbeeld is te zien hoe de document trees er in dit project uitzien. Het is dezelfde document tree als het voorbeeld, dat hierboven besproken is.



Een *child* kan verschillende eigenschappen overnemen van een *parent*. Wanneer er bijvoorbeeld voor het element *Body* gedeclareerd is, dat de gebruikte tekst *zwart* is, zal alle tekst die voorkomt in een *child* van het element *Body* deze eigenschap overnemen en dus ook *zwart* zijn. Dit verschijnsel wordt ook wel aangeduid met *inheritance*.

De ontwikkeling van de document trees is een belangrijke stap binnen dit project, aangezien de ontwikkeling van de style sheets gebaseerd is op deze document trees. Daarom was het van groot belang, dat de systeemanalyse uitgebreid aandacht zou krijgen. Het resultaat van de analyse van de website is een aantal structuren, die de lay-out van de verschillende pagina's beschrijven. Op basis van deze structuren kan vervolgens een set document trees ontwikkeld worden, die samen alle pagina's van de website beschrijven.

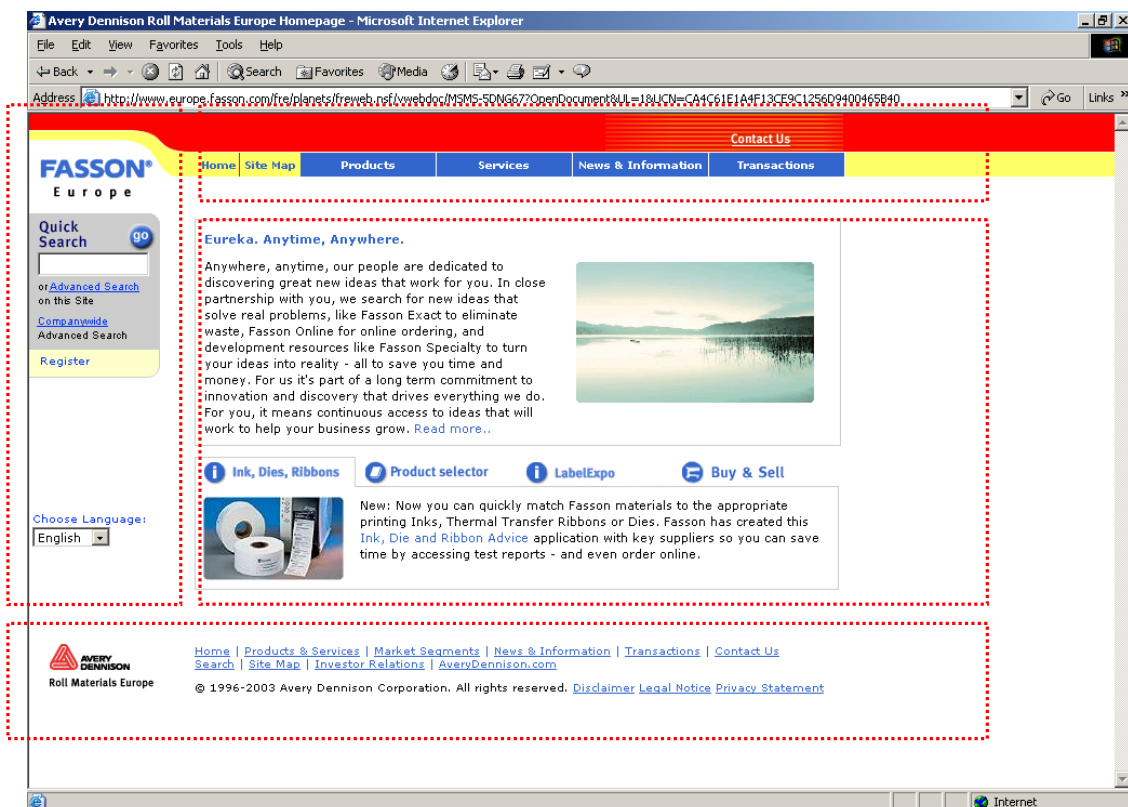
In feite heeft elke pagina een eigen document tree. Het doel is echter om uiteindelijk een set (zo min mogelijk) trees te ontwikkelen voor de gehele website. De reden hiervoor is het feit, dat de document trees uiteindelijk als basis gebruikt worden bij de ontwikkeling van de CSS-documenten. Het is de bedoeling, dat de style sheets over de hele website gebruikt kunnen worden; de style sheets, die ontwikkeld gaan worden, moeten dus gelden voor alle pagina's. De document tree waarop de style sheets gebaseerd worden, moet dus ook voor alle pagina's gelden. Het was dus nodig standaard structuren te ontdekken, om zo document trees te kunnen ontwikkelen die voor meerdere pagina's geldig zijn. In de volgende paragraaf, *3.4.1 Analyse van de website*, zal beschreven worden hoe de analyse van de website uitgevoerd is en hoe de uiteindelijke document trees ontwikkeld zijn.

Naast een analyse van de website voor de ontwikkeling van de document trees wordt ook een analyse gedaan van het huidige management systeem, dat gebruikt wordt voor het menu. Hiervoor bestaat een applicatie in Lotus Notes, die in zeer veel opzichten lijkt op de applicatie die ontwikkeld zal gaan worden als management systeem voor de style sheets. Van deze applicatie wordt een analyse gedaan; er is binnen de documentatie van de systeemanalyse een beschrijving gegeven van het uiterlijk van de applicatie en wat de mogelijkheden zijn. Verder is een globale technische structuur uitgewerkt; dit als voorbereiding op de ontwikkeling van de technische structuur van het management systeem voor de style sheets. De analyse van het management systeem voor het menu zal besproken worden in paragraaf *3.4.3 Analyse van het management systeem voor het menu*.

### 3.4.1 Analyse van de website

De eerste stap binnen deze analyse was het beschrijven van de website met betrekking tot de uiterlijke kenmerken. Hiervoor zijn screenshots gebruikt van verschillende pagina's van de website met daarbij een beschrijving van de belangrijkste elementen. Het uiteindelijke doel is het ontdekken van standaard lay-outstructuren, die gebruikt worden binnen de website.

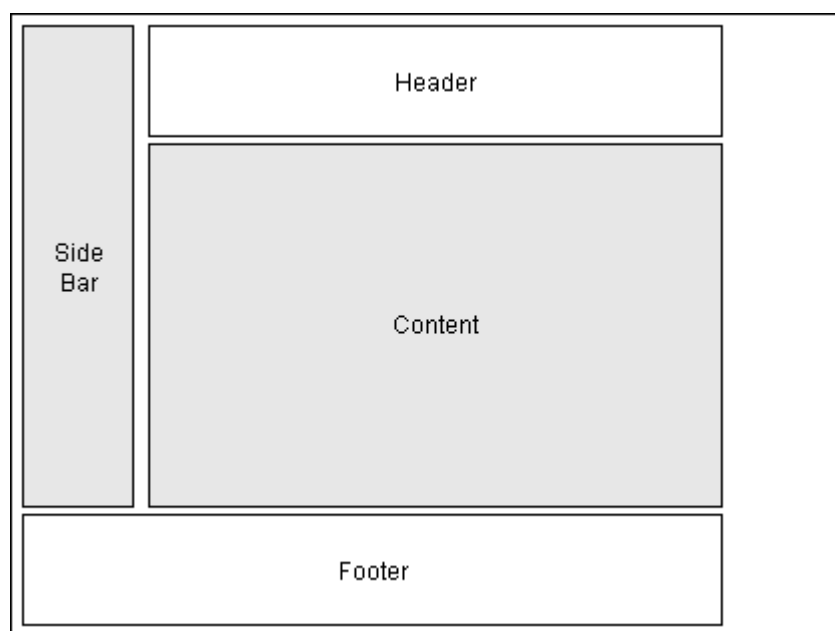
Als eerste is er bepaald welke delen van een pagina altijd hetzelfde zijn. Er wordt namelijk gebruik gemaakt van een aantal statische delen die op elke pagina exact hetzelfde uiterlijk hebben. Na alle pagina's bekeken te hebben is bepaald, dat elke pagina gebruik maakt van dezelfde indeling. Op onderstaande afbeelding is een screenshot te zien van de homepage van de website.



De vlakverdeling van deze pagina komt overeen met die van elke andere pagina. De pagina is in te delen in vier blokken, allen met min of meer een eigen functie. De vier blokken zijn op elke andere pagina ook terug te vinden; in de afbeelding zijn de blokken aangegeven met een rood kader. Sommige delen zijn exact hetzelfde als op alle andere pagina's; dit zijn de statische delen. Andere delen kunnen op verschillende pagina's een andere inhoud hebben; dit zijn de dynamische delen.

- Het bovenste blok bevat het menu en een contact link. Dit deel zal vanaf nu *header* genoemd worden.
- Het linkerblok bevat het Fasson Europe logo, een zoekmodule (Quick Search, voor zoekopdrachten op de website) en een taalkeuze. Dit deel zal vanaf nu *sidebar* genoemd worden.
- Het rechter blok bevat de content van de betreffende pagina. Dit deel zal vanaf nu *content* genoemd worden.
- Het onderste blok bevat informatie over copyright en enkele links. Dit deel zal vanaf nu *footer* genoemd worden.

Er is vervolgens een schematisch overzicht gemaakt van de homepage; dit is op onderstaande afbeelding weergegeven. Elke pagina van de website is op deze manier opgebouwd, uit dezelfde blokken. De header en footer vormen de statische delen en zijn dus voor alle pagina's gelijk. Eén enkele lay-outstructuur is voldoende om deze delen van de pagina te beschrijven. De sidebar en de content vormen de dynamische delen. De inhoud van deze delen kan variëren.



Om te bepalen hoeveel verschillende standaard lay-outstructuren er gebruikt worden, moet er bepaald worden hoeveel verschillende lay-outstructuren er binnen de dynamische delen voorkomen. Wanneer dan namelijk blijkt, dat binnen een blok meerdere lay-outstructuren voorkomen, zal voor dat blok per lay-out een categorie opgesteld worden. In dit geval is één enkele lay-outstructuur namelijk niet genoeg om het betreffende deel van de pagina te beschrijven. Elke lay-outstructuur krijgt dus een eigen categorie, zodat pagina's met een gelijke lay-outstructuur bij elkaar in een categorie gezet kunnen worden. Op deze manier is er een aantal basisstructuren ontstaan, welke als basis hebben gediend voor de verdere ontwikkeling van de document trees.

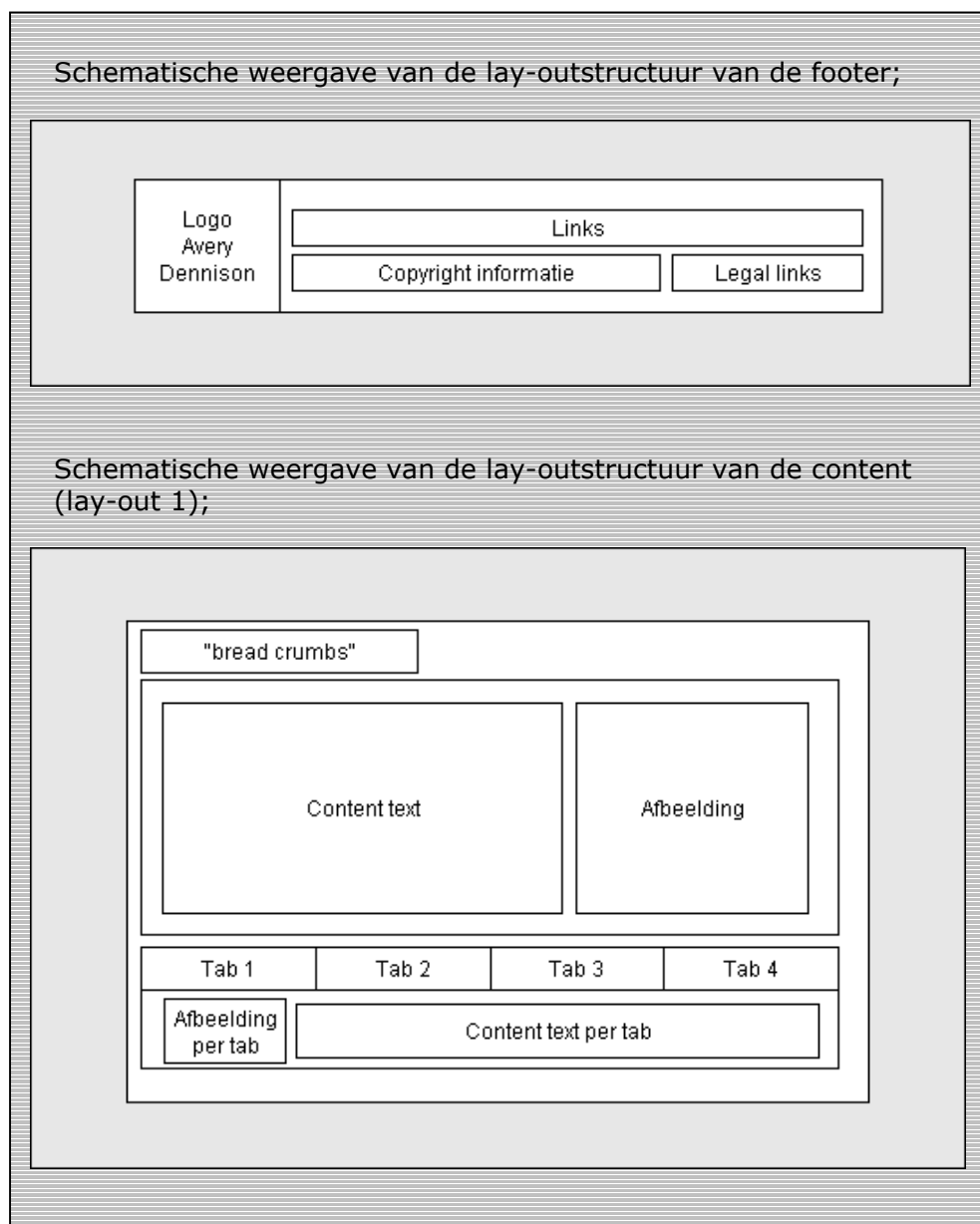
Voor de sidebar, het eerste dynamische blok, is bepaald, dat altijd dezelfde lay-outstructuur gebruikt wordt. Dit is gedaan door alle pagina's van de website te bekijken. Er zijn dan twee verschillende invullingen terug te vinden van de sidebar; deze gebruiken allebei dezelfde lay-outstructuur. Het is voor dit deel dus niet nodig om verschillende categorieën van structuren op te stellen, één lay-outstructuur is voldoende om de sidebar van de website te beschrijven.

Op dezelfde manier is voor het deel met de content bepaald, dat hier wel gebruik gemaakt wordt van verschillende lay-outstructuren. Nadat alle pagina's bestudeerd zijn, wordt vastgesteld, dat er voor de content drie verschillende lay-outstructuren gebruikt worden. Het feit, dat er voor het content blok drie verschillende lay-outs gebruikt worden, betekent dus, dat er voor dit blok ook drie lay-outstructuren opgesteld zullen worden. Voor de overige blokken (sidebar, header en footer) zal steeds één lay-outstructuur per blok voldoende zijn om de lay-out van het betreffende blok te beschrijven. Totaal worden er dus 6 verschillende lay-outstructuren opgesteld, wat als gevolg heeft, dat er uiteindelijk ook 6 document trees ontwikkeld zullen worden. (Deze kunnen samengevoegd worden tot 3 verschillende document trees, waarvan een gedeelte steeds hetzelfde is. Dit is namelijk het gedeelte, dat de header, footer en sidebar beschrijft.)

Na het bepalen van de verschillende lay-outstructuren zijn de structuren visueel gemaakt, door middel van schematische overzichten van de verschillende blokken. Bij elke structuur is ook steeds een korte beschrijving gegeven van de inhoud van het betreffende deel van de website. Er worden totaal 6 overzichten gemaakt van lay-outstructuren;

- Eén overzicht voor de header
- Eén overzicht voor de sidebar
- Eén overzicht voor de footer
- Drie overzichten voor de content (omdat binnen de content gebruik gemaakt kan worden van 3 verschillende lay-outstructuren)

Hieronder worden twee voorbeelden getoond van die schematische overzichten van de lay-outstructuur per blok.



### 3.4.2 Ontwikkelen van de document trees

De volgende belangrijke stap in het toewerken naar de document trees, heeft als doel de HTML-elementen te ontdekken die verantwoordelijk zijn voor de lay-out van de verschillende structuren. Voor elke structuur is dan ook bekeken welke elementen dit zijn en hoe deze met elkaar samenhangen. Van deze elementen kan dan per lay-outstructuur vervolgens een document tree opgesteld worden. Alle trees samen vormen dan de document tree voor een lay-outstructuur. Aangezien er drie verschillende lay-outs zijn, zullen er uiteindelijk ook drie document trees opgesteld worden. Delen van deze trees zullen overeenkomen, aangezien de header, footer en sidebar op elke pagina dezelfde lay-out hebben.

Het ontdekken van de HTML-elementen verantwoordelijk voor de lay-out kan op twee manieren;

- 'van binnen uit' - de code van de pagina's wordt geanalyseerd. Aan de hand van deze code wordt vastgelegd welke elementen hieruit verantwoordelijk zijn voor de lay-out.
- 'van buiten af' - In dit geval worden de pagina's geanalyseerd zoals ze op het scherm te zien zijn. Vervolgens wordt bepaald welke HTML-elementen nodig zijn om dit te bereiken.

Er is gekozen voor de tweede manier, aangezien de pagina's in hun huidige vorm zeer complex zijn opgebouwd. Het is in eerste instantie heel moeilijk te zien hoe de pagina's zijn opgebouwd. Aangezien het hier alleen gaat om stijlkenmerken (uiterlijk van de pagina's) is het heel goed mogelijk om de pagina's 'van buiten af' te analyseren. Er zijn ook veel elementen aanwezig die niet noodzakelijk zijn voor om de huidige lay-out te bereiken, als gevolg van vele aanpassingen en toevoegingen die steeds gedaan zijn aan de code. Op deze manier wordt bereikt, dat die elementen die voor dit project niet van belang zijn uit de code verdwijnen. (de doelstelling is uiteindelijk ook een 'schonere' makkelijker onderhoudbare code)

Om een overzicht te krijgen van de betreffende HTML-elementen, is voor elke lay-outstructuur een pagina gekozen. Deze pagina's zijn opnieuw in HTML uitgewerkt. Het resultaat hiervan was een drietal pagina's, die lokaal op mijn PC opgeslagen waren; uit elke categorie één pagina. Deze pagina's maken dus alleen gebruik van HTML en bevatten verder geen verschillende scripts (zoals bijvoorbeeld JavaScript). Op deze manier is dus zeker, dat het volledige uiterlijk het resultaat is van enkel HTML. Dan zijn dus ook de HTML-elementen verantwoordelijk voor die lay-out aan te wijzen.

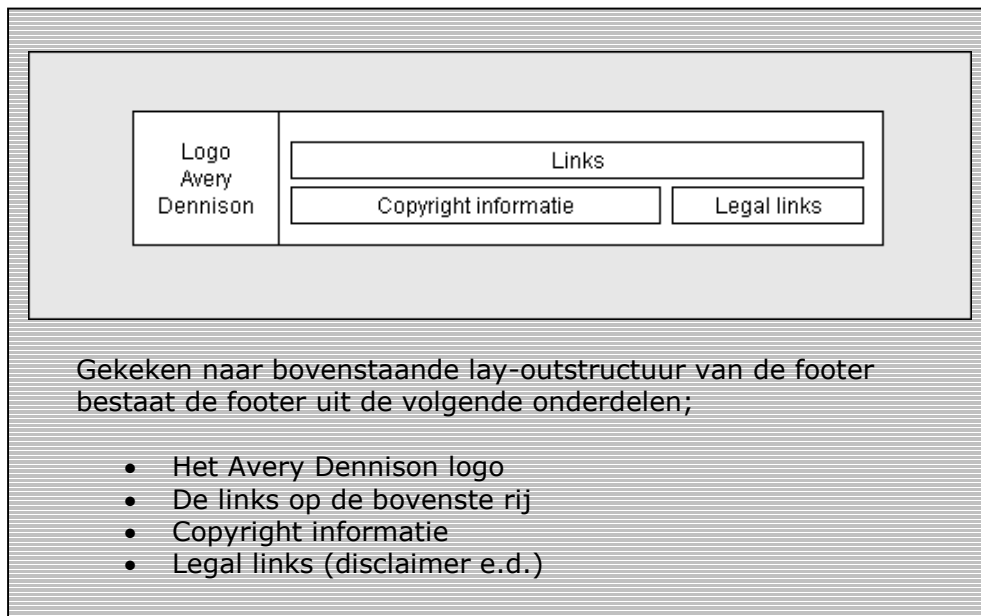
Er zijn drie pagina's uitgewerkt in HTML;

- pagina met header, footer, sidebar en content lay-out 1
- pagina met header, footer, sidebar en content lay-out 2
- pagina met header, footer, sidebar en content lay-out 3

Ook dit is per blok gedaan. Op deze manier werd bekend welke HTML elementen verantwoordelijk zijn voor de lay-out van het betreffende blok. Vanuit die gegevens is vervolgens de document tree voor het betreffende blok af te leiden. Het proces, dat hieronder behandeld wordt, beschrijft de afleiding van de document tree voor de footer. Dit proces is dus voor alle blokken uitgevoerd, de footer wordt hier behandeld als voorbeeld.

#### De afleiding van de document tree voor de footer

Eerst is op basis van de lay-outstructuur van de footer bepaald hoe de footer opgebouwd is;





Om tot een overzicht te komen van HTML-elementen die van belang zijn voor de lay-out van de footer, zal dus voor bovenstaande punten bepaald moeten worden uit welke HTML-elementen (HTML-tags) ze opgebouwd zijn. Deze elementen worden in een schema uiteengezet;

<b>Avery Dennison Logo</b>	
<i>Opgebouwd uit</i>	<i>HTML tag</i>
Een enkele afbeelding / link	<img> <a>

<b>Links op bovenste rij</b>	
<i>Opgebouwd uit</i>	<i>HTML tag</i>
Tekst (verticale streepjes)	<font>
Links	<a>


  

<b>Copyright informatie</b>	
<i>Opgebouwd uit</i>	<i>HTML tag</i>
Tekst	<font>

<b>Legal links</b>	
<i>Opgebouwd uit</i>	<i>HTML tag</i>
Links	<a>

In de onderstaande afbeelding worden alle HTML-tags weergegeven, met voorlopige id's en/of classes;

<div> <div>  <p>Roll Materials Europe</p> </div> <div> <a href="#">Home</a>   <a href="#">Products &amp; Services</a>   <a href="#">Market Segments</a>   <a href="#">News &amp; Information</a>   <a href="#">Transactions</a>   <a href="#">Contact Us</a>  <a href="#">Search</a>   <a href="#">Site Map</a>   <a href="#">Investor Relations</a>   <a href="#">AveryDennison.com</a>            © 1996-2003 Avery Dennison Corporation. All rights reserved. <a href="#">Disclaimer</a> <a href="#">Legal Notice</a> <a href="#">Privacy Statement</a> </div> </div>		
<a class=foot_links> <font class=foot_links>	<a class=img_link> <img id=avery_logo>	<font class=copyright>
		<a class=legal_links>

Nu alle HTML elementen voor de footer bekend zijn, kan een document tree voor dit deel opgesteld worden. Alle bovenstaande elementen in de juiste volgorde leveren de onderstaande document tree voor de footer op;

```
<div id=footer>
  <a class=img_link>
    <img id=avery_logo>
  <font class=foot_links>
    <a class=foot_links>
  <font class=copyright>
  <a class=legal_links>
```

Op dezelfde manier als hierboven beschreven is voor de footer, zijn ook de document trees voor de overige blokken opgesteld. Zo zijn zes document trees ontstaan, voor elke lay-outstructuur één. De volledige document trees zijn terug te vinden in bijlage A, als onderdeel van de definitiestudie.

De analyse van de website heeft langer geduurd dan gepland was. Door de complexiteit en omvang van de pagina's, heeft vooral het opnieuw in HTML uitwerken van de pagina's veel tijd gekost. Als gevolg hiervan is de definitiefase iets uitgelopen. Het is wel van groot belang geweest, dat er voldoende aandacht werd besteed aan deze analyse. Op basis van deze analyse zijn namelijk de document trees ontwikkeld. Deze zijn gebruikt bij de ontwikkeling van de style sheets.

### 3.4.3 Analyse van het management systeem voor het menu

Na het afronden van de analyse van de website is een korte analyse gedaan van het management systeem, dat in gebruik is voor het onderhoud van het menu op de website. De reden voor deze analyse is een aantal overeenkomsten tussen dit systeem en het te ontwikkelen systeem voor het onderhoud van de style sheets. De volgende overeenkomsten zijn te ontdekken;

- Het management systeem voor het menu is ontwikkeld in Lotus Notes. Uit de opdrachtomschrijving komt voort, dat het management systeem voor de style sheets ook in Notes ontwikkeld wordt.
- Het systeem voor het menu draait net als het systeem voor de style sheets in een Lotus Notes omgeving.
- De techniek achter het menu op de website is JavaScript. Met het management systeem worden complete JavaScript documenten gegenereerd. Met het systeem voor de style sheets worden ook documenten gegenereerd; in dit geval gaat het om CSS-documenten.

Vanwege bovenstaande overeenkomsten was het nuttig om het management systeem voor het menu nader te bestuderen. Ik heb eerst een gesprek gehad met de Notes programmeur, die het systeem ontwikkeld heeft, als voorbereiding op het opstellen van een functionele en een technische beschrijving van het systeem. Op basis van de gegevens uit de opdrachtomschrijving met betrekking tot het management systeem voor de style sheets is besproken wat de overeenkomsten waren tussen het systeem voor het menu en het systeem voor de style sheets. Verder heb ik uitleg gehad over de mogelijkheden van het systeem en de globale technische achtergrond.

De resultaten van dit gesprek zijn verwerkt tot een globale beschrijving van de technische structuur van het management systeem voor het menu. Deze beschrijving is later gebruikt als basis voor de ontwikkeling van de technische structuur van het management systeem voor de style sheets.

Verder heb ik zelf de mogelijkheden van het systeem uitgeprobeerd en screenshots gemaakt van de verschillende schermen uit het systeem. Op basis hiervan en de eerder genoemde gespreksresultaten is een functionele beschrijving gemaakt, waarin per scherm beschreven wordt wat de mogelijkheden van het systeem zijn.

### 3.5 Opstellen van de systeemeisen

Omdat het binnen dit project in feite om twee verschillende systemen gaat (enerzijds de style sheets, anderzijds het management systeem voor deze style sheets), wordt ook bij het opstellen van de systeemeisen een tweedeling aangehouden; er worden apart systeemeisen opgesteld voor de style sheets en het management systeem. Als gevolg van het feit, dat er twee verschillende systemen ontwikkeld worden, moet er rekening gehouden worden met twee verschillende doelgroepen.

#### 3.5.1 Doelgroepen

Binnen de definitiestudie is het bepalen van de doelgroep een van de gebruikelijke activiteiten. Enerzijds worden binnen dit project style sheets ontwikkeld worden voor de website; hier moet dus rekening gehouden worden met de bezoekers van de website als doelgroep. Anderzijds wordt er een systeem ontwikkeld voor het onderhoud van die style sheets. De doelgroep van dit systeem bestaat uit de medewerkers van de divisie RME, die met het systeem gaan werken. Er wordt binnen dit project dus onderscheid gemaakt tussen twee verschillende doelgroepen;

- Doelgroep van de website (doelgroep voor de ontwikkelde style sheets)
- Doelgroep van het management systeem

Doelgroep style sheets: Het bepalen van de doelgroep van de te ontwikkelen style sheets was snel gedaan. Hier gaat het namelijk om de bezoekers van de website [www.europe.fasson.com](http://www.europe.fasson.com). Dit zijn zowel geregistreerde klanten als bezoekers. De geregistreerde klanten zijn regelmatige bezoekers van de site; zij zijn bekend met de huidige *look & feel* van de site. Aangezien de invoering van de style sheets geen veranderingen tot gevolg zal hebben voor de *look & feel* van de website, is er verder geen onderzoek gedaan naar het gedrag van de bezoekers van de website; zij zullen immers niets merken van de veranderingen.

Doelgroep management systeem: De doelgroep van het te ontwikkelen management systeem bestaat uit medewerkers van de afdeling e-Business, de Webcontent Manager in het specifiek. Deze groep mensen heeft een gedegen kennis van het vakgebied en zijn dus bekend met verschillende aspecten van met betrekking tot webdesign en de mogelijkheden en beperkingen van CSS en HTML. Het management systeem geldt voor deze doelgroep als een tijdsbesparend middel en moet meer flexibiliteit bieden bij het aanpassen van uiterlijke kenmerken van de website. Tijdens een interview met de web-content manager is vastgesteld wat het kennisniveau is van de toekomstige gebruikers.

Bij het opstellen van de systeemeisen wordt natuurlijk rekening gehouden met de doelgroepen. Aangezien de bezoekers van de website niets van de veranderingen zullen merken, is er verder geen onderzoek gedaan naar deze doelgroep. Bij de doelgroep van het management systeem ligt dit anders. Medewerkers van de afdeling e-Business zullen namelijk gaan werken met dit nieuwe systeem; vooral de webcontent-manager zal veel gebruik maken van het systeem. Daarom zijn de systeemeisen voor het management systeem voor een groot deel opgesteld in samenwerking met de webcontent-manager. In de volgende paragrafen zal de werkwijze bij het opstellen van de systeemeisen beschreven worden.

### 3.5.2 Style sheets

Voor de style sheets is eerst een initiële lijst met systeemeisen opgesteld, gebaseerd op de opdrachtomschrijving. Het ging hierbij dus om eisen die direct uit de opdrachtomschrijving voortvloeien. Enkele voorbeelden van deze eisen zijn hier opgesomd;

- De huidige look & feel van de website blijft in eerste instantie onveranderd. Dit moet bereikt worden met een combinatie van HTML en CSS.
- De website dient compatible te zijn met de volgende browsers;
  - ✓ Microsoft Internet Explorer 5
  - ✓ Microsoft Internet Explorer 6
  - ✓ Netscape 7
- Voor alle pagina's zal een printlay-out ontwikkeld worden.

De lijst is verder aangevuld, op basis van de conclusies die voortkwamen uit de systeemanalyse en de zelfstudie die gedaan is voor CSS. Hiervan zijn enkele voorbeelden;

- De nieuwe structuur van de website, in de vorm van een combinatie van HTML en CSS, zal een zo 'schoon' mogelijke code opleveren. Zo zal bijvoorbeeld het grote aantal *geneste tabellen*<sup>2</sup> zoveel mogelijk verdwijnen. (eis gebaseerd op zelfstudie en systeemanalyse)
- Voor het realiseren van de printlay-out zal gebruik gemaakt worden van aparte style sheets. (eis gebaseerd op zelfstudie)

---

<sup>3</sup> Geneste tabel: Dit is een tabel die geplaatst is binnen een cel van een andere tabel. Binnen de geneste tabel kan weer een volgende tabel voorkomen, waardoor zeer complexe tabelstructuren kunnen ontstaan.

Op deze manier is een lijst met initiële systeemeisen ontstaan. Deze lijst is besproken met de bedrijfsmentor. Enerzijds had dit gesprek als doel de eisen te controleren en te valideren. Anderzijds kreeg ik hierbij de gelegenheid om in samenwerking met de bedrijfsmentor tijdens dit gesprek de lijst met eisen aan te vullen, tot de definitieve lijst met systeemeisen. De belangrijkste conclusie van het gesprek was, dat er een zekere meetbaarheid gecreëerd moest worden met betrekking tot de juistheid van de nieuwe aangepaste HTML en de CSS-documenten. Er is daarom besloten om de volgende eisen toe te voegen;

- Alle pagina's moeten zowel door de CSS-validator als de HTML-validator van het W3C<sup>3</sup> goedgekeurd worden. Dit zijn tools die valideren of de HTML- en CSS-documenten volgens de standaarden van het W3C zijn opgesteld. Deze tools zijn online beschikbaar op <http://www.w3c.org>.
- Alle pagina's van de site zullen gebruik maken van het DOCTYPE voor HTML 4.01, strict<sup>4</sup>

Alle systeemeisen voor de style sheets zijn terug te vinden in de bijlagen, als onderdeel van de definitiestudie.

---

<sup>4</sup> W3C: De afkorting voor World Wide Web Consortium. Dit is een internationaal samenwerkingsverband van bedrijven, dat open specificaties (feitelijk standaarden) ontwikkelt om de onderlinge operationele samenwerking van web-gerelateerde producten te vergroten. W3C Aanbevelingen zijn ontwikkeld door werkgroepen samengesteld door de consortiumleden en daarbij genodigde experts.

<sup>5</sup> DocType: Door de vele verschillen in de manier van renderen die door de verschillende oude browsers werd toegepast, is het heel lang niet mogelijk geweest websites te maken volgens één standaard. (Attributen zoals *margins* en *padding*s worden door verschillende browsers anders geïnterpreteerd) Dit heeft als gevolg dat pagina's er in verschillende browsers compleet anders uit kunnen zien; de nieuwe browsers renderen strict volgens de regels die het W3C opgesteld heeft (*strict rendering*), oudere browsers zullen in veel gevallen een andere lay-out van de pagina laten zien, deze maken gebruik van *loose rendering*. In dit laatste geval gebeurt het renderen niet strict volgens de richtlijnen, gesteld door het W3C. De gekozen oplossing om verschillen in browsers zoveel mogelijk te beperken, maakt gebruik van het DocType dat ieder document moet bevatten. Hierin kan aangegeven worden welke renderingmode door de browser gehanteerd moet worden. Een *loose* rendering wil zeggen dat de pagina gerendered wordt, rekening houdende met de bugs en eigenaardigheden van de oudere (versie 4) browsers. Een *strict* rendering wil zeggen dat de specificaties van het W3C voor HTML, CSS en andere lay-out talen zo goed mogelijk gevolgd wordt. Een declaratie van een DocType is in feite alleen een link naar een online document waarin het gedrag voor het renderen is vastgelegd.

### 3.5.3 Management systeem

Bij het opstellen van de systeemeisen voor het management systeem is een soortgelijke werkwijze aangehouden als bij het opstellen van de eisen voor de style sheets. Eerst is er een initiële lijst gemaakt met systeemeisen. Hierbij gaat het weer om eisen die rechtstreeks uit de opdrachtschrijving voortkomen. Voorbeelden van deze eisen zijn in onderstaand kader genoemd.

- Het systeem moet het mogelijk maken wijzigingen aan te brengen in de CSS bestanden, zonder dat hiervoor de code van de pagina's aangepast hoeft worden.
- Het systeem moet tijd besparen bij het wijzigen van stijkenmerken.
- Het systeem zal draaien binnen een Lotus Notes omgeving, binnen de interface van Lotus Notes.

De opgestelde lijst is vervolgens met de bedrijfsmentor besproken, wederom met de doelen de eisen te valideren en aan te vullen. In dit gesprek komt ook naar voren, dat het gewenst is dat het systeem het mogelijk maakt de CSS bestanden te wijzigen zonder dat daar ook maar enige code voor aangepast hoeft te worden. Op deze manier wordt namelijk bereikt, dat het onderhoud van uiterlijke kenmerken op de website volledig gecentraliseerd wordt; alle aanpassingen kunnen met het management systeem gedaan worden. De volgende eisen worden daarom aan de lijst toegevoegd;

- Alle eigenschappen voor de verschillende HTML-elementen gedefinieerd in de nieuwe style sheets moeten door middel van het management systeem aan te passen zijn.
- Het systeem moet complete style sheets genereren, opgemaakt volgens de standaarden van het W3C, optimaal gebruik makend van de mogelijkheden van CSS.

Op deze manier is een 'voorlopige' lijst met systeemeisen voor het management systeem opgesteld. *Voorlopig*, omdat de eisen verder uitgebreid worden, op basis van de conclusies die getrokken worden uit het gesprek, dat ik heb gehad met de webcontent-manager. Dit gesprek was nodig, omdat de webcontent-manager de belangrijkste gebruiker van het nieuwe management systeem zal worden; dit gesprek had als doel haar eisen en wensen met betrekking tot het systeem te inventariseren.

Op basis van de hierboven genoemde 'voorlopige' lijst van systeemeisen voor het management systeem is een voorlopig storyboard opgesteld, wat als eerste voorstel dient voor het concept van het systeem. Het storyboard is een tekstuele beschrijving van welke stappen een gebruiker in het nieuwe systeem dient uit te voeren bij het wijzigen van stijlkenmerken op de website. Dit storyboard wordt ook besproken tijdens het betreffende gesprek met de Webcontent Manager en de feedback die ik hierop gekregen heb, is gebruikt voor het uitbreiden van de lijst met systeemeisen. Tijdens dit gesprek wordt dus ook de reeds opgestelde lijst van voorlopige systeemeisen voor het management systeem besproken; ook de feedback op deze eisen kan gebruikt worden voor het aanvullen van de voorlopige lijst.

Na het gesprek met de Webcontent Manager kunnen de laatste eisen aan de voorlopige lijst worden toegevoegd, zodat deze lijst nu ook compleet is. Enkele voorbeelden van de eisen die na dit gesprek worden toegevoegd zijn;

- Om de pagina's zo consistent mogelijk te houden wat stijl betreft, moet de keuze van op te geven waardes voor stijlattributen een zekere beperking krijgen. Wanneer een waarde gekozen kan worden, zal deze gekozen worden uit een beperkte lijst. Wanneer een waarde door de gebruiker ingevuld wordt, zal dit beperkt worden door een maximum en minimum waarde.
- Er moet gebruik gemaakt worden van een heldere, eenvoudige navigatie.
- De navigatie moet verlopen via een stap-voor-stap-methode. Geleidelijk navigeert de gebruiker naar het element dat bewerkt moet worden.



De feedback die gegeven werd op het voorlopige storyboard is later ook gebruikt om het definitieve storyboard op te stellen, welke gebruikt is voor het systeemconcept voor het management systeem. Dit wordt besproken in paragraaf 3.6.3 *Management systeem*.

Tijdens het bovengenoemde gesprek met de Webcontent Manager wordt mij gevraagd een extra functionaliteit<sup>6</sup> te ontwikkelen voor de website en deze functionaliteit als systeemeis op te nemen. Aangezien deze functionaliteit buiten de *scope* van de opdracht valt, moet ik duidelijk maken wat de grenzen zijn van het project en dat dit niet binnen dit project uitgevoerd zal kunnen worden en dat hiervoor een andere oplossing bedacht moet worden.

Alle systeemeisen voor het management systeem zijn terug te vinden in de bijlagen als onderdeel van de definitiestudie.

---

<sup>6</sup> Extra functionaliteit voor de website: Het is mogelijk om via de website producten te bestellen. Bij dit proces is het voor een klant heel onduidelijk hoeveel stappen er nog te gaan zijn, voordat het product besteld is. Mij wordt gevraagd om een tijdbalk te ontwikkelen, die de bezoeker precies vertelt waar hij of zij zich in het bestelproces bevindt.

### 3.6 Opstellen van het systeemconcept

Bij het opzetten van het systeemconcept wordt gestreefd naar een globale omschrijving van de oplossing, die bedacht is om de doelstelling van het project te halen. Voor de verschillende onderdelen van het systeem wordt de functionaliteit vastgelegd en een globale opzet ontwikkeld om deze functionaliteit ook te realiseren. Gedurende de pilotontwikkeling worden deze concepten verder uitgewerkt. Ook hier is weer een tweedeling aangehouden; er is zowel een concept ontwikkeld voor de style sheets als een concept voor het management systeem. In de volgende paragrafen zal de werkwijze bij het opstellen van deze concepten beschreven worden.

#### 3.6.1 Style sheets

In de huidige situatie wordt de lay-out van een pagina voor het grootste deel bepaald door de code van die pagina. Er is wel een style sheet aanwezig maar deze wordt lang niet optimaal gebruikt. De meeste stijkenmerken zijn 'hard' in de HTML gecodeerd. Ook is de code erg rommelig geworden door de vele ad-hoc toevoegingen die er gedaan zijn in de loop der jaren.

Op basis van de volgende punten is het systeemconcept voor de style sheets ontwikkeld;

- Resultaten van de systeemanalyse. Hieruit is gebleken hoe de pagina's zijn ingedeeld in blokken, welke standaard lay-outs er gebruikt worden en welke elementen noodzakelijk zijn voor de lay-out en welke elementen weggelaten kunnen worden.
- De opgestelde systeemeisen. Hierbij was het bijvoorbeeld belangrijk, dat de lay-out van de website niet zal veranderen en dat de gebruikte tabellen in de nieuwe situatie zoveel mogelijk weggelaten zullen worden.

Om aan te tonen welke wijzigingen gedaan moeten worden in de HTML van de pagina's, zal op de volgende bladzijde een voorbeeld gegeven worden van de code die in de huidige situatie gebruikt wordt om de footer weer te geven. Vervolgens zal er getoond worden hoe deze code er in de nieuwe situatie uitziet. De bestaande HTML-code wordt gewijzigd, zodat deze geen attributen meer bevat, die stijkenmerken definiëren. Ook zal de bestaande tabelstructuur verdwijnen; er zal geen tabel meer gebruikt worden maar in plaats van deze structuur zal gebruik gemaakt worden van het HTML element *DIV*<sup>7</sup>. Vervolgens zal een deel van de style sheet gegeven worden, om aan te tonen welke oplossingen gekozen kunnen worden, om dezelfde lay-out als in de oude situatie te bereiken.

---

<sup>7</sup> DIV: Een HTML element van het type block-level element. Hiermee kan een blok gedefinieerd worden op een HTML pagina. Een DIV is in feite een container die inhoud groepeerd. Op deze manier kunnen aparte stijlen gedefinieerd worden voor de verschillende DIV elementen.

De volgende code is een fragment van een code uit een van de pagina's van de website en toont de HTML, die in de huidige situatie gebruikt wordt om de footer weer te geven;

```
<TABLE cellSpacing=0 cellPadding=0 width=600 align=left border=0>
<TBODY>
<TR>
<TD vAlign=top width=153><A href="http://www.averydennison.com/">
<IMG height=48 alt=Avery src="Avery Dennison Roll Materials Europe
Homepage_files/ADRMLogo.gif" width=153 border=0></A>
</TD>
<TD vAlign=top width="90%">
<TABLE cellSpacing=0 cellPadding=0 width=600 align=left border=0>
<TBODY>
<TR>
<TD class=link10px vAlign=top align=left width=600>

<A class=link10px href="http://www.europe.fasson.com/etc>Home</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Products &
Services</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Market
Segments</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>News &
Information</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Transactions</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Contact Us</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Search</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Site Map</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Investor
Relations</A> |
<A class=link10px
href="http://www.europe.fasson.com/etc>AveryDennison.com</A>

</TD>
</TR>
<TR>
<TD vAlign=top align=left width=600 height=10>
<IMG height=10 src="Avery Dennison Roll Materials Europe
Homepage_files/space.gif" width=1>
</TD>
</TR>
<TR>
<TD class=copyright vAlign=top align=left width=600>
© 1996-2003 Avery Dennison Corporation. All rights reserved.&nbsp;
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Disclaimer</A>&nbsp;
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Legal
Notice</A>&nbsp;
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Privacy Statement</A>
<BR><BR>
</TD>
</TR>
</TBODY>
</TABLE>
</td>
</tr>
</tbody>
</table>
```

De volgende afbeelding laat de output zien van bovenstaande code; de footer van de website.



Tijdens de zelfstudie voor CSS is geleerd, dat alle vetgedrukte code in bovenstaand fragment in de nieuwe versie weggelaten kan worden. Hiervoor kunnen twee redenen zijn;

- Het gaat om een stijldeclaratie die in de nieuwe situatie in de style sheet gedefinieerd wordt. Voorbeelden hiervan zijn; *vAlign=top* en *width=153*.
- Het element maakt onderdeel uit van de tabel die gebruikt is voor de verdeling van de verschillende elementen binnen de footer. Aangezien deze tabel in de nieuwe situatie niet meer aanwezig is, kunnen deze elementen ook verwijderd worden. Dit heeft ook tot gevolg, dat de pixelgrote afbeelding space.gif, waarmee ruimte gecreëerd wordt tussen de cellen van de tabel in de nieuwe situatie niet meer terug te vinden is.

De code voor de footer in de nieuwe situatie is vergelijkbaar met het volgende fragment;

```
<div id=footer>
  <A href="http://www.averydennison.com/" class=img_link>
  <IMG alt=Avery src="Avery Dennison Roll Materials Europe
Homepage_files/ADRMElogo.gif" id=avery_logo>
</A>
  <font class=foot_links>
  <A class=foot_links href="http://www.europe.fasson.com/etc>Home</A> |
  <A class=foot_links href="http://www.europe.fasson.com/etc>Products &
Services</A> |
  <A class=foot_links href="http://www.europe.fasson.com/etc>Market Segments</A> |
  <A class=foot_links href="http://www.europe.fasson.com/etc>News &
Information</A> |
  <A class=foot_links href="http://www.europe.fasson.com/etc>Transactions</A> |
  <A class=foot_links href="http://www.europe.fasson.com/etc>Contact Us</A> |
  <A class=foot_links href="http://www.europe.fasson.com/etc>Search</A> |
  <A class=foot_links href="http://www.europe.fasson.com/etc>Site Map</A> |
  <A class=foot_links href="http://www.europe.fasson.com/etc>Investor Relations</A> |
  <A class=foot_links
href="http://www.europe.fasson.com/etc>AveryDennison.com</A>
  </font>
  <font class=copyright>
  <br>© 1996-2003 Avery Dennison Corporation. All rights reserved.&nbsp;
  </font>
  <A class=legal_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Disclaimer</A>&nbsp;
  <A class=legal_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Legal
Notice</A>&nbsp;
  <A class=legal_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Privacy Statement</A>
</div>
```

In bovenstaande code is te zien, dat de tabelstructuur vervangen is door een *DIV* element met het id *footer*, waarbinnen zich alle elementen voor de footer bevinden. Er zijn geen attributen meer in de HTML-code, die betrekking hebben op lay-out van de pagina; deze attributen worden allemaal in de style sheet gedefinieerd. Door de juiste stijlen te schrijven voor de verschillende elementen van het fragment, kan dezelfde lay-out bereikt worden.

De nieuwe code in combinatie met de volgende stijlen, leveren dezelfde lay-out van de footer op;

```
Div#footer {
    width: 753px;}

Img#avery_logo {
    width: 153px;
    height: 48px;
    vertical-align: top;
    border: 0;
    float: left;}

Font.foot_links{
    font-size: 10px;
    color: #3366cc;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

Font.copyright {
    line-height: 3.5em;
    font-size: 10px;
    color: #000000;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

A.foot_links:link {
    font-size: 10px;
    color: #3366cc;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

A.foot_links:hover {
    font-size: 10px;
    color: #ff0000;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

A.foot_links:visited {
    font-size: 10px;
    color: #3366cc;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

A.legal_links:link {
    font-size: 10px;
    color: #3366cc;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

A.legal_links:hover {
    font-size: 10px;
    color: #ff0000;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

A.legal_links:visited {
    font-size: 10px;
    color: #3366cc;
    font-family: Verdana, Arial, Helvetica, sans-serif;}
```

Bovenstaand fragment zou onderdeel kunnen zijn van de style sheet in de nieuwe situatie. In de nieuwe situatie zal deze style sheet gegenereerd worden door het management systeem. Op de HTML pagina zelf wordt vervolgens de style sheet gelinkt.

Het concept voor de style sheets beschrijft dus in feite 2 stappen;

- Het schrijven van de stijlen in de style sheets.
- Het aanpassen van de HTML op de pagina's. Hierbij gaat het op het filteren van de overbodige HTML (zoals de geneste tabelstructuren en de hard gecodeerde stijlenmerken). Daarnaast worden aan de HTML-elementen classes en/of id's toegekend, zodat de elementen uniek geïdentificeerd kunnen worden in de style sheet documenten.

### 3.6.2 Style sheets voor het printen

Onderdeel van de opdracht is het ontwikkelen van een printlay-out voor elke pagina. Het is echter niet nodig om van elke pagina een tweede versie te ontwikkelen, welke dan als printversie zou kunnen voldoen. Tijdens de zelfstudie voor CSS is namelijk geleerd, dat CSS de mogelijkheid biedt om stijlen te declareren, die beschrijven hoe de HTML-elementen eruit moeten zien op papier. Bij het gebruik van een externe style sheet kan namelijk aangegeven worden, voor welk medium de stijldeclaraties op die style sheet bedoeld zijn, bijvoorbeeld voor papier of het scherm. Vanuit een HTML-document kan gelinkt worden naar meerdere style sheets en voor elk van deze style sheets kan aangegeven worden voor welk medium de style sheet gebruikt moet worden. Er worden dan ook twee style sheets ontwikkeld worden; een style sheet die beschrijft hoe de elementen op het scherm moeten worden afgebeeld en een style sheet die beschrijft hoe deze elementen eruit moeten zien op papier. Beide style sheets zullen op de pagina's aangeroepen worden. Er zijn verder geen extra aanpassingen in de HTML nodig; een aparte style sheet voor het printen is voldoende om een printlay-out te verwezenlijken.

Om dit te testen is het idee eerst uitgetest met een test pagina. Dit was een simpele HTML-pagina, met daarop een aantal teksten. Ook worden hierbij twee simpele style sheets ontwikkeld, waaraan de pagina gekoppeld wordt. Op de eerste style sheet is gedeclareerd hoe de tekst er op het scherm uit moet zien; welk lettertype wordt er op het scherm gebruikt, welke kleur heeft de tekst, welke maat wordt er voor het font gebruikt, etc. Ook wordt op de style sheet aangegeven welke elementen zichtbaar zijn op het scherm. De pagina is ook aan een tweede style sheet gelinkt, die hetzelfde doet voor dezelfde elementen, alleen dan voor de presentatie op papier. Op deze manier kunnen dus met verschillende style sheets meerdere stijldeclaraties opgegeven worden voor hetzelfde HTML-element; voor elk medium een aparte declaratie. Wijzigingen in de HTML-code zijn hiervoor niet nodig. Deze techniek bleek goed te werken en kon dus opgenomen worden binnen het systeem concept.

De stijldeclaraties van de style sheet voor het printen hebben geen effect op de presentatie van de elementen op het scherm. Andersom, wanneer een pagina afgedrukt wordt, hebben de stijldeclaraties van de style sheet voor het scherm geen invloed op de presentatie van de elementen op papier. Bij elk medium wordt namelijk alleen gebruik gemaakt van de style sheet voor dat betreffende medium. Wanneer een pagina bijvoorbeeld uitgeprint wordt, wordt de style sheet voor het scherm volledig genegeerd.

### 3.6.3 Management systeem

Om te bereiken, dat de nieuwe style sheets gemakkelijk onderhouden kunnen worden, wordt een management systeem voor de style sheets ontwikkeld. Met dit systeem zal het mogelijk zijn wijzigingen aan te brengen in de gebruikte style sheets. Met het systeem moeten de waarden van de stijlattributen op de style sheets gewijzigd kunnen worden. Vervolgens zullen de opgegeven waarden verzameld moeten worden, zodat deze samengesteld kunnen worden tot een volledig CSS-document. Het concept voor het management systeem bestaat in feite uit twee delen;

- Een functionele beschrijving in de vorm van een storyboard
- Een globale technische structuur

De ontwikkeling van beide delen zal in deze paragraaf toegelicht worden.

#### Het storyboard

Om een duidelijke functionele beschrijving te kunnen geven van het systeem, is het storyboard, dat eerder gebruikt is bij het opstellen van de systeemeisen voor het management systeem, voor de ontwikkeling van het concept van het management systeem verder uitgewerkt. Het was een tekstuele beschrijving van de stappen die een gebruiker in het nieuwe systeem dient uit te voeren bij het wijzigen van stijlkenmerken op de website. Gebaseerd op de conclusies van het gesprek, dat ik heb gehad met de webcontent-manager bij het opstellen van de systeemeisen, is de tekstuele beschrijving verder aangevuld met schetsen van schermen van het management systeem. Deze conclusies hadden vooral betrekking op de systeemeisen die de gebruiksvriendelijkheid van het systeem zullen moeten verhogen. Het gaat om bijvoorbeeld de volgende eisen;

- Er moet gebruik gemaakt worden van een heldere, eenvoudige navigatie.
- De navigatie moet verlopen via een stap-voor-stap-methode. Geleidelijk navigeert de gebruiker naar het element dat bewerkt moet worden.

Met het aanvullen van het storyboard is een scenario ontstaan, waarin beschreven staat wat de mogelijkheden van het nieuwe systeem zullen zijn. In dit storyboard wordt namelijk stap voor stap beschreven wat de nodige en mogelijke acties zijn bij het wijzigen van stijlkenmerken van de pagina's.

Hieronder is een fragment te zien van het storyboard (stap 3), zoals deze is opgesteld voor het systeemconcept. Het complete storyboard is terug te vinden in de bijlagen, als onderdeel van de definitiestudie.

3. *Er wordt nu een overzicht gegeven van de stijlattributen die bij de gekozen selector en medium horen. In het geval van links, zijn dit vier overzichten, aangezien een link vier statussen kan hebben (link, hover, visited en active)*

Category	
Select your category:	Links
Links	
Select your type:	Links in footer for: Screen
Links in footer for SCREEN	
Save	
<div>a.foot_links : link</div> <div>font-size 10 px</div> <div>color <input type="text"/></div> <div>font-family <input type="text"/></div> <div>background <input type="text"/></div>	<div>a.foot_links : hover</div> <div>font-size 10 px</div> <div>color <input type="text"/></div> <div>font-family <input type="text"/></div> <div>background <input type="text"/></div>
<div>a.foot_links : visited</div> <div>font-size 10 px</div>	<div>a.foot_links : active</div> <div>font-size 10 px</div>

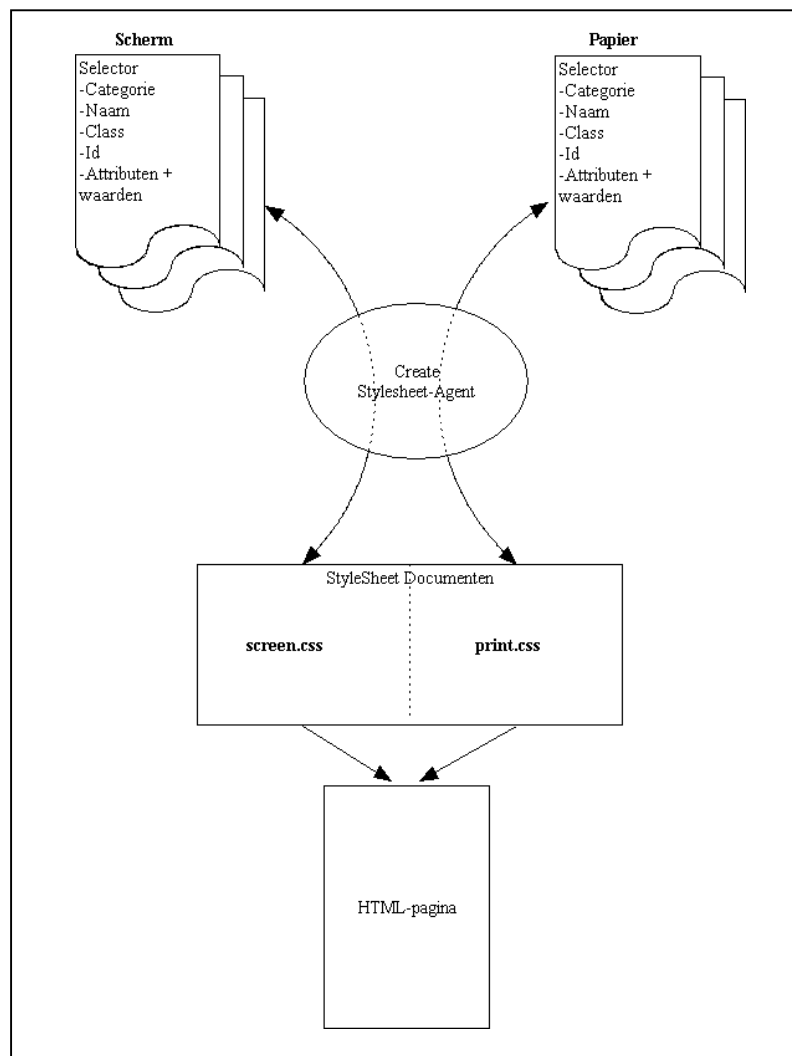
Schets van de interface van het management systeem



### Technische structuur

Het management systeem wordt ontwikkeld in Lotus Notes. Aangezien ik in dit vroege stadium waarin het systeemconcept opgesteld wordt nog niet voldoende kennis heb van de technische aspecten van Lotus Notes, is het op dat moment onmogelijk om een technisch gedetailleerd concept te ontwikkelen voor het management systeem. De systeemeisen voor het systeem heb ik daarom besproken met één van de Lotus Notes specialisten werkzaam op de afdeling. De conclusie van dat gesprek was, dat de functionaliteit van het management systeem voor de style sheets sterke overeenkomsten heeft met de functionaliteit van het systeem, dat al gebruikt wordt voor het onderhoud van het menu op de website. Dit systeem is ook geanalyseerd, wat besproken is in paragraaf 3.4.3. *Analyse van het management systeem voor het menu*. Op basis van de technische structuur van het systeem voor het menu kon, in samenwerking met de betreffende Notes specialist, een globale technische structuur voor het management systeem voor de style sheets opgesteld worden, met de eerder opgestelde systeemeisen als uitgangspunt.

In onderstaand schema wordt een overzicht getoond van de globale technische structuur voor het management systeem, gebaseerd op de technische structuur van het systeem, dat gebruikt wordt voor het onderhoud van het menu op de website.



Toelichting bij globale technische structuur

In de gebruikte database, die ontwikkeld wordt voor het systeem, zijn voor elke selector twee documenten opgeslagen met daarin alle gegevens die nodig zijn voor die betreffende selector. Voor elke selector worden *twee* documenten gebruikt, omdat er uiteindelijk ook *twee* style sheets samengesteld moeten worden; het eerste document bevat een stijldeclaratie voor het scherm, waarin dus is vastgelegd hoe het betreffende HTML-element op het scherm getoond moet worden. In het andere document wordt vastgelegd hoe het betreffende element op papier gepresenteerd moet worden. In elk document worden de volgende gegevens opgeslagen;

- Categorie; bijvoorbeeld tekst, links, afbeeldingen, etc.
- Naam; bijvoorbeeld *font* voor tekst of *a* voor links.
- Class en/of id; dit is om de selector uniek te kunnen identificeren.
- Attributen en waarden; dit zijn de eigenlijke stijldeclaraties, bijvoorbeeld *font-size: 10px;*.

De inhoud van een document kan bijvoorbeeld zijn;

Categorie:	<i>links</i>
Naam:	<i>font</i>
id:	<i>footer</i>
Font-size:	<i>10px</i>
Font-family:	<i>Verdana, Arial</i>

Deze gegevens bij elkaar vormen een stijldeclaratie voor het element font met het id footer;

Font#footer {font-size: 10px; font-family: Verdana, Arial;}

De categorie wordt niet gebruikt in de stijldeclaratie. Deze wordt alleen gebruikt om binnen het systeem met verschillende categorieën te kunnen werken.

De *agent*<sup>8</sup> verzamelt de documenten en stelt deze samen tot een kloppend style sheet document. Ook dit gebeurt twee maal; van de documenten voor het scherm wordt een style sheet gemaakt voor het medium *screen*, van de documenten voor papier wordt een style sheet gemaakt voor het medium *print*. Vervolgens worden beide style sheet documenten op de (aangepaste) HTML pagina's aangeroepen. Het gaat hier niet om een bestaande agent, de agent maakt onderdeel uit van de applicatie en wordt ook door mij binnen dit project ontwikkeld.

In het bovenstaande voorbeeld zijn maar een aantal attributen voor een selector te zien. Het gaat hier om een *voorbeeld*; een document uit het uiteindelijke systeem bevat uiteraard veel meer gegevens.

<sup>8</sup> Agent: Een stand-alone programma, dat geautomatiseerd specifieke taken uit kan voeren in één of meerdere databases. Een agent wordt ook wel *macro* genoemd.

### 3.7 Opstellen van het pilotplan

Het laatste deel van de definitiestudie is het pilotplan. In dit hoofdstuk wordt een pilotstructuur en –strategie bepaald. Er wordt een indeling gemaakt in pilots; dit levert een overzicht op van de afzonderlijke pilots die ontwikkeld zullen worden. Verder wordt er een planning gemaakt voor de ontwikkeling van de pilots. De pilotstrategie geeft aan op welke manier de pilots ontwikkeld worden. Dit kan bijvoorbeeld *parallel* zijn. In dat geval worden de pilots tegelijk ('naast elkaar') ontwikkeld. Hier tegenover staat serieel ontwikkelen, waarbij de pilots *na* elkaar ontwikkeld worden. Verder kan er ook nog onderscheid gemaakt worden tussen een strategie waarbij gestreefd wordt naar brede of diepe functionaliteit. Een pilot die ontwikkeld wordt met diepe functionaliteit wil zeggen, dat de pilot zo gekozen is, dat de deel van het systeem tot in detail uitgewerkt wordt, voordat aan het volgende deel begonnen wordt. Bij brede functionaliteit worden de pilots zo gekozen, dat bij oplevering van een pilot veel functionaliteit reeds ingebouwd is in de pilot, maar deze functionaliteit wordt nog niet tot in detail uitgewerkt. Een situatie waarbij de opdrachtgever snel veel resultaten wil zien leent zich ideaal voor deze manier van ontwikkelen; na de eerste iteratie is er al veel functionaliteit ontwikkeld waarop de opdrachtgever zijn of haar feedback kan geven.

#### 3.7.1 Verdeling van pilots

Binnen dit project is er gekozen voor pilots met een diepe functionaliteit; de pilots die ontwikkeld worden bevatten functionaliteit die volledig uitgewerkt wordt voor die betreffende pilot. Een reden hiervoor is, dat de opdrachtgever niet direct resultaten hoefde te zien om deze te kunnen evalueren. Een tweede reden voor deze keuze is de grote complexiteit van de website. Een database in Lotus Notes is complexer om mee te werken dan een 'traditionele' relationele database. Omdat databases in Lotus Notes geen relationele databases zijn en een database in feite een volledige applicatie bevat, is de relatie tussen de data minder inzichtelijk dan bij een relationele database. Omdat de website gebruik maakt van 74 verschillende databases, was het gewenst om te kunnen testen met volledige pilots. Op deze manier is het wordt het duidelijk of een pilot werkt of niet. Wanneer er met brede functionaliteit ontwikkeld zou zijn, had er na elke toevoeging van functionaliteit opnieuw getest moeten worden, om te achterhalen welke gevolgen de toevoeging heeft op de werking van de website.

Verder is er een combinatie gebruikt van parallel en serieel ontwikkelen, waarbij elke pilot in twee iteraties ontwikkeld wordt. De voorkeur lag eigenlijk bij volledig serieel ontwikkelen. Aangezien dit project door één persoon wordt uitgevoerd, was het gewenst om de ontwikkeling van de pilots zoveel mogelijk gescheiden te houden. Op deze manier blijven de ontwikkelingen overzichtelijk, omdat er steeds maar aan één pilot tegelijk gewerkt wordt.

Tijdens de definitiefase is de volgende verdeling opgesteld voor de pilots;

- Eerste style sheet + aangepaste HTML-pagina's
- Eerste style sheet voor printen
- Tweede style sheet + aangepaste HTML-pagina's
- Tweede style sheet voor printen
- Database voor het management systeem
- Management systeem voor de style sheets

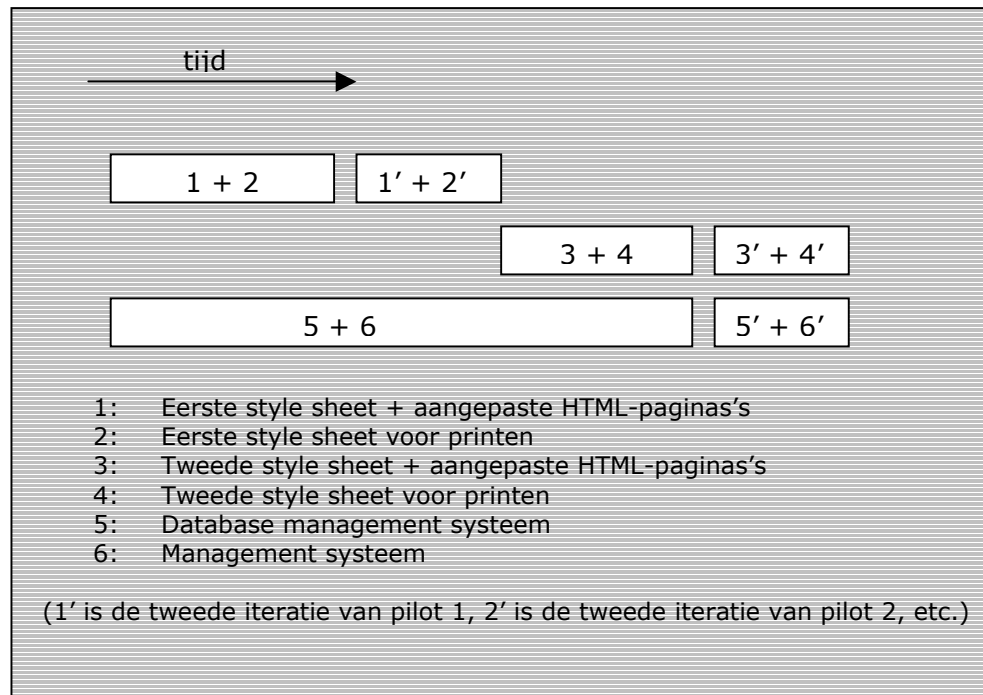
Er is toch een combinatie gebruikt van parallel en serieel ontwikkelen, omdat de geschatte ontwikkeltijd voor het management systeem zodanig hoog was, dat er voor volledige seriële ontwikkeling geen tijd was. Deze pilot wordt namelijk ontwikkeld in Lotus Notes, een omgeving waarin ik nooit eerder ontwikkeld heb. De pilotontwikkeling voor het management systeem bestaat derhalve uit een deel zelfstudie, waardoor de ontwikkeltijd voor het systeem geschat is op ongeveer 5 weken.

De style sheets zullen wel serieel ontwikkeld worden. Zo zal eerst de eerste versie van de style sheets ontwikkeld worden, voor zowel de presentatie op het scherm als de presentatie op papier (printversie). Vervolgens zullen deze pilots geëvalueerd worden. Deze evaluatie geldt als input voor de ontwikkelingen van deze pilots in de tweede iteratie.

Vervolgens zullen de tweede versies van de style sheets ontwikkeld worden. Voor deze ontwikkeling wordt dezelfde indeling toegepast wat betreft iteraties als bij de ontwikkeling van de eerste versie.

Parallel aan de ontwikkeling van de beide style sheets wordt het management systeem ontwikkeld met bijbehorende database. Ook deze ontwikkeling is opgedeeld in twee iteraties, waarbij de uitkomsten van de evaluatie van iteratie 1 weer gelden als input voor de ontwikkelingen in iteratie 2.

In onderstaande overzicht is de verdeling van pilots en iteraties schematisch weergegeven;



### 3.7.2 Pilots en prioriteiten

Het is gebruikelijk om binnen een project per pilot prioriteiten toe te kennen aan de verschillende onderdelen van een pilot. Per functionaliteit wordt dat aangegeven wat de prioriteit is. In het geval, dat door bepaalde omstandigheden het project niet binnen de tijd afgerond kan worden, wordt op basis van die prioriteiten bepaald welke functionaliteit wel en niet ontwikkeld zal worden.

Bij dit project zijn alle onderdelen van een pilot even belangrijk; het heeft voor de opdrachtgever geen zin om een style sheet in te voeren, die niet compleet is. Dan blijven dezelfde problemen bestaan, welke in de opdrachtomschrijving beschreven zijn bij de probleemstelling. In dat geval blijft het namelijk zo, dat stijlenmerken op verschillende manieren worden gedefinieerd; enerzijds in de style sheet, anderzijds hard gecodeerd in de HTML, omdat de style sheet niet compleet is. Voor het management systeem voor de style sheets geldt in feite hetzelfde; ook hier heeft de opdrachtgever niets aan een half product. Een belangrijk doel van het systeem is namelijk het besparen van tijd bij het wijzigen van stijlenmerken op de website. Wanneer er functionaliteit weggelaten zou worden, zou dit betekenen, dat er taken alsnog handmatig uitgevoerd moeten worden. (bijvoorbeeld het linken van de style sheets aan de pagina's) Van tijdsbesparing is dan geen sprake meer. Verder geldt voor het management systeem, dat er een concept is opgesteld, dat niet heel gedetailleerd is. Bij het opstellen van de pilots was dus geen gedetailleerde invulling van het management systeem bekend. Ook dit is een reden geweest, dat er voor de pilots geen prioriteiten per functionaliteit opgesteld zijn.

Om bij problemen binnen het project, zoals bijvoorbeeld tijdnood, toch enig overzicht te hebben met betrekking tot prioriteiten voor dit project, zijn er prioriteiten toegekend aan de pilots zelf. De eerste style sheets voor het scherm en het printen krijgen samen met het management systeem prioriteit 1, omdat hiermee de problemen die in de opdrachtomschrijving genoemd worden, opgelost worden. De tweede style sheets voor het scherm en het printen krijgen prioriteit 2. Deze zijn 'minder belangrijk', omdat het in wezen een verbetering op de eerste style sheets is. Het volgende eenvoudige prioriteitschema is opgesteld;

Pilot	Prioriteit
Eerste style sheet + aangepaste HTML-pagina's	1
Eerste style sheet voor printen	1
Database voor het management systeem	1
Management systeem voor de style sheets	1
Tweede style sheet + aangepaste HTML-pagina's	2
Tweede style sheet voor printen	2

Mochten er problemen ontstaan waardoor de voortgang van het project in gevaar kan komen, wordt in samenwerking met de bedrijfsmentor op basis van dit prioriteiten schema bepaald wat de verdere invulling van het project zal zijn.

## **4. Ontwikkelfase**

### *4.1 Inleiding*

Tijdens de ontwikkelfase worden de pilots ontwikkeld volgens de specificaties uit de definitiestudie. De ontwikkelingen zijn gestart met het ontwikkelen van de eerste style sheet voor het scherm en de daarbij behorende aanpassingen in de HTML code. Parallel hieraan is begonnen met de zelfstudie voor het ontwikkelen in Lotus Notes, welke nodig was voor de ontwikkeling van het management systeem voor de style sheets. Tijdens de zelfstudie is begonnen met het creëren van onderdelen van het management systeem in conceptvorm. Deze onderdelen worden uiteindelijk samengevoegd en uitgebreid tot het uiteindelijke systeem.

In de volgende paragrafen wordt het proces beschreven van deze verschillende ontwikkelingen en de keuzes, die hierbij gemaakt zijn.

## 4.2 Eerste style sheets

De ontwikkeling van de eerste style sheets is gebaseerd op de lokaal opgeslagen HTML pagina's met standaard lay-out, die tijdens de systeemanalyse voor de ontwikkeling van de document trees opnieuw opgesteld zijn. Ook hierbij werd weer per blok (header, sidebar, footer en content) gewerkt; er is voor elk blok apart een CSS file ontwikkeld, volgens het systeem concept, dat tijdens de definitie fase gespecificeerd is. Daarbij behorende wijzigingen in de HTML, zoals aan bod gekomen is bij de bespreking van het systeem concept, zijn eveneens per blok gedaan. Dit is in eerste instantie voor de lokale pagina's gedaan. Er is voor gekozen, om ook per blok de CSS te ontwikkelen en de HTML aan te passen, om zoveel mogelijk overzicht te bewaren. Op deze manier zijn er dus verschillende CSS-documenten met de daarbij behorende aangepaste HTML-documenten ontstaan;

- Voor de header, sidebar en footer zijn voor elk blok één CSS-document en één HTML-document opgesteld. Het gaat hier om de lokaal opgeslagen HTML pagina's. Hiervan is per blok de HTML aangepast. Voor elk blok waren er aparte documenten aanwezig.
- Voor de content zijn drie CSS-documenten en drie HTML-documenten opgesteld. Hierbij was er sprake van drie documenten, als gevolg van de drie verschillende lay-outstructuren, die gebruikt worden voor dit blok. Voor elke lay-out is een apart HTML-document en CSS-document gebruikt.



Zoals is uitgelegd bij de bespreking van de theorie van Lotus Notes (zie interne bijlage 1), is de HTML van de pagina's opgeslagen in Notes databases. De code is ook nergens als complete pagina terug te vinden (zoals in een lokaal opgeslagen pagina alle HTML in één HTML-document staat) maar is verspreid over verschillende forms en subforms die in een database opgeslagen zijn; de website draait tenslotte in een Lotus Notes omgeving. Als laatste stap bij de ontwikkeling van de CSS moeten de wijzigingen, die voor de lokaal opgeslagen pagina's gedaan worden, dus ook nog op de forms en subforms uit de betreffende database op de ontwikkelserver worden doorgevoerd. Hierbij gaat het deels om het wijzigen van Lotus Script en Formula Language, omdat delen van de HTML door hierdoor gegenereerd worden. Daarom worden alle wijzigingen, die gedaan worden in de lokale pagina's, gedocumenteerd. Deze documentatie wordt gebruikt bij het doorvoeren van de wijzigingen op de forms en subforms in de betreffende Lotus Notes databases.

Ik wil hier nogmaals benadrukken, dat binnen dit project alleen voor de pagina's met standaard lay-out ontwikkeld wordt. Wijzigingen in de overige pagina's zullen in een ander project uitgevoerd worden, op basis van de wijzigingen die tijdens dit project gedaan zijn. Voor het maken van deze aanpassingen in de overige pagina's wordt door mij een script geschreven, dat als leidraad gebruikt kan worden. Dit script baseer ik op de wijzigingen die ik aanbreng voor de pagina's met de standaard lay-out in de databases op de ontwikkelserver.

Hieronder wordt beschreven hoe het proces van de ontwikkeling van CSS verlopen is, aan de hand van een beschrijving van de ontwikkeling van CSS voor de footer voor presentatie op het scherm. Dit proces is dus voor alle blokken uitgevoerd, de footer wordt hier behandeld als voorbeeld.

#### De ontwikkeling van CSS voor het scherm voor de footer

De eerste stap bestaat uit het filteren van de HTML zoals beschreven is bij de bespreking van het systeem concept. Het gaat hierbij om de overbodige HTML;

- Het gaat om een stijldeclaratie die in de nieuwe situatie in de style sheet gedefinieerd wordt. Voorbeelden hiervan zijn; `vAlign=top` en `width=153`.
- Het element maakt onderdeel uit van de tabel die gebruikt is voor de verdeling van de verschillende elementen binnen de footer. Aangezien deze tabel in de nieuwe situatie niet meer aanwezig is, kunnen deze elementen ook verwijderd worden. Dit heeft ook tot gevolg, dat de pixelgrote afbeelding `space.gif`, waarmee ruimte gecreëerd wordt tussen de cellen van de tabel in de nieuwe situatie niet meer terug te vinden is.

Eerder dit hoofdstuk is al gezegd, dat ik bij de ontwikkeling van CSS uitgegaan ben van de lokaal opgeslagen HTML-documenten, die opgesteld zijn voor de ontwikkeling van de document trees. Hier wordt nogmaals de code getoond die in de beginsituatie gebruikt werd, om de footer weer te geven, waarbij alle vet gedrukte code overbodig is in de nieuwe situatie.

```
<TABLE cellSpacing=0 cellPadding=0 width=600 align=left border=0>
<TBODY>
<TR>
<TD vAlign=top width=153><A href="http://www.averydennison.com/">
<IMG height=48 alt=Avery src="Avery Dennison Roll Materials Europe
Homepage_files/ADRMLogo.gif" width=153 border=0></A>
</TD>
<TD vAlign=top width="90%">
<TABLE cellSpacing=0 cellPadding=0 width=600 align=left border=0>
<TBODY>
<TR>
<TD class=link10px vAlign=top align=left width=600>

<A class=link10px href="http://www.europe.fasson.com/etc>Home</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Products & amp;
Services</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Market
Segments</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>News & amp;
Information</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Transactions</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Contact Us</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Search</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Site Map</A> |
<A class=link10px href="http://www.europe.fasson.com/etc>Investor
Relations</A> |
<A class=link10px
href="http://www.europe.fasson.com/etc>AveryDennison.com</A>

</TD>
</TR>
<TR>
<TD vAlign=top align=left width=600 height=10>
<IMG height=10 src="Avery Dennison Roll Materials Europe
Homepage_files/space.gif" width=1>
</TD>
</TR>
<TR>
<TD class=copyright vAlign=top align=left width=600>
© 1996-2003 Avery Dennison Corporation. All rights reserved.&nbsp;
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Disclaimer</A>&nbsp;
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Legal
Notice</A>&nbsp;
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Privacy Statement</A>
<BR><BR>
</TD>
</TR>
</TBODY>
</TABLE>
</td>
</tr>
</tbody>
</table>
```

De overbodige HTML wordt uit dit document verwijderd. Uit de document tree van de footer is af te leiden welke elementen dit zijn; alleen de HTML elementen die in de document tree voorkomen, zijn van belang voor de lay-out van de footer, overige code is overbodig. Alle nodige wijzigingen worden gedocumenteerd. Deze documentatie zal later gebruikt worden als leidraad bij het doorvoeren van de wijzigingen in de HTML code uit de databases op de ontwikkelserver. De code van de footer in de nieuwe situatie is als volgt;

```
<div id=footer>
  <A href="http://www.averydennison.com/" class=img_link>
    <IMG alt=Avery src="Avery Dennison Roll Materials Europe
Homepage_files/ADRMLogo.gif" id=avery_logo>
  </A>

  <div id=foot_links>
    <A href="http://www.europe.fasson.com/etc>Home</A> |
    <A href="http://www.europe.fasson.com/etc>Products & Services</A> |
    <A href="http://www.europe.fasson.com/etc>Market Segments</A> |
    <A href="http://www.europe.fasson.com/etc>News & Information</A> |
    <A href="http://www.europe.fasson.com/etc>Transactions</A> |
    <A href="http://www.europe.fasson.com/etc>Contact Us</A> |
    <A href="http://www.europe.fasson.com/etc>Search</A> |
    <A href="http://www.europe.fasson.com/etc>Site Map</A> |
    <A href="http://www.europe.fasson.com/etc>Investor Relations</A> |
    <A href="http://www.europe.fasson.com/etc>AveryDennison.com</A>
  </div>

  <div id=copyright>
    <br>© 1996-2003 Avery Dennison Corporation. All rights reserved.&nbsp;
    <A
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Disclaimer</A>&nbsp;
    <A href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Legal
Notice</A>&nbsp;
    <A href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Privacy
Statement</A>
  </div>
```

Zoals vastgelegd is in het systeemconcept is in bovenstaande code, de tabelstructuur vervangen door een *DIV* element met het id *footer*, waarbinnen zich alle elementen voor de footer bevinden. Er zijn geen attributen meer in de HTML-code, die betrekking hebben op de lay-out van de pagina; deze attributen worden allemaal in de style sheet gedefinieerd. Er zijn een aantal verschillen te ontdekken tussen bovenstaande code en de code zoals deze is opgesteld voor het systeemconcept. Deze verschillen worden hier genoemd en verklaard;

- Er wordt geen gebruik meer gemaakt van het HTML element *font*; deze elementen zijn vervangen door het element *DIV*. (*div id=foot\_links* en *div id=copyright*)
- De links maken geen gebruik meer van classes.

Deze twee verschillen hebben alles met elkaar te maken. Er is voor gekozen, om het element *font* te vervangen met het element *DIV*, omdat met een *DIV* element meer flexibiliteit gecreëerd wordt, met betrekking tot het wijzigen van stijkenmerken. Op deze manier wordt de tekst, die weergegeven wordt binnen de *DIVs*, namelijk een *child* van de betreffende *DIV*. Er wordt hier dus gebruik gemaakt van inheritance-eigenschappen van HTML . Als gevolg hiervan kunnen bijvoorbeeld ook de classes weggelaten worden voor de links; door gebruik te maken van inheritance is het namelijk mogelijk om voor de links toch aparte stijlen te schrijven in het CSS-document. Het element *DIV* heeft meer attributen die gebruikt kunnen worden voor het wijzigen van stijkenmerken dan het element *font*. Zo wordt het bijvoorbeeld mogelijk om met *marges* en *padding*s te werken, wat het positioneren van de elementen op het scherm makkelijker maakt.

Zoals in het systeem concept vastgelegd is, is de volgende stap het opstellen van de style sheet, die ervoor zorgt, dat de nieuwe code van de footer dezelfde uitvoer oplevert als de beginsituatie. Met de geleerde technieken uit de zelfstudie, worden de juiste stijlen opgesteld voor de verschillende elementen van de footer, zodat diezelfde lay-out bereikt wordt. Volgend fragment laat de code zien uit de style sheet voor de footer. De nieuwe code in combinatie met de volgende stijlen, leveren dezelfde lay-out van de footer op;

```
Div#footer {
    width: 753px;
    margin: 0;
    padding: 0;}

img#avery_logo {
    width: 153px;
    height: 48px;
    vertical-align: top;
    border: 0;
    float: left;}

Div#foot_links{
    font-size: 10px;
    color: #3366cc;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

Div#copyright {
    margin-top: 10px;
    font-size: 10px;
    color: #000000;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

Div#foot_links a:link {
    color: #3366cc;}

Div#foot_links a:hover {
    color: #ff0000;}

Div#foot_links a:visited {
    color: #3366cc;}

Div#copyright a:link {
    color: #3366cc;}

Div#copyright a:hover {
    color: #ff0000;}

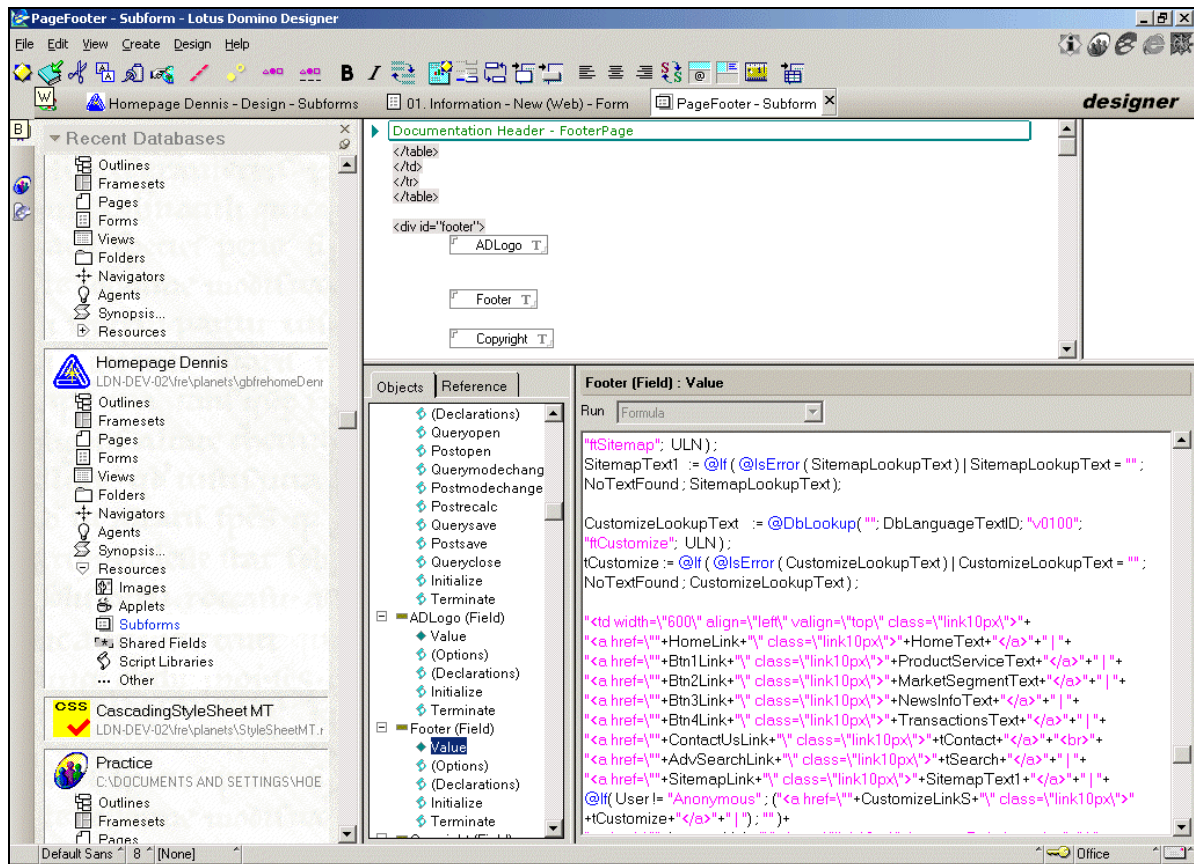
Div#copyright a:visited {
    color: #3366cc;}
```

Ook dit fragment is verschillend ten opzichte van de style sheet, die bij de bespreking van het systeem concept aan bod gekomen is. Dit is ook een gevolg van het feit, dat de *font* elementen vervangen zijn door *DIV* elementen.

Bovenstaande stappen, het wijzigen van de HTML en het opstellen van de style sheet voor het betreffende blok, is voor alle blokken en elke lay-outstructuur uitgevoerd. Op basis van de geleerde technieken uit de zelfstudie voor CSS is op dezelfde manier per blok een style sheet voor het printen ontwikkeld. Hiervoor waren geen aanpassingen meer nodig in de HTML; een extra style sheet voor de presentatie op papier is voldoende om de print lay-out te realiseren. Nadat de HTML voor alle blokken en lay-outs gewijzigd is en hiervoor de juiste stijlen opgesteld zijn in de style sheets, kunnen de losse documenten samengevoegd worden, zodat complete pagina's ontstaan, die samenwerken met de complete samengevoegde CSS-documenten. Op dit moment waren dus de volgende lokaal opgeslagen documenten aanwezig;

- Drie complete aangepaste HTML pagina's (samengevoegde HTML-documenten)
- Eén nieuwe style sheet voor de presentatie op het scherm (samengevoegde CSS-documenten voor het scherm)
- Eén nieuwe style sheet voor de presentatie op papier (samengevoegde CSS-documenten voor papier)

De volgende en laatste stap bij het ontwikkelen van de CSS is het doorvoeren van de gemaakte wijzigingen in de HTML op de ontwikkelserver. Zoals al eerder gezegd is, is de HTML voor een bepaalde pagina niet terug te vinden als één document waarin alle code opgenomen is voor die betreffende pagina. De website draait in een Lotus Notes omgeving en als gevolg daarvan, is de code opgeslagen in verschillende documenten, welke weer opgeslagen zijn in de databases, die in gebruik zijn voor de website. In deze databases wordt gebruik gemaakt van verschillende *forms* en *subforms*, waarin de HTML ofwel hard gecodeerd is, ofwel door script gegenereerd wordt. Als laatste stap wordt deze code nu aangepast op basis van de documentatie, die opgesteld is bij het aanpassen van de HTML in de lokale pagina's. Op onderstaande afbeelding is één van de subforms te zien, waarop HTML wijzigingen gedaan zijn. Het is een screenshot uit het programma *Lotus Domino Designer*; het pakket waarmee ontwikkeld wordt voor Lotus Notes.



Het aanpassen van de HTML op de ontwikkelserver leverde een probleem op, omdat dit meer tijd kostte dan hiervoor ingepland was. Door de complexiteit van de verschillende forms en subforms heb ik er langer over gedaan om de wijzigingen door te voeren. De complexiteit is vooral een gevolg van het feit, dat de HTML code zo verspreid is, waardoor het overzicht wegvalt. Het was de bedoeling, dat de aanpassingen voor alle blokken en alle lay-outstructuren op de ontwikkelserver doorgevoerd zouden worden, zodat er van elke standaard lay-out een pagina op de ontwikkelserver zou staan.

De wijzigingen op de ontwikkelserver zijn bedoeld als voorbeeld voor de wijzigingen die in alle overige pagina's gedaan moeten gaan worden. Hier gaat het om de aanpassingen in pagina's die niet binnen dit project aan bod komen. Voor dit proces wordt door mij een script geschreven, waarin beschreven staat welke aanpassingen nodig zijn in de HTML en hoe deze uitgevoerd moeten worden in de Notes databases. Dit script is gebaseerd op de wijzigingen die ik in dit project in de pagina's met de standaard lay-out uitvoer. Aangezien het te veel tijd zou gaan kosten om de wijzigingen voor drie pagina's door te voeren (één pagina voor elke standaard lay-out), moest hiervoor een andere oplossing bedacht worden. Het probleem is besproken met de bedrijfsmentor en de conclusie van het gesprek was, dat het script ook te baseren is op de aanpassingen die gedaan zijn voor één pagina. Een groot deel van de pagina's is immers voor elke pagina hetzelfde, aangezien header, footer en sidebar voor elke pagina gelijk zijn.

Om tijd te besparen is besloten om slechts één aangepaste pagina in te voeren op de ontwikkelserver. Voor de volgende blokken is dan ook de HTML aangepast op de ontwikkelserver in de betreffende databases;

- Header
- Sidebar
- Footer
- Eén lay-outstructuur voor het content blok

Het script, dat geschreven wordt als leidraad voor de wijzigingen in alle overige pagina's, heb ik gebaseerd op de wijzigingen die gedaan zijn in bovenstaande delen. Het script is opgenomen in externe bijlage B.

#### 4.3 Tweede style sheets

Na het ontwikkelen van de eerste style sheets, was gepland om de tweede style sheets te ontwikkelen, ook voor zowel het scherm als voor het printen. De ontwikkeling van de tweede (verbeterde) style sheet richt zich vooral op het optimaliseren van het gebruik van CSS, met andere woorden, deze ontwikkeling richt zich op het toepassen van de *best practices* in het gebruik van CSS. De ontwikkeling van de tweede style sheets, zou gebaseerd zijn op een adviesrapport, dat ook door mij binnen dit project opgesteld wordt. In dit adviesrapport worden voorstellen voor wijzigingen in structuur en lay-out van de website gedaan, waardoor CSS optimaal toegepast wordt. Voor het rapport worden gedurende het hele traject aantekeningen bijgehouden. Het rapport wordt gebaseerd op de conclusies van de zelfstudie voor CSS en de ontwikkeling van de eerste style sheets.

Ik ben echter niet toegekomen aan de ontwikkeling van de tweede versie van de style sheets. Hiervoor is een aantal redenen aan te wijzen;

- De uitloop die ontstaan is tijdens de definitie fase; de definitie fase heeft langer geduurd als gevolg van de uitgebreide systeemanalyse. Ook is er bij het opstellen van de planning geen rekening gehouden met het maken van Engelse vertalingen van delen van de documentatie. Hierdoor is de definitiefase verder uitgelopen.
- Parallel aan het ontwikkelen van de eerste style sheets, werd ook gewerkt aan de ontwikkeling van het management systeem (dit zal in paragraaf 4.4 *Het management systeem* besproken worden). Ook dit kost meer tijd dan verwacht en dit zijn pilots met de hoogste prioriteit; hieraan moet dus voldoende aandacht besteed worden.
- Het wijzigen van de HTML op de ontwikkelserver kost meer tijd dan verwacht. Dit probleem is in de vorige paragraaf ook al aan bod geweest.
- Tijdens het ontwikkelen van de eerste style sheets wordt bekend, dat het bedrijf rond de kerstdagen een week extra gesloten zal zijn. Dit betekende een vakantie van twee weken, waarmee de beschikbare tijd voor het ontwikkelen dus een week korter werd.

Om de voortgang van het project niet in gevaar te laten komen, moesten er maatregelen genomen worden en aangezien het hier gaat om pilots met een lagere prioriteit, zijn dit de pilots die bij problemen als eerste herzien worden. Ik had het geluk, dat ik de problemen van te voren zag aankomen; de definitiefase was inmiddels al gedaan, eveneens als de wijzigingen van de HTML op de ontwikkelserver. Op dit moment was het dus al bekend, dat er minder tijd over was voor de rest van het traject. Ook het feit, dat de ontwikkeling van het management systeem langer ging duren, was al vroeg bekend. De reden hiervoor is, dat de ontwikkeling van het management systeem parallel gedaan werd aan de ontwikkeling van de eerste style sheets. Dat de ontwikkeling van de tweede style sheets in gedrang kwam, was dus al bekend toen ik nog aan de ontwikkeling van de eerste style sheets bezig was. Hierdoor heb ik de mogelijkheid gehad om veel van de *best practices* ook al in de eerste style sheets te verwerken. Zo is bijvoorbeeld de techniek, die gebruikt is voor de print lay-out van de pagina's volledig gebaseerd op deze *best practices*. Op deze manier heb ik geprobeerd om zoveel mogelijk *best practices* al in de eerste versie van de style sheets te verwerken. De verdere ontwikkeling van de tweede versie is niet meer uitgevoerd. Wel heb ik een kort adviesrapport geschreven, met daarin aanbevelingen voor de website. Deze aanbevelingen hebben betrekking op *best practices*, die eventueel op de website doorgevoerd zouden kunnen worden, maar welke nog niet in de eerste versie van de style sheets zijn opgenomen. Een overzicht van de aanbevelingen is opgenomen in externe bijlage B.



#### 4.4 Het management systeem

De ontwikkeling van het management systeem is in het pilotplan opgedeeld in twee pilots;

- Database voor het management systeem
- Management systeem voor de style sheets

Er is in eerste instantie dus een tweedeling gemaakt tussen het ontwikkelen van de database en de applicatie zelf. In de periode, dat het pilotplan opgesteld is en dus ook de verdeling van deze pilots gedaan is, was ik mij er nog niet voldoende van bewust hoe een database in Lotus Notes opgezet is. Bij het opstellen van het pilotplan ben ik uitgegaan van de gebruikelijke structuur bij het gebruik van een relationele database, waarbij een database wordt gekoppeld aan een applicatie. Een Notes database daarentegen bevat naast opgeslagen data ook de code van een eventuele applicatie. (zie interne bijlage 1) De applicatie is dus onderdeel van de database en als gevolg daarvan, is het onmogelijk om een applicatie los van de database te ontwikkelen. Er is daarom gekozen voor een andere aanpak. De volgende verdeling van pilots is opgesteld;

- Database met de technische functionaliteit van de applicatie
- Graphical User Interface (GUI) van de applicatie

Bij deze indeling zal de applicatie met het afronden van de eerste pilot af zijn met betrekking tot alle functionaliteit, die de applicatie bevat. Alle taken die uiteindelijk met het systeem uit te voeren zijn worden voor deze pilot ontwikkeld. De tweede pilot is een kleinere pilot, waarin de interface van de applicatie ontworpen en ontwikkeld wordt. Deze interface moet ervoor zorgen, dat de applicatie overzichtelijk en makkelijk in het gebruik wordt. Deze twee pilots worden serieel ontwikkeld, te beginnen met de technische functionaliteit.

#### 4.4.1 Technische functionaliteit

Zoals al eerder aangegeven is, is voor de ontwikkeling van het management systeem eerst begonnen met zelfstudie voor Lotus Domino Designer, het pakket waarmee ontwikkeld wordt voor Lotus Notes. Deze zelfstudie was nodig, omdat ik nog nooit eerder ontwikkeld heb met dit pakket. Ik heb vanuit Avery Dennison cursus materiaal gekregen, waarmee ik in staat ben geweest zelfstandig de basis technieken voor het ontwikkelen in Lotus Notes onder de knie te krijgen. Voor deze zelfstudie is een soortgelijke aanpak gebruikt als voor de zelfstudie van CSS. Tijdens de zelfstudie worden namelijk weer aantekeningen bijgehouden van de geleerde technieken, die van belang zijn voor dit project. Een aantal belangrijke onderwerpen die bestudeerd zijn, wordt hier opgesomd;

- Basistheorie met betrekking tot het gebruik van *forms*, *documents* en *views*
- Het automatiseren van taken met behulp van een *Agent*
- Het gebruik van *Lotus Script* en *Formula Language*
- Ontwerp en ontwikkeling van een *user interface*

De zelfstudie voor Lotus Notes was in eerste instantie eerder in het traject gepland maar door dit pas tijdens de ontwikkeling van het management systeem te doen, konden de geleerde technieken meteen in de praktijk gebracht worden, in de vorm van conceptdelen van het eindproduct. Op deze manier is gedurende de hele ontwikkeling gebruik gemaakt van *prototyping*, wat het mogelijk gemaakt heeft verschillende oplossingen voor onderdelen van de functionaliteit te proberen. Het eerste prototype bestond uit een form, waarop een selector gedeclareerd kan worden met enkele attributen. Op basis hiervan is veel functionaliteit van het management systeem ontwikkeld. De volgende punten waren hierbij belangrijke onderdelen;

- Het ontwikkelen van verschillende invoer validaties
- Het ontwikkelen van de *Agent*

Voor de verschillende invoervelden zijn invoervalidaties opgesteld, die ervoor moeten zorgen, dat een foutieve invoer correct afgehandeld wordt. De code van de invoervalidaties die voor het eerste prototype gemaakt zijn, kon voor veel velden die in latere prototypes toegevoegd zijn, hergebruikt worden.

Het ontwikkelen van het script voor de *Agent* heeft de meeste tijd gekost. De *Agent* is verantwoordelijk voor een belangrijk deel van het management systeem. Het opstellen van de *Agent* was een uitdaging, omdat ik nooit eerder met Domino Designer gewerkt had, en dus ook geen ervaring met *Lotus Script* had. In eerste instantie was ook een andere, makkelijkere oplossing bedacht tijdens de ontwikkeling van het eerste prototype. Bij de zelfstudie had ik namelijk geleerd, dat de inhoud van een *view* te exporteren is als tekst bestand. Aangezien een *view* met daarin de velden die de samengestelde stijldeclaraties bevatten vrij gemakkelijk te creëren is en de exportfunctie van deze view een standaard functie van Lotus Notes is, hoefde hiervoor geen enkel ingewikkeld script opgesteld te worden. Omdat er bij het testen van deze functionaliteit problemen ontstonden, op het moment, dat de inhoud van de geëxporteerde style sheets te groot van omvang werd, heb ik besloten om toch de *Agent* te programmeren, met de volgende functionaliteit.

De *Agent* is verantwoordelijk voor het samenstellen en exporteren van de style sheets. Er kunnen met behulp van de *Agent* 2 style sheets geëxporteerd worden;

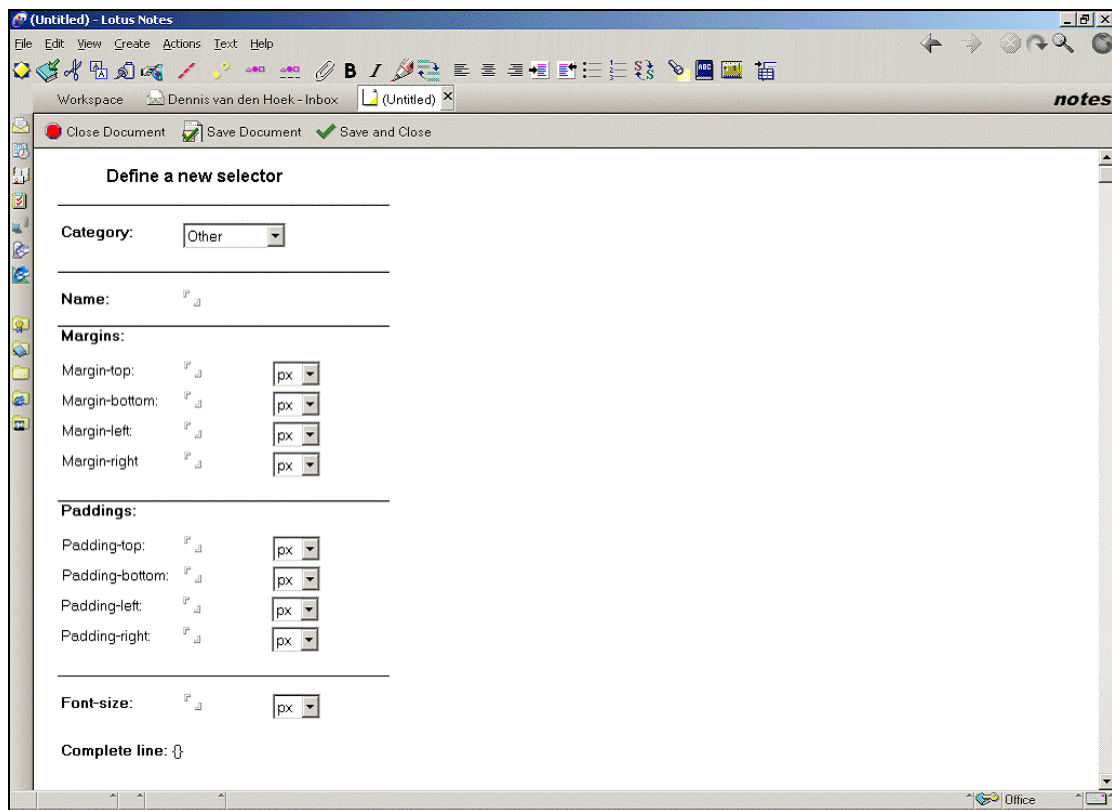
- Voor presentatie op het scherm
- Voor presentatie op papier

Het samenstellen en exporteren gebeurt met de volgende stappen;

- De juiste documenten worden geselecteerd uit de database op basis van het medium, *screen of print*.
- Elk document bevat een veld waarin alle opgegeven waarden voor de betreffende selector samengesteld zijn tot een complete stijldeclaratie. De *Agent* selecteert dit veld uit alle geselecteerde documenten.
- De *Agent* stelt alle stijldeclaraties samen tot een style sheet.
- De style sheet wordt geëxporteerd en als CSS bestand opgeslagen in een Notes database. In de nieuwe situatie wordt op de HTML pagina's naar dit bestand gelinkt.

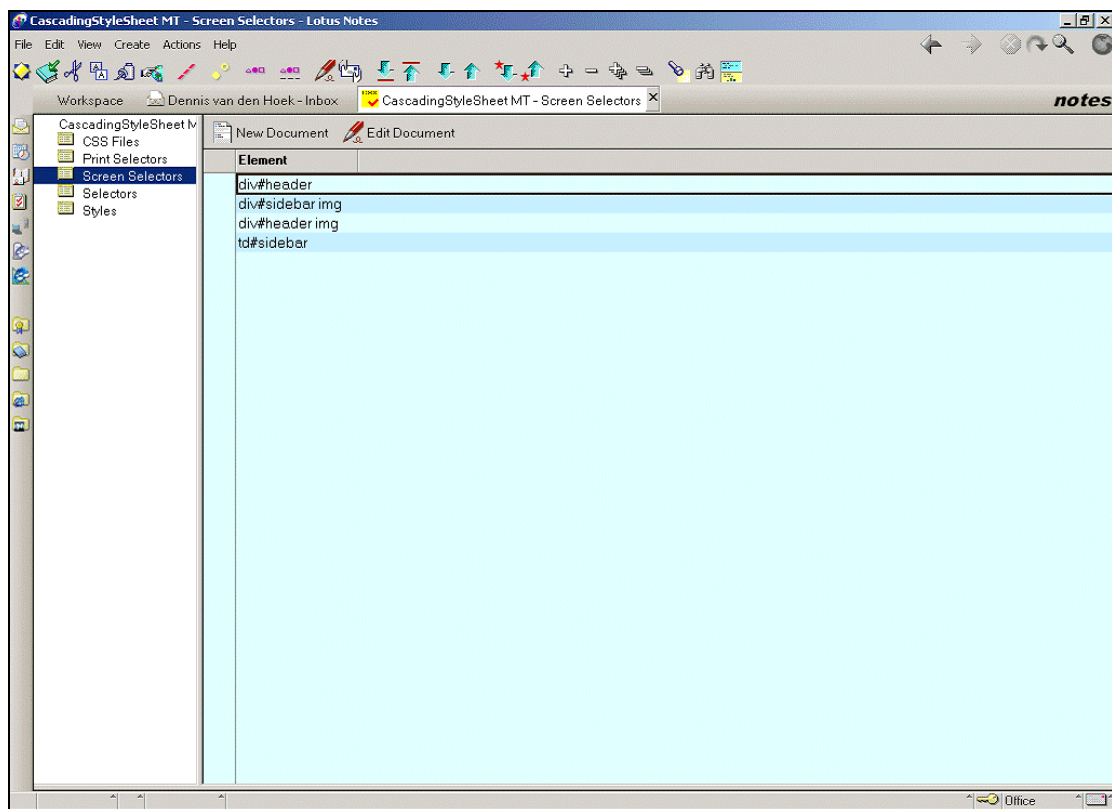
Het ontwikkelen van deze functionaliteit heeft dus meer tijd gekost dan gepland, temeer omdat ook eerst oplossingen geprobeerd zijn voor de zojuist genoemde methode, waarbij de style sheet als *view* geëxporteerd werd.

De volgende afbeelding is een screenshot van het eerste prototype



Bij het eerste prototype konden een naam, categorie en een aantal waarden voor *marges* en *paddings* opgegeven worden voor een selector.

Met een werkende *Agent* is het belangrijkste deel van de functionaliteit van het management systeem af. Het prototype wordt vanaf nu uitgebreid met meerdere velden, zodat elk attribuut, dat op de ontwikkelde style sheet voor de website voorkomt, ook aan te passen is met het management systeem. Veel van de invoervalidaties, die al eerder gemaakt zijn, kunnen voor deze nieuwe velden gebruikt worden. Voor enkele velden worden extra validaties opgesteld. Naast het uitbreiden van de velden voor de attributen, wordt het prototype ook uitgebreid met meerdere *views*. Deze *views* maken het mogelijk snel bepaalde selecties uit de style sheets op te zoeken en te bekijken. In de volgende afbeelding wordt één van de *views* getoond. In het geval van deze view, worden alle documenten voor de style sheet voor het scherm getoond, die op dat moment in de database aanwezig zijn.



#### 4.4.2 User interface

Met het afronden van de eerste pilot voor het management systeem zoals beschreven in de vorige paragraaf, was alle technische functionaliteit van het systeem afgerond. De tweede en laatste pilot bestond uit de ontwikkeling van een interface, waarmee de applicatie overzichtelijk gemaakt werd, zodat er makkelijk mee gewerkt kan worden. Om aan de systeemeisen met betrekking tot de interface te kunnen voldoen, zijn in eerste instantie schetsen gemaakt, die in het storyboard opgenomen zijn, welke besproken is bij het opstellen van het systeemconcept. (hoofdstuk 3.6) De volgende afbeelding toont één van de schetsen;

De interface, zoals deze opgesteld is voor het systeemconcept is uiteindelijk op twee belangrijke punten gewijzigd;

- De interface uit het systeem concept maakt geen gebruik van *views*. Zoals besproken is in de vorige paragraaf, wordt er in de ontwikkelde interface *wel* gebruik gemaakt van *views*
- Er wordt in de uiteindelijk ontwikkelde interface gebruik gemaakt van een tabstructuur om de informatie gespreid en overzichtelijk te houden.

Deze twee punten worden hieronder kort toegelicht.

### Het toevoegen van views in de applicatie

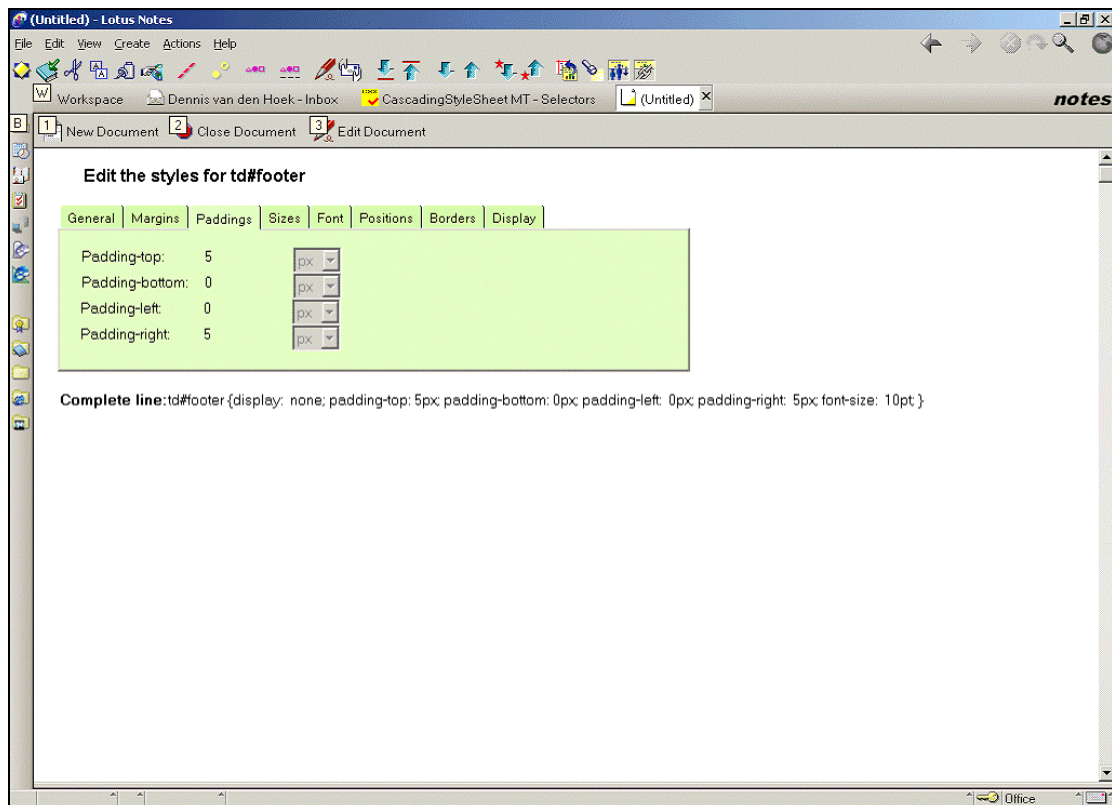
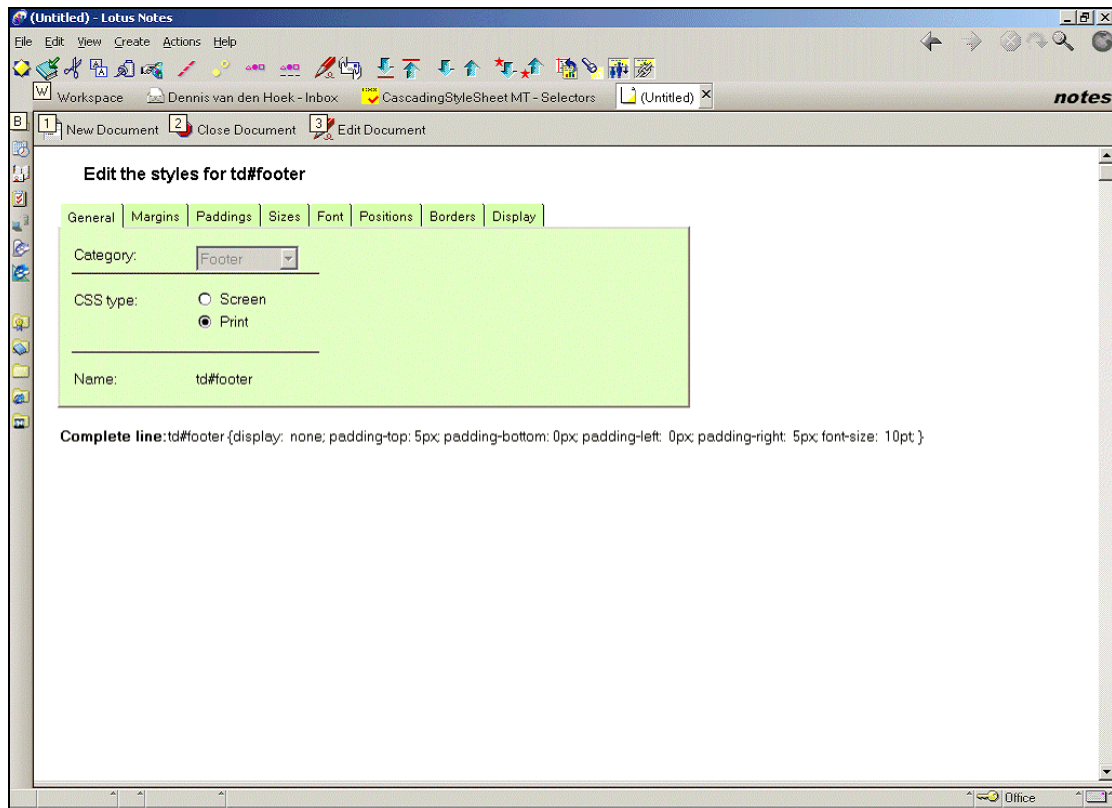
Bij het opstellen van het systeemconcept heb ik nog onvoldoende kennis van de verschillende aspecten van Lotus Notes, wat als gevolg heeft gehad, dat de interface, die voor het systeem concept ontworpen was, geen gebruik maakte van *views*. Een *view* is echter een zeer veel gebruikte functionaliteit in Notes en tijdens de zelfstudie kwam ik erachter hoe handig het gebruik van *views* kan zijn. Het doel van een *view* is het creëren van een georganiseerd overzicht van documenten. Om de volgende redenen heb ik toch gebruik gemaakt van *views* in het uiteindelijk ontwikkelde systeem;

- De *Lotus Notes client* is de omgeving waarbinnen het systeem zal werken. Deze omgeving is geoptimaliseerd voor het werken met *views*
- Door met de *views* te werken, heb ik de mogelijkheid verschillende overzichten van alle documenten te creëren. Op deze manier ontstaat er een overzicht van complete style sheets. In het ontwerp voor het systeemconcept konden slechts de eigenschappen per selector bekeken of bewerkt worden.
- De *views* maken het mogelijk om snel een bepaald HTML element op te zoeken en hiervoor de stijlenmerken aan te passen. Binnen de verschillende *views* worden de documenten namelijk op verschillende manieren gecategoriseerd.

### Het toevoegen van de tabstructuur binnen een document

Uit de systeem eisen kwam voort, dat de informatie, die op het scherm getoond wordt, zoveel mogelijk gespreid moet zijn, zodat het overzichtelijk blijft. Bij het ontwerp van de interface uit het systeem concept stonden alle attributen voor een HTML element op hetzelfde scherm. In de schetsen voor het systeemconcept zijn voor elke selector slechts vier attributen in te vullen maar in werkelijkheid zouden dit er natuurlijk veel meer worden, wat uiteindelijk een onoverzichtelijk resultaat zou hebben. Hiervoor is een tabstructuur ontworpen, waarbij niet alle attributen voor een HTML element op hetzelfde scherm getoond worden. In verschillende tabs worden attributen met een soortgelijke betekenis gegroepeerd. Op deze manier blijft het geheel overzichtelijk en zijn toch alle attributen direct bereikbaar om aan te passen.

De afbeeldingen op de volgende bladzijde laten de tabstructuur zien, zoals deze is ontwikkeld voor de uiteindelijke interface van het management systeem.



In het document uit bovenstaande voorbeelden zijn attributen gedeclareerd voor de selector *td#footer*. Op het eerste screenshot kunnen algemene attributen opgegeven worden, zoals de naam voor de selector. In het tweede screenshot kunnen waarden voor de *padding* opgegeven worden.



## 5. Invoeringsfase

Na het voltooien van de ontwikkelfase is de laatste fase van het traject aangebroken; de invoering. De invoeringsfase is een korte fase, mede doordat de style sheets niet binnen dit project ingevoerd worden. Daarnaast heeft het invoeren van het management systeem weinig tijd gekost. Voor beide delen zal in dit hoofdstuk de invoering toegelicht worden, voor zowel de ontwikkelde style sheets met de aangepaste HTML als het ontwikkelde management systeem.

### 5.1 Style sheets voor scherm en printen

De invoering van de style sheets en de bijbehorende aangepaste HTML pagina's, zou in eerste instantie door mij gedaan worden en dit zou tijdens dit project gebeuren. In de eerste weken van het project is echter al besloten, dat niet alle pagina's van de website binnen dit project aangepast worden, zoals aan bod gekomen is bij de bespreking van het Plan van Aanpak. Aangezien het voor de opdrachtgever geen nut heeft, om een aantal aangepaste pagina's in te voeren en daarbij de overige pagina's voorlopig zo te laten zoals ze zijn, is er voor gekozen om de style sheets pas volledig in te voeren, samen met de aangepaste HTML, op het moment, dat alle overige pagina's aangepast zijn. Het is al eerder aan bod gekomen, dat deze aanpassingen in een ander project gedaan zullen worden. Om dit proces zo goed mogelijk te laten verlopen, is na de afronding van de ontwikkeling van de style sheets en de bijbehorende aanpassingen aan de HTML het script geschreven, waarin beschreven wordt, welke aanpassingen nodig zijn in de HTML en hoe deze aanpassingen uitgevoerd moeten worden in de Notes databases.

### 5.2 Management systeem

De invoering van het management systeem heeft weinig tijd gekost. De invoering van het management systeem bestaat uit twee stappen;

- Het 'vullen' van de applicatie
- Het kopiëren van de applicatie naar de juiste server.

Met het 'vullen' van de applicatie wordt bedoeld, dat de juiste documenten ingevoerd worden; alle stijldeclaraties die op de ontwikkelde style sheets voorkomen worden ook in het systeem gezet, zodat het systeem nu de nieuwe style sheet kan genereren. Op deze manier is het systeem gevuld met verschillende documenten; voor elke selector een apart document, met daarop de attributen en hun waarden voor die selector. Het management systeem kan nu dan ook gebruikt gaan worden bij het aanpassen van de overige HTML pagina's. Wanneer het nodig is declaraties op de style sheet te veranderen, kan dit gedaan worden met behulp van het management systeem.

De tweede stap bij de invoering van het management systeem, bestaat uit het kopiëren van de applicatie naar de juiste server. Dit is meteen ook de laatste stap bij de invoering van het systeem. Doordat het systeem zal draaien in een Lotus Notes omgeving en tevens ontwikkeld is in zo'n zelfde omgeving, wordt het systeem vrij gemakkelijk en snel ingevoerd. De server die gebruikt is bij de ontwikkeling van het systeem is een exacte kopie van de server waarop het systeem uiteindelijk zal draaien; hiervoor hoeven dus verder geen aanpassingen gedaan te worden aan het systeem.

## 6. Evaluatie

In dit hoofdstuk zal een evaluatie van het project gegeven worden. Aan de hand van een proces- en productevaluatie zal ik reflecteren op de afstudeeropdracht.

### 6.1 Procesevaluatie

Ik heb het project over het algemeen als een uitgebreid project ervaren, waarbij ik vooral in het begin twijfelde of ik dit op een juiste manier en binnen de gestelde tijd af zou krijgen. Het is voor het eerst, dat ik een project van dergelijke omvang zelfstandig heb uitgevoerd. Uiteraard heb ik hierbij van verschillende mensen steun ontvangen. Deze steun kwam zowel vanuit de organisatie, als vanuit school en natuurlijk mijn thuisomgeving. Hierdoor heb ik het project wel op tijd en in goede staat tot een einde kunnen brengen, waardoor ik over het algemeen zeer tevreden ben over het resultaat van het project. Uiteraard zijn hierbij ook de nodige problemen geweest, bijvoorbeeld problemen op het gebied van ontwikkelen, maar ook situaties die ervoor gezorgd hebben dat de opzet van mijn opdracht er anders uit kwam te zien. De belangrijkste processen uit het project zullen hier toegelicht worden.

#### Opdracht

Het vinden van een geschikte afstudeeropdracht heeft voor mij even geduurd. Ik zou in eerste instantie al eerder begonnen zijn met de afstudeerstage, maar als gevolg van het feit, dat de eerste opdrachtschrijving, die ik heb ingeleverd ter goedkeuring, een aantal malen afgewezen is, was ik door tijdgebrek genooddaakt om een periode later te beginnen met afstuderen. De eerste opdrachtschrijving die ik heb ingeleverd, was ook voor Avery Dennison. Toen deze een aantal maal afgekeurd was, heb ik samen met de bedrijfsmentor een nieuwe opdracht in elkaar gezet, gebaseerd op de eerste opdracht. In feite is de eerste opdracht uitgebreid, waarbij de ontwikkeling van het management systeem voor de style sheets de belangrijkste toevoeging is geweest. Ik was zeer tevreden met deze opdracht en deze is ook uiteindelijk goedgekeurd. Heel belangrijk vond ik het grote voordeel, dat het opgeleverde werk ook daadwerkelijk gebruikt zal gaan worden in de bedrijfsvoering van de afdeling waar ik werkzaam ben geweest. De style sheets zullen dan niet direct ingevoerd worden, het management systeem daarentegen zal wel al gebruikt worden.

De opdracht is alleen in het begin van het project iets aangepast ten opzichte van het origineel; bij het bespreken van het Plan van Aanpak is al aan bod gekomen, dat het in eerste instantie de bedoeling was, dat ik voor alle pagina's de HTML-code zou aanpassen. Aangezien dit te veel tijd zou gaan kosten, is besloten om dit anders aan te pakken en dit alleen te doen voor een aantal pagina's met een standaard lay-outstructuur. Omdat het probleem zo vroeg in het trajectesignaleerd is, kon tijdig een oplossing bedacht worden.

### Methodiek

Het feit dat binnen de opleiding VIA zoveel aandacht besteed wordt aan de ontwikkelmethodiek IAD, is een grote motivatie geweest om binnen dit project ook IAD toe te passen. In de loop van het project is echter wel gebleken, dat deze opdracht zich minder goed leende voor deze methodiek. Bij de ontwikkelde style sheets is bijvoorbeeld weinig sprake van een itererend proces en ook was het moeilijk dit deel van de opdracht in te delen in verschillende pilots. Voor de ontwikkeling van het management systeem daarentegen was IAD beter toe te passen, aangezien hier het systeem duidelijk in verschillende stappen (iteraties) naar het eindproduct heeft kunnen evolueren.

IAD kent doorgaans veel documentatie; veel hiervan is gericht op het werken aan een project met een volledige projectgroep, die bestaat uit meerdere mensen. Aangezien ik alleen aan dit project gewerkt heb, is het niet nodig geweest om volledig aan de IAD-eisen met betrekking tot de documentatie te voldoen. Niet alle onderdelen van IAD zijn dus volledig volgens de IAD standaarden uitgevoerd, maar IAD is wel door het hele project als richtlijn gebruikt. Er is steeds aan de bedrijfsmentor voorgesteld, welke onderdelen van IAD uitgevoerd zouden kunnen worden.

### Planning

Tijdens het gehele traject heb ik mij redelijk aan de initiële planning kunnen houden. De definitiefase is uitgelopen door een inschattingfout; de systeemanalyse heeft langer geduurd dan gepland en er was geen rekening gehouden met het maken van de engelse vertalingen van de systeemdokumentatie. De ontwikkelfase is ook uitgelopen, als gevolg van het oponthoud dat ik heb gehad bij het invoeren van de aangepaste HTML pagina's op de ontwikkelserver en de ontwikkeling van de agent van het management systeem. Door deze uitloop heb ik de tweede style sheets uiteindelijk niet volledig als pilots kunnen ontwikkelen.

Er werd regelmatig teruggekoppeld naar de planning, zodat eventuele problemen tijdig bekend werden. Zo was het dus al vroeg bekend, dat de ontwikkeling van de tweede style sheets in het gedrang zou komen. Hierdoor heb ik op tijd een oplossing kunnen bedenken om dit probleem zo goed mogelijk op te vangen.

## Zelfstudie

Het is al een aantal maal aan bod gekomen, dat een groot deel van dit project bestond uit zelfstudie voor zowel CSS als het ontwikkelen in Lotus Notes. Het feit dat ik aan een project begonnen ben, waarvoor het nog nodig was studie te doen, bracht natuurlijk een risico met zich mee; wanneer ik niet in staat geweest zou zijn om mij de theorie in korte tijd eigen te maken, zou de voortgang van het project ofwel door tijdgebrek ofwel door gebrek aan kennis in het gedrang kunnen komen. Doordat ik voldoende steun vanuit de organisatie heb gekregen in de vorm van goed studiemateriaal en uitleg wanneer ik iets te vragen had, ben ik toch in staat geweest om bij beide studies in korte tijd zeer veel te leren.

## Analyse

De systeemanalyse is van groot belang geweest tijdens dit project. Tijdens de systeemanalyse zijn namelijk de document trees ontwikkeld. Deze document trees zijn gebruikt voor de verdere ontwikkeling van de CSS. De systeemanalyse heeft langer geduurd dan verwacht, vanwege de complexiteit en de omvang van de verschillende pagina's. Toch was het van groot belang voldoende aandacht te besteden aan de systeemanalyse, omdat de document trees zo belangrijk waren voor dit project.

Bij de analyse van de website is weinig gekeken naar de code op de verschillende forms en subforms; de analyse is namelijk van 'buitenaf' gedaan, zoals uitgelegd is bij de bespreking van de systeemanalyse in dit verslag. Achteraf gezien had ik bij deze analyse meer aandacht moeten besteden aan de code op de forms en subforms. Doordat ik dit niet gedaan heb, ben ik later tegen problemen opgelopen bij het doorvoeren van de HTML wijzigingen op de ontwikkelserver. De tijd, die nodig zou zijn voor het doorvoeren van de HTML wijzigingen op de ontwikkelserver, is daardoor verkeerd ingeschat.

## Ontwikkeling

Bij het ontwikkelen van de verschillende pilots heb ik weinig problemen ondervonden, naast het feit dat de meeste ontwikkelingen langer hebben geduurd dan verwacht. Bij het ontwikkelen van de CSS kwam dit vooral door de vertraging die ik opliep bij het doorvoeren van de wijzigingen in de HTML op de ontwikkelserver. Hierbij viel de complexiteit van de verschillende databases, forms en subforms tegen, waardoor het geheel langer geduurd heeft. Bij de ontwikkeling van het management systeem is vooral de zelfstudie een stuk sneller verlopen dan verwacht. Hierdoor kon al vroeg begonnen worden met het ontwikkelen van prototypes van het management systeem. De ontwikkeling van de *Agent* die ervoor zorgt, dat de style sheets samengesteld en geëxporteerd worden, heeft daarentegen weer wat langer geduurd, temeer omdat ik eerst een andere oplossing in plaats van de *Agent* geprobeerd heb. Deze oplossing bleek uiteindelijk voor problemen te zorgen en daarom is ervoor gekozen om de *Agent* alsnog te ontwikkelen.

### Ondersteuning

Ondersteuning heb ik zowel vanuit school gekregen als vanuit het bedrijf. Het aantal contactmomenten met school is eigenlijk zeer beperkt; er vindt één contactmoment plaats ongeveer halverwege het traject, waarbij de examinatoren langskomen bij het bedrijf. Verder hebben er twee contactmomenten op school plaatsgevonden, waarbij het concept voor het eindverslag besproken is. Deze contactmomenten zijn zeer nuttig geweest, vooral de gesprekken over het concept voor het eindverslag aan het eind van de periode.

De ondersteuning vanuit het bedrijf heb ik als zeer positief ervaren. Allereerst ben ik heel tevreden over de begeleiding die ik heb gekregen van de bedrijfsmentor. Regelmatig vond er overleg plaats, waarbij de voortgang van het project besproken werd. Ook zie ik het al als een groot pluspunt, dat ik de mogelijkheid heb gekregen om mij te ontwikkelen door middel van zelfstudie en dat ik hierbij ook het juiste studiemateriaal tot mijn beschikking heb gekregen. Verder zijn verschillende ontwikkelaars van de afdeling ASD bereid geweest om mij te helpen op het gebied van de ontwikkelingen in Lotus Notes, wanneer ik daar behoefte aan had.

### Leerervaring

Het project is voor mij op verschillende gebieden een leerervaring geweest. Ten eerste is dit voor mij de eerste keer geweest, dat ik heb ontwikkeld in Lotus Notes. In een korte tijd ben ik in staat geweest om mij de basis theorie van Lotus Notes eigen te maken, en hiermee een werkend systeem te ontwikkelen. Verder heb ik de ruimte gekregen om zelfstandig aan het project te werken. Hierbij had ik vooral in het begin een beetje moeite met het wennen aan de engelse voertaal; alle communicatie binnen het bedrijf gebeurt in het Engels. Ook dit is een goeie en vooral ook leuke leerervaring geweest.

## 6.2 Productevaluatie

Over de verschillende eindproducten ben ik zeer tevreden. Hoewel ik niet alles af heb kunnen ronden zoals het in eerste instantie overeengekomen was, ben ik tevreden met de gekozen oplossing hiervoor en het uiteindelijke resultaat. De verschillende producten zullen in deze paragraaf toegelicht worden.

### Documentatie

Een redelijk deel van de tijd van dit project is besteed aan het schrijven en vertalen van de documentatie, waarbij vooral de definitiestudie van groot belang was. Dit was belangrijk, omdat de systeemanalyse tijdens deze fase heeft plaatsgevonden. Op basis van die systeemanalyse zijn de document trees ontwikkeld, welke van groot belang zijn geweest binnen dit project. Om deze reden was het dus noodzakelijk voldoende aandacht te besteden aan de documentatie.

Voor de gemaakte documentatie is steeds met de bedrijfsmentor overeengekomen welke delen hiervan vertaald zouden worden in het Engels, zodat deze delen als systeemdokumentatie opgeleverd kon worden aan de opdrachtgever.

Het schrijven van het eindverslag heeft mij niet zoveel problemen opgeleverd, naast het feit dat het veel meer tijd gekost heeft dan verwacht. Het eerste concept voor het verslag, dat ik ingeleverd heb, is bijna volledig herschreven. De conclusie van de evaluatie van het concept was namelijk, dat het proces, dat ik doorlopen heb tijdens dit project, niet voldoende duidelijk werd. Nadat ik dit besproken heb met de examinatoren, had ik een beter beeld van wat er van mij verwacht werd met betrekking tot het eindverslag. Vanaf dat moment is het schrijven van het verslag een stuk soepeler verlopen.

### Style sheets

Een deel van het project bestond uit de ontwikkeling van style sheets voor zowel de presentatie van HTML elementen op het scherm, als de presentatie van deze elementen op papier. Het is in het verslag al een aantal keer aan bod geweest, dat na het afronden van de ontwikkeling van de eerste style sheet, een tweede set style sheets ontwikkeld zou worden. Deze style sheets zouden gericht zijn op het in praktijk brengen van verschillende *best practices* voor het gebruik van CSS. De ontwikkeling van de tweede style sheets heeft echter nooit plaatsgevonden, zoals uit dit verslag al gebleken is. De gekozen oplossing hiervoor heeft toch tot een bevredigend resultaat geleid. Door het verwerken van *best practices* in de eerste style sheets, is veel functionaliteit toch gerealiseerd. De overige *best practices* zijn verwerkt in een adviesrapport in de vorm van aanbevelingen, zodat er toch een overzicht ontstaan is van eventuele wijzigingen die in de toekomst nog gedaan zouden kunnen worden aan de website, om zo het gebruik van CSS nog verder te optimaliseren.

De tweede style sheets zijn dus niet meer ontwikkeld, maar door de bovengenoemde oplossing, is een groot deel van de functionaliteit die eigenlijk in de tweede style sheets zou voorkomen, reeds ontwikkeld voor de eerste style sheet. Op deze manier is een groot deel van het doel van de tweede style sheets toch gedekt. Deze eerste style sheets zijn dus wel helemaal ontwikkeld. Ook zijn hierbij alle HTML aanpassingen gedaan in de lokale HTML pagina's. Deze aanpassingen zijn voor één standaard pagina doorgevoerd in de database op de ontwikkelserver. Op basis van deze aanpassingen is vervolgens het script geschreven, dat gebruikt zal worden bij het aanpassen van de overige HTML pagina's.

### Management systeem

Over het management systeem ben ik zeer tevreden; de applicatie bevat alle functionaliteit die het zou moeten bevatten. Alle eisen die aan het systeem gesteld zijn, zijn er dus in verwerkt. Verder is het systeem bijna volledig ontwikkeld volgens het concept dat tijdens de definitiefase opgesteld is. (de interface wijkt alleen af van het concept)

Tijdens de beginperiode van het project zag ik vooral op tegen de ontwikkeling van het management systeem, omdat het ontwikkeld zou gaan worden in Lotus Notes, wat toen nog onbekend terrein was voor mij. Doordat ik de juiste begeleiding en studiemateriaal kreeg, ben ik toch in staat geweest in een vrij korte tijd een goed werkende applicatie te ontwikkelen. Het inplannen van de zelfstudie op het juiste moment is hierbij van belang geweest; door dit op het juiste moment in te plannen heb ik de mogelijkheid gehad om de geleerde technieken meteen in de praktijk te brengen in de vorm van conceptdelen van het eindproduct. Op deze manier heb ik dus gebruik kunnen maken van prototyping, waarbij het management systeem van prototype naar eindproduct evolueert.

## **Literatuurlijst**

### **Boeken**

Eric Meyer on CSS, Mastering the Language of Web Design  
E. Meyer, 2002, ISBN 0-73571-245-X

Working in Domino Databases/Domino Designer Fundamentals  
Lotus Authorized Education, 1999

Lotus Domino R5.0: A Developer's Handbook  
(IBM Redbooks) F. Collins, D. Morrison, S.P. Nielsen, S. Serpola, R. Strobl,  
1999, ISBN 0738414123

IAD, het evolutionair ontwikkelen van informatiesystemen  
R.J.H. Tolido, 1996, ISBN 90-395-0401-6

Guide, Graphical User Interface Design and Evaluation, a practical process  
David Redmond-Pyle & Alan Moore, 1995, ISBN 0-13-315193-X

Enquêteren- Het opstellen en gebruiken van vragenlijsten  
E.P.W.A. Jansen & Th. H. Joostens, 1998, ISBN 90-01-05103-0

### **Websites**

IBM DeveloperWorks: [www-136.ibm.com/developerworks/lotus](http://www-136.ibm.com/developerworks/lotus)

IBM Forum: [www-10.lotus.com/ldd/café.nsf](http://www-10.lotus.com/ldd/café.nsf)

World Wide Web Consortium: [www.w3c.org](http://www.w3c.org)

A List Apart: [www.alistapart.com](http://www.alistapart.com)

Eric Meyer on CSS: [www.ericmeyeroncss.com](http://www.ericmeyeroncss.com)

MeyerWeb.com: [www.meyerweb.com](http://www.meyerweb.com)



---

## **Interne bijlage 1- Theorie Lotus Notes**

Volledig gebruik van CSS op europe.fasson.com

Dennis van den Hoek  
Avery Dennison/RME - ASD

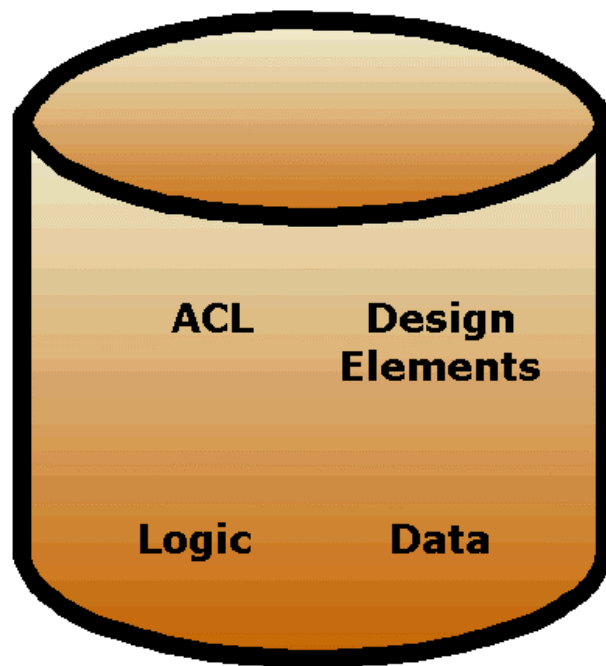
---

## Theorie Lotus Notes

### Over Notes databases

Een database in Lotus Notes is het best te omschrijven als een *container* voor een volledige applicatie. Binnen een Notes database zijn zowel het *application design* als de *data* opgeslagen in een bestand, genaamd *Notes Storage Facility (NSF)*.

Onderstaande afbeelding illustreert een database in Lotus Notes met de componenten die de database bevat.



Notes Storage Facility

Een Notes Database bevat de volgende componenten;

Component	Functie	Voorbeeld
Acces Control List	De ACL definieert de gebruikers van de applicatie en wat de bevoegdheden van deze gebruikers zijn binnen de applicatie	Elke gebruiker mag data lezen maar alleen een bepaalde groep mag ook data aanmaken.
Design Elements	Design elements zijn de bouwstenen van een applicatie. De design elements worden gebruikt voor de interactie van de gebruiker met de data.	Een webpagina bevat tekst, afbeeldingen en links. Dit zijn allemaal design elements.
Logic	<p>Aan de design elements wordt code gekoppeld voor;</p> <ul style="list-style-type: none"> <li>▪ Het berekenen, aanpassen of benaderen van waardes uit de database.</li> <li>▪ Het automatiseren van taken.</li> </ul>	Het opslaan van data wanneer de gebruiker op een button klikt.
Data	Data is de informatie waarmee de applicatie werkt. Gebruikers creëren, wijzigen en lezen data met behulp van de design elements.	De NAW gegevens van een klant uit een database.

Een Notes database wordt benaderd vanuit de Lotus Notes Client, of vanaf het internet.

## Over Notes *Documents* en *Forms*

Data in een Notes database is opgeslagen in een *Document*. Een document kan allerlei verschillende data bevatten, zoals tekst, afbeeldingen, animaties, etc. Een document is dus een onderdeel van de applicatie, dat data opslaat. Deze data kan door de gebruiker ingevoerd worden, berekend worden door bepaalde formules of geïmporteerd worden uit een andere applicatie.

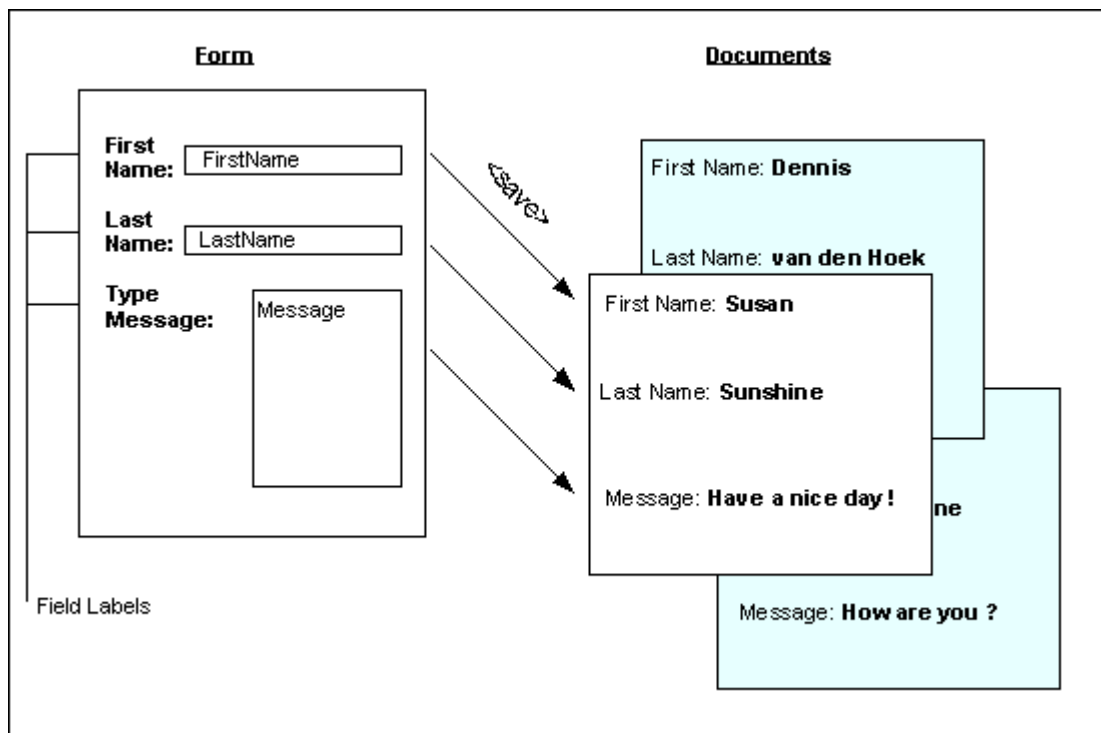
Normaal gesproken is de data, die in één document voorkomt, aan elkaar gerelateerd. Een document uit een database met adresgegevens zal bijvoorbeeld de adresgegevens van één bepaalde persoon of instelling bevatten.

Een gebruiker kan de data van een document invoeren, bewerken of lezen via het design element *Form* (of *SubForm*). Een Form is eigenlijk een template, dat door de ontwikkelaar van de applicatie gemaakt wordt. De template biedt een structuur en opmaak voor de data, die in het document wordt opgeslagen.

Wanneer een gebruiker bijvoorbeeld een nieuw *Document* wil invoeren, zal dit gedaan worden met behulp van het *Form*, dat bij dat type *Document* hoort. Op een *Form* kan verschillende statische informatie voorkomen maar belangrijker zijn de *Fields*. *Fields* zijn genaamde velden waarop de gebruiker de data invoert. Wanneer het document opgeslagen wordt, wordt alle data uit de *Fields* opgeslagen. Bijvoorbeeld; de achternaam van een persoon zal in een Field genaamd LastName ingevoerd worden. (De naam van het *Field* is willekeurig, de ontwikkelaar van de applicatie is hierin vrij)

Voor dezelfde soort documenten wordt steeds hetzelfde Form gebruikt. Elke keer als er een nieuwe set met data met behulp van het Form ingevoerd wordt en opgeslagen, wordt er een nieuw document gecreëerd, waarin de zojuist ingevoerde data opgeslagen wordt.

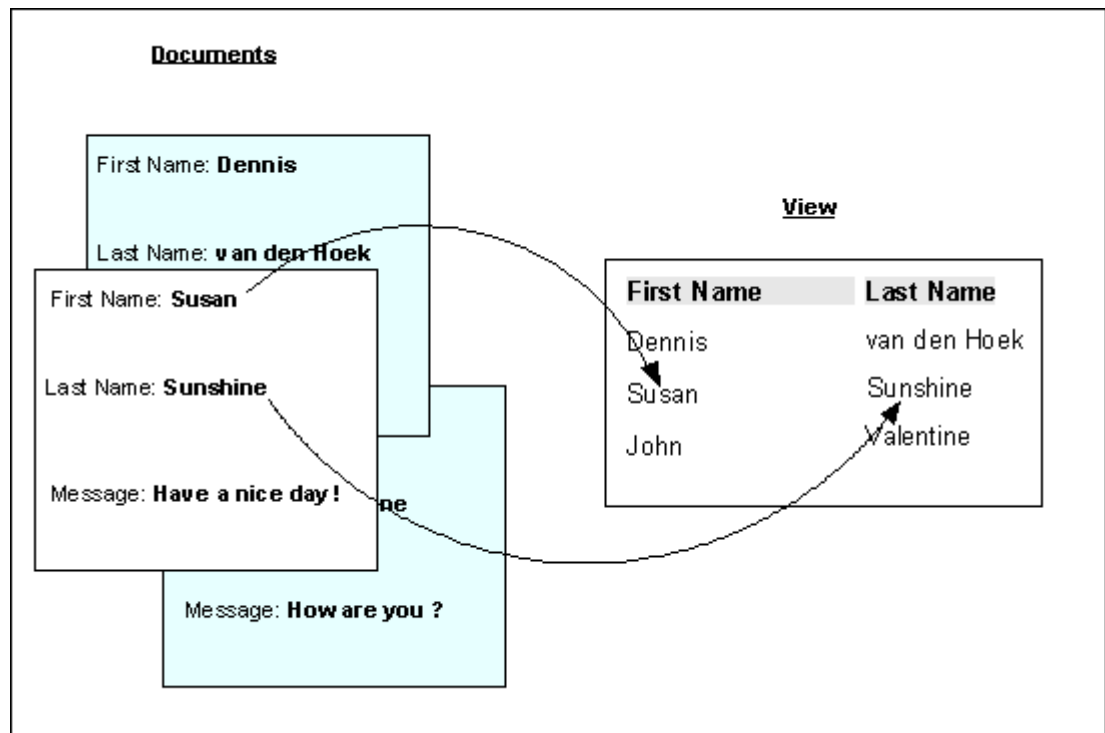
Het volgende diagram toont een Form met drie Fields, elk met een label.  
Het diagram toont ook drie bijbehorende Documents.



## Over Views

Lotus Notes maakt gebruik van een speciaal kenmerk; *Views*. Een *View* toont een georganiseerde lijst van documenten uit een database. Elk *Document*, of set van *Documents*, kan geïdentificeerd worden, aan de hand van de waarde van één of meer *Fields*. Op deze manier biedt de applicatie overzichten van *Documents*, zodat de gebruiker makkelijk de informatie kan vinden, die hij of zij zoekt. Elke database heeft op z'n minst één *View*, in de meeste gevallen zullen dit er echter meerdere zijn.

Het volgende diagram laat de relatie zien tussen Documents en Views.



Elke rij in de *View* vertegenwoordigt één *Document* in de database. Elke kolom laat informatie zien uit het *Document* van die rij. Een *View* kan alle *Documents* uit een database bevatten; het kan echter ook een selectie zijn, gebaseerd op bepaalde selectiecriteria. Vanuit een *View* kunnen *Documents* geopend worden.

## Over Notes Formula Language en Script

Lotus Notes maakt gebruik van *Formula Language*. Hierbij wordt gewerkt met standaard formules, die gebruikt kunnen worden bij het uitvoeren van 'simpele', veel voorkomende taken, zoals het opslaan van een *Document*. Een formule kan opgebouwd zijn uit verschillende componenten, en op deze manier verschillende elementen combineren. Zo kunnen veel verschillende functies geprogrammeerd worden, variërend van simpel tot zeer complex, welke gebruikt kunnen worden op de Forms en Views, om zo bijvoorbeeld interactie met de gebruiker te creëren.

Componenten van een formule zijn;

- Variabelen
- Constanten
- Operatoren
- Keywords
- @Functions

Naast de *Formula Language* kan bij het ontwikkelen in Lotus Notes ook gebruik gemaakt worden van *LotusScript*. Dit is een zeer uitgebreide script taal, met alle eigenschappen van een moderne object georiënteerde script taal. Voor meer complexe programmatuur kan *LotusScript* gebruikt worden in plaats van de *Formula Language*.

## Over Notes Agents

Een agent is vergelijkbaar met een macro en kan gezien worden als een stand-alone applicatie, die een specifieke taak uitvoert in een of meer databases. Met een *Agent* kunnen bijvoorbeeld *Documents* gevuld worden of mail berichten verstuurd worden. Ook kunnen met een *Agent* meer complexe acties geautomatiseerd worden, bijvoorbeeld de interactie met andere software. Een *Agent* kan zowel door de gebruiker gebruikt worden, als door de applicatie zelf. In het eerste geval start de gebruiker de *Agent*, in het tweede geval draait de *Agent* op de achtergrond en wordt deze vanuit de applicatie opgestart. Er kan gebruik gemaakt worden van een aantal standaard *Agents* in Notes. Een *Agent* kan ook naar eigen inzicht ontwikkeld worden; waarmee zeer complexe en krachtige software ontwikkeld kan worden.

## Over Notes en europe.fasson.com

Voor de website europe.fasson.com is de HTML code opgeslagen in verschillende Notes Databases. De code is terug te vinden op verschillende Forms en SubForms verspreid over verschillende databases. Sommige code staat hard gecodeerd op deze Forms, andere code wordt gegenereerd door script of formules.

---

## **Bijlage A – Definitiestudie**

Volledig gebruik van CSS op europe.fasson.com

Dennis van den Hoek  
Avery Dennison/RME – ASD  
08-01-04

---



## Inhoudsopgave

<b>1 Inleiding.....</b>	<b>91</b>
<b>2 Plan van aanpak.....</b>	<b>92</b>
2.1 Inleiding .....	92
2.2 Achtergrondinformatie.....	93
2.2.1 Over Avery Dennison .....	93
2.2.2 Over RME .....	93
2.3 Probleemstelling .....	94
2.4 Doelstelling .....	95
2.5 Uitgangssituatie.....	97
2.6 Resultaat .....	97
2.7 Randvoorwaarden .....	98
2.7.1 Voorwaarden vanuit opleiding en organisatie .....	98
2.7.2 Voorwaarden aan opdracht.....	98
2.8 Methoden en technieken .....	99
2.8.1 Algemeen.....	99
2.8.2 Ontwikkelomgeving .....	100
2.9 Planning en activiteiten.....	101
2.9.1 De globale fasering.....	101
2.9.2 De globale planning .....	101
2.10 Beheersaspecten.....	105
2.10.1 Beheersen van de te maken kosten .....	105
2.10.2 Beheersen van de organisatie .....	105
2.10.3 Beheersen van tijd.....	106
2.10.4 Beheersen van informatie.....	106
2.10.5 Beheersen van kwaliteit .....	106
2.11 Risicofactoren .....	107
<b>3 Resultaten systeemanalyse.....</b>	<b>108</b>
3.1 Inleiding .....	108
3.2 Beschrijving website.....	109
3.2.1 Beschrijving homepage .....	109
3.3 Structuren per blok .....	118
3.3.1 Layout-structuur header.....	118
3.3.2 Layout-structuur footer .....	119
3.3.3 Layout-structuur sidebar .....	120
3.3.4 Layout-structuur 1 content.....	121
3.3.5 Layout-structuur 2 content.....	122
3.3.6 Layout-structuur 3 content.....	123
3.4 Van layout structuur naar document tree.....	124
3.5 Beschrijving management systeem voor menu .....	130
3.5.1 Globale functionele omschrijving .....	130
3.5.2 Globale technische omschrijving.....	134
<b>4 Systeemeisen .....</b>	<b>136</b>
4.1 Doelgroepbeschrijving .....	136
4.2 Basis systeemeisen .....	137
4.3 Performance eisen .....	138
4.4 Interface eisen .....	139
4.5 Operationele eisen .....	140

<b>5</b>	<b>Systeemconcept .....</b>	<b>141</b>
5.1	Systeemconcept stylesheet .....	141
5.1.1	Systeemconcept stylesheet – een voorbeeld .....	141
5.1.2	Stylesheets voor printlayout .....	145
5.2	Systeemconcept management systeem .....	146
5.2.1	Globaal storyboard management systeem .....	147
5.3	Globale technische omschrijving .....	150
5.4	Globaal objectmodel .....	152
<b>6</b>	<b>Technische structuur .....</b>	<b>153</b>
6.1	Ontwikkelomgeving .....	153
6.2	Te gebruiken hard- en software .....	153
<b>7</b>	<b>Pilotplan .....</b>	<b>154</b>
7.1	Onderverdeling van de pilots .....	154
7.2	Pilots en iteraties .....	154
7.3	Beschrijving per pilot.....	155
7.3.1	Eerste stylesheet & aangepaste HTML-pagina's .....	155
7.3.2	Eerste stylesheet voor printen.....	156
7.3.3	Tweede stylesheet & aangepaste HTML-pagina's .....	157
7.3.4	Tweede stylesheet voor printen.....	158
7.3.5	Database voor management systeem .....	159
7.3.6	Management systeem voor stylesheets .....	160

Deze bijlage bevat de definitiestudie, die tijdens het project is opgesteld. Deze definitiestudie is (deels) vertaald in het Engels en in die vorm opgeleverd aan de opdrachtgever. Dit is de (complete) Nederlandse versie van de definitiestudie, die tijdens het project gebruikt is.

## **1 Inleiding**

Dit document is de definitiestudie voor de ontwikkeling van style sheets met een bijbehorend management systeem voor de website europe.fasson.com voor Avery Dennison, divisie Roll Materials Europe (RME). Als start van de definitiestudie is een systeem analyse gedaan; de resultaten van deze analyse zijn opgenomen in dit document (hoofdstuk 3). In dit document staat verder beschreven wat er ontwikkeld gaat worden en aan welke eisen het nieuwe systeem moet voldoen. Ook wordt er een indeling en beschrijving gegeven van de pilots, waarin dit project is opgedeeld.

Het systeem zal ontwikkeld worden in kader van de afstudeeropdracht voor de opleiding VIA aan de Haagse Hogeschool

Na goedkeuring van de definitiestudie zullen de verschillende pilots ontwikkeld gaan worden met de specificaties zoals gedefinieerd in dit document.

## **2 Plan van aanpak**

### *2.1 Inleiding*

Dit document is het plan van aanpak voor het traject van de ontwikkeling van nieuwe style sheets met bijbehorend management systeem voor de website europe.fasson.com. De opdracht is in feite een optimalisatie van het gebruik van Cascading Style Sheets, in het vervolg CSS genoemd, op de website. De nieuw te ontwikkelen style sheets zullen met een management systeem gemakkelijk te onderhouden zijn. De ontwikkeling van dit management systeem is ook onderdeel van dit project.

De opdracht zal uitgevoerd worden in opdracht van Avery Dennison, divisie Roll Materials Europe (RME). De stagiair zal werkzaam zijn op het hoofdkantoor, gevestigd te Leiden. De uitvoerende stagiair, D.A. van den Hoek, is student aan de Haagse Hogeschool, sector Informatica.

Dit plan van aanpak zal fungeren als basisplan voor het gehele ontwikkeltraject. Er zal een korte achtergrond en probleemstelling gegeven worden waarin de huidige situatie beschreven staat. Verder beschrijft het plan de doelstellingen van het traject met bijbehorende eindresultaten, op te leveren producten en randvoorwaarden. Ook zullen de beheersaspecten van het project besproken worden, evenals de belangrijkste risicofactoren.

Na goedkeuring van dit plan van aanpak zal het ontwikkeltraject daadwerkelijk ingezet worden.

## 2.2 Achtergrondinformatie

### 2.2.1 Over Avery Dennison

Avery Dennison behoort tot de 500 grootste Amerikaanse bedrijven ("Fortune 500 company") en beschikt over 200 vestigingen verspreid over 39 landen wereldwijd. Avery Dennison is wereldwijd leider op het gebied van drukgevoelige technologieën en zelfklevende materialen, kantoorbenodigdheden, etiketteersystemen en speciale chemische producten, met consumenten- en industriële markten overal ter wereld.

### 2.2.2 Over RME

De stagiair zal ingezet worden bij de divisie Roll Materials Europe (RME), onderdeel van de Roll Materials World Wide Division. De stage zal volbracht worden op het hoofdkantoor, gevestigd in Leiden, op de afdeling IT Application Support and Development (ASD). De divisie RME heeft zich toegelegd op de productie van het merk Fasson; wereldmarktleider op het gebied van zelfklevende labels. De afdeling ASD draagt hierbij zorg voor de ontwikkeling en ondersteuning van de verschillende applicaties. Er worden applicaties ontwikkeld voor zowel klanten van Avery Dennison als voor de afdelingen van Avery Dennison zelf.

- Op het hoofdkantoor zijn 140 mensen werkzaam.
- De totale IT-afdeling bestaat uit 50 mensen, waarvan er 40 op het hoofdkantoor werkzaam zijn.
- De afdeling IT Application Support and Development bestaat uit 20 mensen waarvan de meeste ook op het hoofdkantoor werkzaam zijn.
- Er wordt ontwikkeld m.b.v. Lotus Notes en Visual Basic. Het ontwikkelteam bestaat uit 10 mensen. Zij houden zich vooral bezig met de behoeften van het e-Business team.
- De meeste IT-toepassingen zijn gecentraliseerd en worden onderhouden door de 40 IT-specialisten in Leiden.

### 2.3 Probleemstelling

De RME website is ontwikkeld in de Lotus Notes Domino omgeving. Dit is een gelimiteerde omgeving, als gevolg van het feit, dat Lotus Notes niet specifiek gericht is op het ontwikkelen van websites. De website is complex opgebouwd; de HTML van de pagina's wordt voor een groot deel door software gegenereerd. De codes komen uit verschillende databases. Dit heeft als gevolg, dat uiterlijke kenmerken op verschillende manieren gedefinieerd zijn; er wordt gebruik gemaakt van een style sheet, maar veelal worden stijlen in stukken HTML in de verschillende delen van de software geprogrammeerd. Dit gebeurt door verschillende ontwikkelaars, waarbij geen gebruik wordt gemaakt van standaarden of richtlijnen voor de stijlen. Deze inconsistente implementatie van verschillende oplossingen geeft grote problemen bij het onderhouden van de gebruikte stijlen op de website. Het is moeilijk aanpassingen te maken en in sommige gevallen is dit niet mogelijk.

Verder is het zo dat er op dit moment geen printbare versies van de pagina's op de site aanwezig zijn, iets dat ook gewenst wordt.

Men heeft het belang van Cascading Style Sheets, in het vervolg CSS genoemd, als oplossing voor de verschillende problemen m.b.t. de uiterlijke kenmerken ingezien. Ook de implementatie van CSS is echter ongeorganiseerd en gebeurt meer ad-hoc, zonder onderliggende structuur, waardoor de genoemde problemen blijven bestaan.

Op het moment is er een in Lotus Notes ontwikkelde omgeving voor het bijhouden van de content op de website, zoals teksten en afbeeldingen; een content management system. Ook is er een soortgelijke omgeving ontwikkeld om het menu te onderhouden. Er is nog geen generieke manier om vormgevingsaspecten te onderhouden. Het is gewenst dat de style sheets gemakkelijk en snel kunnen worden aangepast, m.b.v. een vergelijkbaar management systeem.

## 2.4 Doelstelling

Doelstelling van de opdracht is het opnieuw opzetten van de website, zodat deze optimaal gebruik maakt van CSS, waarbij in eerste instantie de huidige lay-out zoveel mogelijk gehandhaafd blijft. Hiervoor zal een style sheet ontwikkeld worden, waarin vastgelegd is hoe de verschillende HTML-elementen gepresenteerd dienen te worden op het scherm en een style sheet waarin vastgelegd is hoe deze elementen gepresenteerd moeten worden op papier.

Er zal een systeemanalyse gedaan worden omtrent de huidige functionaliteit en structuur. Ook zal er een adviesrapport opgesteld worden, gericht op het toepassen van 'best practices' voor CSS. Uitgaande van dit adviesrapport zal een tweede versie van de style sheets ontwikkeld worden, die de 'best practices' realiseren voor de website. De tweede versie zal voor beide style sheets ontwikkeld worden; een style sheet voor de presentatie op het scherm en een style sheet voor de presentatie op papier.

Om te bereiken dat de stijlen op de site gemakkelijk onderhouden kunnen worden, zal een management systeem voor de style sheets ontwikkeld worden, zodat aanpassingen in de style sheets gemakkelijk en snel kunnen worden doorgevoerd. Dit systeem zal gebruik maken van een eigen database, welke ook tijdens dit project ontwikkeld zal worden.

Hiervoor zullen style sheets volledig opnieuw ontwikkeld worden in CSS2, versie 2 van CSS. De student zal deze style sheets voor een aantal pagina's implementeren; hiervoor zal de HTML-code van deze pagina's ook gewijzigd worden. Dit zal gebeuren voor een aantal pagina's, waarvan vastgelegd is, dat deze gebruik maken van een standaard lay-out structuur. De wijzigingen zullen gedocumenteerd worden; deze documentatie zal gebruikt worden bij de implementatie van de overige pagina's van de website. Deze implementatie zal binnen een ander project plaatsvinden. Ook zal het management systeem voor de style sheet en de database volledig door de student ontwikkeld worden. Dit systeem wordt wel in z'n geheel door de student tijdens dit project geïmplementeerd. De ontwikkeling van het management systeem zal in Lotus Notes gedaan worden.

De website dient compatible te zijn met de volgende browsers, op zowel het Windows als het MacOS platform;

- Internet Explorer 5
- Internet Explorer 6
- Netscape 7

De opdracht zal bestaan uit de volgende onderdelen;

- De ontwikkeling van een heldere document tree van de huidige website, geldig voor alle pagina's, zowel de publieke als beveiligde transactiepagina's.
- Het ontwerp van een lay-outstructuur in CSS2, die zoveel mogelijk identiek is aan de huidige lay-out.
- Het ontwerp van een nieuwe, verbeterde lay-outstructuur.
- Systeemanalyse van de site in de huidige vorm. Samen met het ontwerp van de lay-out structuur en bevindingen uit de definitiestudie zal dit resulteren in een adviesrapport voor het e-Business team m.b.t. nodige aanpassingen op het gebied van lay-out en CSS-codering.
- De ontwikkeling van een style sheet die het mogelijk maakt deze lay-out te gebruiken voor alle eerder genoemde browser/OS combinaties.
- Het ontwerp en ontwikkeling van het management systeem met bijbehorende database, t.b.v. onderhoud van de style sheets
- De ontwikkeling van een generieke manier om de nieuwe style sheets en structuur met bijbehorend management systeem te implementeren.
- Er zijn op dit moment geen printbare versies van de pagina's. Er dient een nieuwe printlay-out ontworpen te worden
- De ontwikkeling van een tweede style sheet voor het printen van de pagina's in print lay-out.
- Implementatie van de aangepaste HTML-pagina's en de style sheets; het gaat hier om de pagina's die door de student tijdens dit project zijn aangepast zullen worden.
- Implementatie van het management systeem voor de style sheets
- Opstellen documentatie t.b.v. implementatie overige pagina's; het gaat hier om de pagina's die niet door de student zijn aangepast. Deze pagina's zullen binnen een ander project aangepast en geïmplementeerd worden.

De onderdelen staan niet in chronologische volgorde. (Zie hoofdstuk 9. Planning en activiteiten voor een overzicht van de planning)

Het is voor de opdracht van belang dat de stagiair een grondig inzicht ontwikkelt in de mogelijkheden en huidige ontwikkelstandaarden van CSS. De student zal dit bereiken door intensieve zelfstudie tijdens de eerste weken van het project.



## 2.5 Uitgangssituatie

### *-documentatie*

Er is nooit eerder onderzoek gedaan naar de eerder genoemde probleemstelling. Er is dan ook geen documentatie met betrekking tot dit onderwerp beschikbaar, die de student van dienst zou kunnen zijn.

Documentatie met betrekking tot de huidige programmering van de website e.d. kan wel gebruikt worden.

### *-software*

Het huidige CMS en het management systeem voor het menu tezamen met de huidige style sheet en de programmering van de website zullen bij de opdracht gebruikt worden.

Verder heeft de student de beschikking over de volgende specifieke software;

- Lotus Notes 5
- Internet Explorer 5
- Internet Explorer 6
- Netscape 7.1

## 2.6 Resultaat

De volgende producten zullen aan de opdrachtgever opgeleverd worden;

- Plan van aanpak voor de opdracht
- Definitiestudie, uitgebreid met;
  - ✓ Eén of meerdere document trees die de huidige website volledig beschrijven
  - ✓ Adviesrapport ten behoeve van het e-Business team. Dit rapport zal als los document opgeleverd worden.
- Producten;
  - ✓ De nieuwe style sheets; een style sheet voor presentatie op het scherm en een style sheet voor presentatie op papier.
  - ✓ Tweede, verbeterde, versie van de style sheets
  - ✓ Het management systeem voor de style sheets tezamen met de nieuw te ontwikkelen database voor dit management systeem.
- Systeemdokumentatie

## 2.7 Randvoorwaarden

### 2.7.1 Voorwaarden vanuit opleiding en organisatie

De stagiair is gebonden aan de eisen die de opleiding VIA stelt aan het ontwerpen en realiseren van producten. Het gehele ontwikkeltraject zal een iteratief karakter hebben en het systeem zal in nauwe samenwerking met de toekomstige gebruikers van het systeem gedefinieerd en ontworpen moeten worden. Als ontwikkelmethodiek is gekozen voor IAD om de gebruikersparticipatie te optimaliseren.

Verder zal binnen de voorwaarden van "Avery Dennison" gewerkt worden, wat inhoudt dat de stagiair in het belang van orde, veiligheid, en gezondheid de gegeven gedragsregels, voorschriften en aanwijzingen zoals deze voor het personeel van het bedrijf van toepassing zijn, in acht neemt.

### 2.7.2 Voorwaarden aan opdracht

Met betrekking tot de opdracht zal aan de volgende randvoorwaarden voldaan moeten worden:

- De style sheets zullen volledig opnieuw ontwikkeld worden in CSS2, versie 2 van CSS.
- Het uiterlijk van de site zal in eerste instantie bij de eerste versie van de style sheet zoveel mogelijk gelijk blijven, terwijl de achterliggende code opnieuw gestructureerd zal worden.
- De huidige website maakt gebruik van een CMS in Lotus Notes en een soortgelijk systeem t.b.v. het onderhoud van het menu. Het management systeem voor de style sheets zal hierop moeten aansluiten.
- Alle pagina's dienen compatible te zijn met zowel de IE5 en IE6 browsers, als de NS7 browser, op zowel het Windows als het MacOS platform.

## 2.8 Methoden en technieken

### 2.8.1 Algemeen

Voor de opdracht zal de ontwikkelmethode IAD (Iterative Application Development) gebruikt worden, toegepast in de vorm van RAD.

De IAD-ontwikkelmethodiek kent de volgende fases;

1. Definitiestudie – in deze fase wordt het nieuwe systeem in kaart gebracht. Mogelijkheden en beperkingen worden vastgelegd.
2. Pilotontwikkeling – in deze fase wordt het nieuwe systeem opgedeeld in pilots, die vervolgens ontwikkeld worden.
3. Invoering – het nieuwe systeem wordt ingevoerd.

Bij elk van deze fases kunnen iteraties plaatsvinden, de betreffende fase wordt dan dus meerdere keren doorlopen. IAD kent verschillende varianten; elke variant heeft een andere indeling van iteraties. Voor dit project zal de variant 'Rapid Application Development' (RAD) gebruikt gaan worden. Hierbij wordt alleen de fase pilotontwikkeling geïtereerd.

De volgende punten zijn de belangrijkste redenen geweest voor de keuze van RAD;

- De duidelijke probleemstelling en doelstelling hebben als gevolg dat een eenmalige definitiestudie voldoende is voor het hele project.
- Wanneer de fase van implementatie meerdere malen uitgevoerd zou worden, zou dit betekenen dat er *delen* van het systeem ingevoerd worden. Voor dit project zullen ook niet alle pagina's van de website aangepast worden, deze aanpassingen zullen binnen een ander project worden uitgevoerd. Het heeft voor de opdrachtgever dus geen nut om een *deel* van het systeem ingevoerd te hebben. Ook de fase invoering zal dus maar eenmalig uitgevoerd worden.
- Gekeken naar de uit te voeren activiteiten vergeleken met de beschikbare tijd is het wenselijk een minimum aan documentatie op te leveren. Alleen de noodzakelijke documenten zullen gemaakt worden. RAD is de IAD-variant met de minste documentatie.

Zie *par. 9.1 De globale fasering* voor een verdere toelichting en globale invulling van de 3 fases.

Verder zullen de volgende technieken worden toegepast;

- GUIDE (Graphical User Interface Design and Evaluation)
- Interviewtechnieken

### 2.8.2 Ontwikkelomgeving

De ontwikkelomgeving van het hele project is Lotus Notes Domino. Het operating system waarop gewerkt zal worden is Microsoft Windows 2000 professional. Verder zal gebruikt gemaakt worden van Lotus Domino servers voor het ontwikkelen en testen van de nieuwe style sheets en het management systeem.

Verder wordt er gebruik gemaakt van de volgende technieken;

- HTML
- JavaScript
- CSS (CSS2.0)
- Lotus Script
- Lotus Notes Formula Language

### **Hard – en software**

Er wordt gewerkt op een Dell werkstation (pentium IV, 512 MB SDRAM) met de volgende software;

- Lotus Notes 5
- Microsoft Internet Explorer 5.0
- Microsoft Internet Explorer 6.0
- Netscape 4.7 / 7.1
- AceHTML 5 Pro

Ook zal er getest worden op een I-Mac, met Microsoft Internet Explorer 5.2.

Er zal niet *ontwikkeld* worden voor Netscape 4.7. Het is echter wel gewenst te weten in hoeverre de pagina's correct getoond worden en waar ongeveer de knelpunten liggen. Dit zal kort gedocumenteerd worden.

## 2.9 Planning en activiteiten

### 2.9.1 De globale fasering

Het gehele traject wordt binnen de komende 18 weken volbracht. Als leidraad voor een fasering kan gekeken worden naar de activiteiten die de IAD methodiek beschrijft.

De IAD-ontwikkelmethodiek kent de volgende fases;

1. Definitiestudie
2. Pilotontwikkeling
3. Invoering

Zoals eerder vermeld is, zal voor dit project de IAD variant RAD toegepast worden. Hierbij wordt alleen de fase Pilotontwikkeling geïtereerd.

#### *Toelichting fasen IAD:*

1. Hierin worden de doelen van het systeem geanalyseerd, evenals de beperkingen. Er worden systeemeisen opgesteld. Vervolgens wordt een systeemconcept opgesteld, zijnde een eerste stap in de richting van een oplossing. Delen van het systeemconcept worden hierna geclusterd tot samenhangende eenheden die dienen als te implementeren pilots. Aan het eind van de definitiestudie wordt tevens, in het pilotplan, de richting van toekomstige inspanning bepaald.

In deze fase zal de student een grondig inzicht ontwikkelen in de mogelijkheden en huidige ontwikkelstandaarden van CSS (CSS2). Ook zal de student zich de Lotus Notes omgeving eigen maken in deze periode. Ook zal in deze fase de systeemanalyse plaatsvinden. De student zal op basis van deze analyse de benodigde document tree(s) ontwikkelen. Ook zal de nieuwe lay-out structuur (ook print lay-out) ontworpen worden. Dit alles zal resulteren in een definitiestudie.

2. In het pilotplan van de definitiestudie wordt uitgelegd welke diverse pilots er zijn. Nu zullen de pilots verder ontwikkeld en gebouwd worden. Dit alles wordt gedocumenteerd. Uiteindelijk zullen de ontwikkelde pilots samengevoegd worden tot de uiteindelijke CSS-documenten en het management systeem voor het onderhoud van deze documenten. Er wordt gestreefd naar 2 iteraties voor elke pilot.

In deze fase worden de nieuwe style sheets met de het bijbehorende managementsysteem en database ontwikkeld. Hiervoor zullen ook de aanpassingen gedaan worden in een aantal HTML-pagina's. Ook wordt de tweede versie van de style sheet ontwikkeld, evenals het adviesrapport.

3. In de laatste fase zal de acceptatie van het nieuwe systeem plaatsvinden. In samenwerking met de overige ontwikkelaars van ASD zal de student het nieuwe systeem implementeren; het gaat hierbij om de aangepaste pagina's, de style sheets en het ontwikkelde management systeem voor de style sheets. Ook hierbij zullen de nodige stappen gedocumenteerd worden.

### 2.9.2 De globale planning

Hieronder zal een globale weekplanning weergegeven worden van het gehele traject.

<b>Week</b>	<b>Activiteiten</b>	<b>Fase (IAD) (*1)</b>
<b>36</b>	-Opstellen Plan van Aanpak (IAD documentatie) -Introducties -Leer Lotus Notes	D
<b>37</b>	-Afronden Plan van Aanpak -Introducties -Systeem analyse (t.b.v. de ontwikkeling van document trees en ontwerp van de nieuwe lay-out structuren) -Zelfstudie "best practices in using CSS2.0"	D
<b>38</b>	-Systeem analyse (t.b.v. de ontwikkeling van document trees en ontwerp van de nieuwe lay-out structuren) -Opstellen definitiestudie (IAD documentatie) -Zelfstudie "best practices in using CSS2.0"	D
<b>39</b>	-Systeem analyse (t.b.v. de ontwikkeling van document trees en ontwerp van de nieuwe lay-out structuren) -Opstellen definitiestudie (IAD documentatie) -Zelfstudie "best practices in using CSS2.0"	D
<b>40</b>	-Systeem analyse (t.b.v. de ontwikkeling van document trees en ontwerp van de nieuwe lay-out structuren) -Opstellen definitiestudie (IAD documentatie) -Ontwikkeling definitieve document trees (*2) -Zelfstudie "best practices in using CSS2.0"	D
<b>41</b>	-Afronden systeem analyse -Ontwikkeling definitieve document trees (*2) -Opstellen definitiestudie (IAD documentatie) -Afronden zelfstudie "best practices in using CSS2.0" -Documentatie	D
<b>42</b>	-Zelfstudie Lotus Script/Lotus Notes Formula Language -Afronden definitiestudie -Documentatie	D

<b>43</b>	<ul style="list-style-type: none"> <li>-Zelfstudie Lotus Script/Lotus Notes Formula Language</li> <li>-Ontwikkeling eerste (*3) nieuwe style sheets (incl. style sheet t.b.v. printen)</li> <li>-Opstellen adviesrapport</li> <li>-Documentatie</li> </ul>	P
<b>44</b>	<ul style="list-style-type: none"> <li>-Zelfstudie Lotus Script/Lotus Notes Formula Language</li> <li>-Ontwikkeling eerste (*3) nieuwe style sheet (incl. style sheet t.b.v. printen)</li> <li>-Opstellen adviesrapport</li> <li>-Ontwikkeling van management systeem voor de style sheets en bijbehorende database</li> <li>-Documentatie</li> </ul>	P
<b>45</b>	<ul style="list-style-type: none"> <li>-Zelfstudie Lotus Script/Lotus Notes Formula Language</li> <li>-Afronden ontwikkeling eerste (*3) style sheets (incl. style sheet t.b.v. printen)</li> <li>-Afronden adviesrapport</li> <li>-Ontwikkeling van management systeem voor de style sheets en bijbehorende database</li> <li>-Documentatie</li> </ul>	P
<b>46</b>	<ul style="list-style-type: none"> <li>-Zelfstudie Lotus Script/Lotus Notes Formula Language</li> <li>-Ontwikkeling tweede (*3) nieuwe style sheets (incl. style sheet t.b.v. printen)</li> <li>-Ontwikkeling van management systeem voor de style sheets en bijbehorende database</li> <li>-Documentatie</li> </ul>	P
<b>47</b>	<ul style="list-style-type: none"> <li>-Ontwikkeling tweede (*3) nieuwe style sheets (incl. style sheet t.b.v. printen)</li> <li>-Ontwikkeling van management systeem voor de style sheets en bijbehorende database</li> <li>-Documentatie</li> </ul>	P

<b>48</b>	-Afronden ontwikkeling tweede (*3) nieuwe style sheets (incl. style sheet t.b.v. printen) -Afronden ontwikkeling van management systeem voor de style sheets en bijbehorende database -Implementatie (in samenwerking met ontwikkel team) -Documentatie	P/I
<b>49</b>	-Implementatie (in samenwerking met ontwikkel team) -Documentatie	I
<b>50</b>	-Implementatie (in samenwerking met ontwikkel team) -Documentatie	I
<b>51</b>	-Opstellen eindverslag	N.V.T.
<b>52</b>	-Opstellen eindverslag	N.V.T.
<b>1/2</b>	-Opstellen eindverslag	N.V.T.

**\*1**

Er bestaan verschillende varianten van IAD, afhankelijk van welke fases iteratief zijn. Voor dit project zal RAD (Rapid Application Development) gebruikt worden. In deze variant wordt alleen fase 2 (pilotontwikkeling) geïtereerd. Definitiestudie en invoering worden elk eenmalig uitgevoerd.

**\*2**

Er wordt hier gesproken over *definitieve* document trees, omdat er tijdens de systeemanalyse ook al document trees ontwikkeld zullen worden, als hulpmiddel bij de analyse en als voorbereiding op de ontwikkeling van de definitieve document trees in week 40. Deze document trees zullen gebruikt worden bij de ontwikkeling van de nieuwe style sheets.

**\*3**

Er zullen 2 nieuwe lay-outs ontworpen worden en er zullen dus ook 2 (sets) nieuwe style sheets ontwikkeld worden. De eerste lay-out zal zoveel mogelijk in de bestaande look & feel van de website zijn. De tweede lay-out zal een verbeterde lay-out zijn. Voor beide lay-outs zal uiteraard ook een printversie ontworpen worden. Beide lay-outs moeten bereikt worden in overeenstemming met Anne Christine van Geelen (Web Content Manager) en Monique Houben (e-Business Manager), dus dit is een iteratief proces op zichzelf. Er wordt verwacht dat hiervoor maximaal 3 iteraties nodig zijn.



## 2.10 Beheersaspecten

Onder beheersaspecten worden verstaan de zogenaamde GOTIK-factoren. Dit zijn Geld, Organisatie, Tijd, Informatie en Kwaliteit. Deze zullen in de volgende paragrafen behandeld worden.

### 2.10.1 Beheersen van de te maken kosten

Dit project zal in principe met een gesloten beurs uitgevoerd worden met uitzondering van de daarvoor vastgestelde bruto stagevergoeding van € 410,- per maand. Verder moet er rekening gehouden worden met kleine kosten voor lesmateriaal e.d. De student zal tijdens het project nl. veel zelfstudie uitvoeren.

Verder is er (op dit moment) weinig inzicht in de financiële kant van het project. In een later stadium zou eventueel nog door de stagiair onderzocht kunnen worden wat een dergelijk project kost in termen van b.v. de tijd dat verschillende projectleden met het project bezig zijn.

### 2.10.2 Beheersen van de organisatie

Opdrachtgever:	Avery Dennison - RME
Bedrijfsmentor:	Frank Steeman
Technische ondersteuning:	Frank Steeman en Mee Lian Lee

Contactpersonen:	Dennis van den Hoek
------------------	---------------------

Projectleden:	Dennis van den Hoek
---------------	---------------------

Examinatoren:	Patrick Deters / Peter van Leeuwen
Keurend docent:	Patrick Deters

#### Bevoegdheden:

Binnen de marges van het in de projectopdracht gestelde, is de stagiair beslissingsbevoegd. In geval van twijfel wordt een beslissing voorgelegd aan de bedrijfsmentor, die vervolgens een beslissing neemt of verwijst naar de projectbegeleider. Er zal intensief overleg plaatsvinden met de bedrijfsmentor over de verschillende aspecten van het project, zoals systeemconcepten, kwaliteit, planning en voortgang.

### *2.10.3 Beheersen van tijd*

Er zal in totaal 19 weken aan het project besteed worden. Een globale invulling hiervan is terug te vinden in *par. 9.2 De globale planning*.

Om het project te kunnen beheersen in het kader van de tijd die voor het project beschikbaar is, worden gedurende het project de volgende (projectbeheersings)documenten opgeleverd:

- Plan van Aanpak
- Definitiestudie
- Pilotontwikkelplannen

### *2.10.4 Beheersen van informatie*

Voor het beheersen van de systeemontwikkeling is het belangrijk dat voldoende informatie m.b.t. deze systeemontwikkeling beschikbaar is. Parallel aan de systeemontwikkeling wordt dan ook de systeemdokumentatie opgesteld. Deze documentatie verschaft duidelijkheid aan zowel de stagiair als de bedrijfsmentor tijdens het ontwikkelen. Na oplevering dient de systeemdokumentatie als naslagwerk bij onderhoudswerkzaamheden of eventuele uitbreidingen. De systeemdokumentatie kan tevens dienen als basis voor het maken van documentatie voor de toekomstige gebruikers.

### *2.10.5 Beheersen van kwaliteit*

De opdracht behoudt kwaliteit, mede doordat de stagiair en de bedrijfsmentor intensief contact zullen onderhouden en alle belangrijke stappen van het project met elkaar bespreken. Door de bedrijfsmentor regelmatig op de hoogte te houden van de vorderingen krijgt de stagiair feedback die tevens de kwaliteit waarborgt en voorkomt dat er afgedwaald wordt van het doel.

### 2.11 Risicofactoren

Tijdens het project worden een aantal punten als belangrijkste risicofactoren onderkent;

- De ontwikkeling van het managementsysteem zou te zwaar kunnen zijn. Door het gebruik van technieken nieuw voor de stagiair, zou kunnen blijken dat de ontwikkeling van dit systeem zodanig ingewikkeld wordt dat dit te veel tijd zal gaan kosten waardoor de voortgang van het project in het gedrang zou kunnen komen.
- Ook moet er rekening gehouden worden met de complexiteit van de gehele website in de huidige vorm. De systeemanalyse heeft als primair doel duidelijke document trees op te stellen. Deze document trees samen beschrijven als het ware de gehele lay-outstructuur van de website. Met het oog op de complexiteit van de website zou kunnen blijken dat het opstellen van algemeen geldende document trees veel tijd gaat kosten, wat ook weer de voortgang van het project zou kunnen storen.

Wanneer een van bovengenoemde punten voorkomen tijdens het project, zal in overleg met de bedrijfsmentor en de keurend docent (projectbegeleider) de planning van de uitvoering van het project aangepast worden. De opdracht zal dan zodanig aangepast moeten worden, dat het project alsnog binnen de gestelde tijd afgerond kan worden.

- Een laatste 'valkuil' is de mogelijkheid dat er steeds meer wijzigingen gewenst worden in de pagina's van de website, die steeds meer van de opdracht afwijken.

Om dit te voorkomen zal de stagiair voor zichzelf duidelijke grenzen moeten stellen en zich houden aan de afgesproken systeemeisen. Op deze manier moet bereikt worden dat alle uit te voeren activiteiten met het oog op de doelstelling van het project uitgevoerd worden. De bedrijfsmentor zal hierbij ook een controlerende rol hebben.

### 3 Resultaten systeemanalyse

#### 3.1 Inleiding

De eerste stap van de definitiestudie bestaat uit een systeemanalyse. Deze analyse heeft als belangrijkste doelen de huidige situatie in kaart te brengen en voldoende input te genereren voor de ontwikkeling van een aantal document trees voor de pagina's van de website. Deze document trees zijn in feite schematische overzichten van de structuur van de betreffende pagina en beschrijven uit welke elementen een pagina is opgebouwd en hoe deze elementen met elkaar samenhangen. Het gaat hierbij uiteraard alleen om de elementen die samenhangen met de lay-out van een pagina. Het resultaat van de systeemanalyse is een aantal structuren, die de lay-out van de verschillende pagina's beschrijven. Op basis van deze structuren zal een set document trees ontwikkeld worden, die samen alle pagina's van de website beschrijven, op enkele uitzonderingen na.

In feite heeft elke pagina een eigen document tree. Het doel is echter om uiteindelijk een set (zo min mogelijk) trees te ontwikkelen voor de gehele website. Er zullen dus standaard structuren ontdekt moeten worden, om zo document trees te kunnen ontwikkelen die voor meerdere pagina's geldig zijn. Eerst zal de website beschreven worden m.b.t. de uiterlijke kenmerken. Hiervoor zullen screenshots gebruikt worden van verschillende pagina's van de website met daarbij een beschrijving van de belangrijkste elementen. Vervolgens zal bekeken worden hoeveel verschillende standaard lay-outs te ontdekken zijn, hiervan zullen categorieën gemaakt worden. Van elke lay-outcategorie zal de structuur in kaart gebracht worden.

Er zal een overzicht ontstaan van verschillende structuren; elke categorie heeft een eigen structuur. Dit overzicht zal als input dienen voor de ontwikkeling van de document trees.

Dit hoofdstuk is in feite een uittreksel van de volledige systeemanalyse; er zal een beschrijving van de website gegeven worden, gevolgd door een beschrijving van de verschillende lay-out structuren. Hierna zal een hoofdstructuur van de volledige website gegeven worden met daarna de verschillende document trees die opgesteld zijn. Voor gedetailleerde informatie over de ontwikkeling van deze document trees wordt verwezen naar de volledige documentatie van de systeemanalyse.

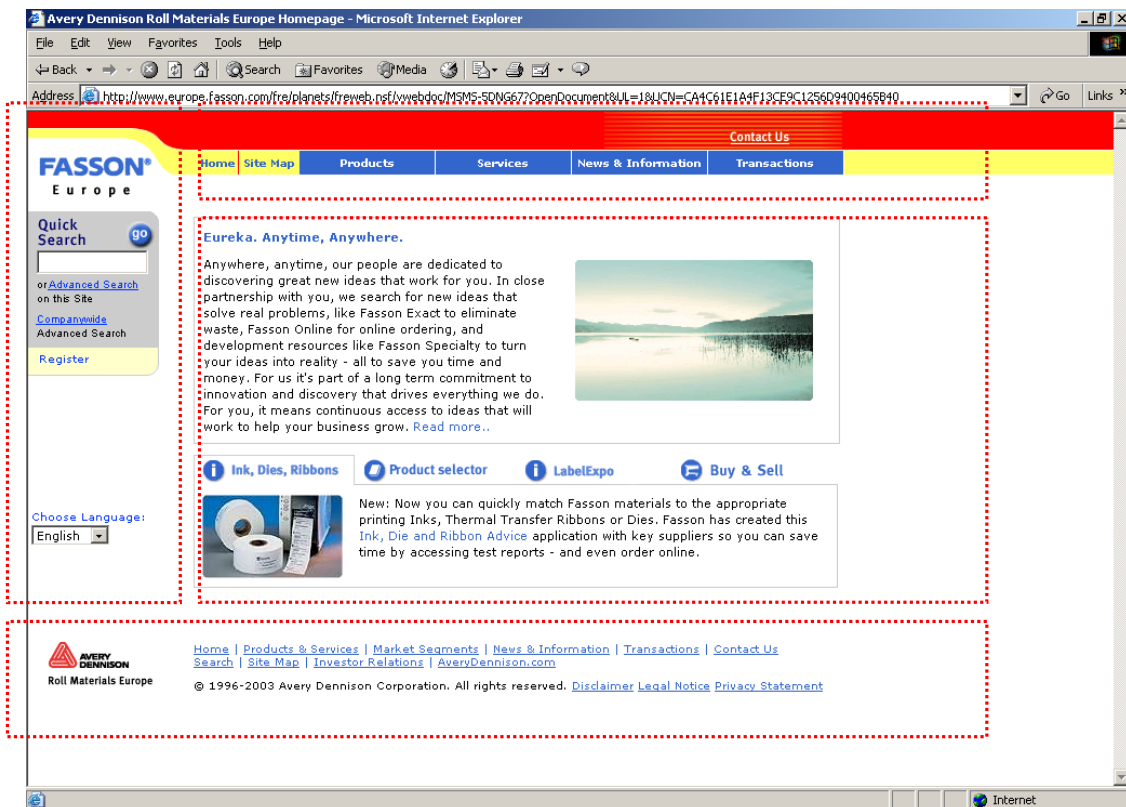
Naast een analyse voor de document trees zal ook een analyse gedaan worden van het huidige management systeem, dat gebruikt wordt voor het menu. Hiervoor bestaat een applicatie in Lotus Notes, die in zeer veel opzichten zal lijken op de applicatie die ontwikkeld zal gaan worden als management systeem voor de style sheets. Van deze applicatie zal een analyse gedaan worden; er zal een beschrijving gegeven worden van het uiterlijk van de applicatie en wat de mogelijkheden zijn. Verder zal een globale technische beschrijving gegeven worden; dit als voorbereiding op de ontwikkeling van de technische structuur van het management systeem voor de style sheets.

### 3.2 Beschrijving website

In dit hoofdstuk zal een beschrijving gegeven worden van de website aan de hand van screenshots van verschillende pagina's. Vervolgens zal er een overzicht gegeven worden van verschillende categorieën; elke pagina hoort thuis in één van de categorieën.

#### 3.2.1 Beschrijving homepage

Het volgende screenshot is een afbeelding van de homepage van de website.



Afb. 1 - De homepage

Deze lay-out is in te delen in 4 blokken, allen met min of meer een eigen functie. De blokken zijn in de afbeelding aangegeven m.b.v. een rood kader. Het bovenste blok bevat het menu en een contact link. Het linkerblok bevat het Fasson Europe logo, een zoekmodule (Quick Search, voor zoekopdrachten op de website) en een taalkeuze. Het rechter blok bevat de content van de betreffende pagina. Het onderste blok bevat informatie over copyright en enkele links. De inhoud en indeling van het linker en rechter blok kan variëren; hierop wordt terug gekomen in *par. 3.2.1.2 Dynamische blokken*.

#### 3.2.1.1 Statische elementen

Er zijn een aantal elementen op de homepage, die op elke pagina terug komen. Het gaat om de volgende elementen;

- De header; het menu en de contact-link
- De Quick Search in de linker kolom en het Fasson Europe logo
- De complete footer; het Avery Dennison logo, copyright informatie en links

Op de volgende pagina zijn deze elementen uitvergroot weergegeven.  
(Afbeelding 2, 3 en 4)



Afb. 2 - De header van de website



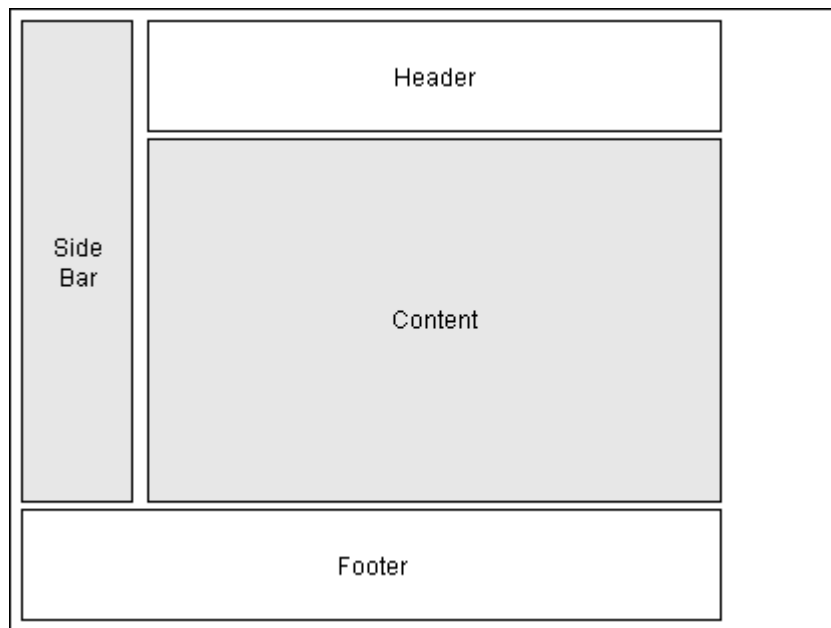
Afb. 3 - De Quick Search en het logo



Afb. 4 - De footer van de website

De elementen op afbeeldingen 2, 3 en 4 zijn dus op elke andere pagina ook aanwezig. Het is in feite het frame waarbinnen alle content van de verschillende pagina's getoond wordt. (De website maakt overigens geen gebruik van framesets) De lay-out van deze delen is dus voor alle pagina's gelijk.

Afbeelding 5 is een schematische weergave van de homepage. Elke pagina van de website is op deze manier opgebouwd, uit dezelfde blokken. De header en footer zijn voor alle pagina's gelijk. De inhoud van de sidebar kan variëren, de Quick Search is wel altijd aanwezig. De inhoud van de content varieert uiteraard ook per pagina.



Afb. 5 - Schematische weergave van de homepage

### 3.2.1.2 Dynamische blokken

Er zijn twee dynamische 'blokken' op de pagina's aanwezig zijn. Het gaat om de volgende delen;

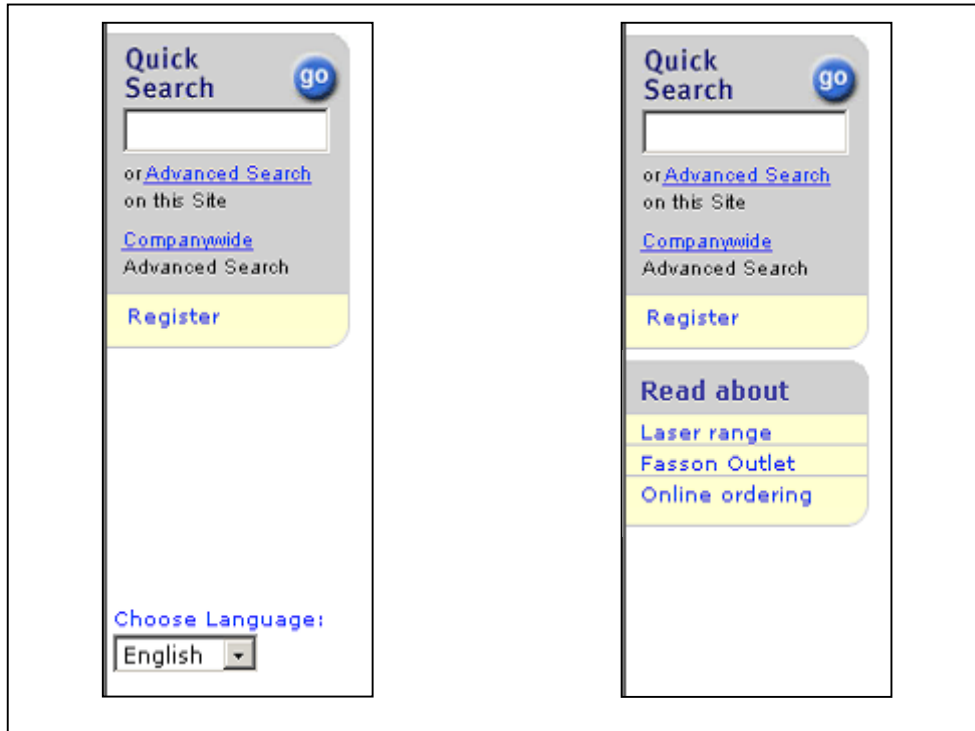
- De inhoud van de sidebar
- De inhoud van de content

In de volgende paragrafen zullen deze delen (blokken) apart besproken worden. Voor beide blokken zullen de verschillende varianten van invulling beschreven worden. Wanneer blijkt dat een blok verschillende lay-outs kan bevatten, zal er voor dat blok voor elke lay-out een categorie opgesteld worden. Elke lay-outstructuur krijgt dus een eigen categorie, zodat pagina's met een gelijke lay-outstructuur bij elkaar in een categorie gezet kunnen worden. Op deze manier ontstaat er dus een aantal basisstructuren, welke als basis zullen dienen voor de verdere ontwikkeling van de document trees. Er wordt gestreefd naar een zo klein mogelijk aantal categorieën.



### *Inhoud van de sidebar*

Op de volgende afbeelding zijn twee invullingen te zien van de sidebar, het eerste dynamische blok. Het logo is hier overigens uitgelaten; deze staat op elke pagina, boven de Quick Search. Het gaat hier om verschillende invullingen;



Afb. 6 - Invullingen van de sidebar


Er zijn in totaal twee varianten van de sidebar te vinden, beide afgebeeld op afbeelding 6. Zoals op de afbeelding te zien is, wordt voor beide invullingen dezelfde lay-out gehanteerd; het is voor dit blok niet nodig verschillende categorieën op te stellen. Eén lay-outstructuur zal dus voldoende zijn om de lay-out van dit blok te beschrijven.


### Inhoud van de content


Het andere dynamische blok is de inhoud van de content, waarvan op de volgende afbeelding de content van de homepage te zien is.


**Eureka. Anytime, Anywhere.**


Anywhere, anytime, our people are dedicated to discovering great new ideas that work for you. In close partnership with you, we search for new ideas that solve real problems, like Fasson Exact to eliminate waste, Fasson Online for online ordering, and development resources like Fasson Specialty to turn your ideas into reality - all to save you time and money. For us it's part of a long term commitment to innovation and discovery that drives everything we do. For you, it means continuous access to ideas that will work to help your business grow. [Read more..](#)





**Ink, Dies, Ribbons**


**Product selector**


**LabelExpo**


**Buy & Sell**



New: Now you can quickly match Fasson materials to the appropriate printing Inks, Thermal Transfer Ribbons or Dies. Fasson has created this [Ink, Die and Ribbon Advice](#) application with key suppliers so you can save time by accessing test reports - and even order online.

Afb. 7 - Inhoud van een contentblok – Lay-out zoals homepage


Op meerdere achterliggende pagina's is de lay-out structuur van de homepage terug te vinden. Afbeelding 8 is een voorbeeld van een andere pagina met die betreffende lay-out.


[Home](#) > Non-adhesive Products


**Fasson Rapid-Roll® Homepage**


When you need to move FAST, Fasson Rapid-Roll delivers at high speed.


It's your single source for a wide range of non-adhesive materials - paper, cardboard, film, siliconized materials and flexible packaging products slit to order - all available in surprisingly low minimum order quantities, and in a uniquely fast 48-hour turnaround time from order to dispatch. [Read more..](#)





**Product/price list**


**New product**


**Ink, Dies, Ribbons**


**Product Datasheets**



Fasson Rapid-Roll has an extensive product and price list. Click [here](#) to view the product list. To view the latest prices go to our [pricelist](#). Please note that you will need to be an Advantage member to do so. Advantage members can avail of a wide range of both online and offline services. Please [register](#) here to become an Advantage member.

Afb. 8 - Inhoud van een contentblok – Lay-out zoals homepage

Deze lay-out bestaat uit een aantal elementen; bovenaan zijn de 'bread crumbs' te vinden. Daaronder is een kader geplaatst met een introducerende tekst en een afbeelding, een eigen titel en meestal een link naar een pagina met meer informatie over het betreffende onderwerp. In het kader daaronder is content te vinden in de vorm van tekst en een afbeelding. Dit kader heeft 4 tabs, zoals te zien is op de afbeeldingen 7 en 8. De inhoud van de content die bij een tabblad hoort is steeds op dezelfde manier ingedeeld.

Bij het navigeren door de verschillende pagina's van de website, komt men naast de lay-out van de homepage nog een aantal andere lay-outs voor het contentblok tegen. Voorbeelden hiervan zijn getoond op de volgende afbeeldingen.

[Home](#) > [News & Information](#) > Brochures

## Brochures

---

**We have a range of brochures available. All are grouped by market segment. You can choose between a hard copy or an electronic version (PDF). Both are free of charge.**

---

Select your preferred format first, then choose a market segment.

[Request hardcopy](#)

Or view electronic version (PDF) by choosing a Market Segment:

<a href="#">Beverage</a>	<a href="#">Electronics / Automotive</a>
<a href="#">Food</a>	<a href="#">Health &amp; Beauty Care</a>
<a href="#">Household Chemical</a>	<a href="#">Industrial Chemical &amp; Oil</a>
<a href="#">Logistics &amp; Transport</a>	<a href="#">Office</a>
<a href="#">Retail</a>	<a href="#">Various</a>

Afb. 9 - Inhoud van een contentblok

[Home](#) > [Products & Services](#) > Product Datasheets



## Product Datasheets


You can access the technical data sheets for the Fasson and Jac standard product range in several ways. To request a matching A4 sample, click on the product code on the product data sheet.


---

**Brand :**

**Face :**

**Adhesive :**   

**Product number :**  

**EAN Code :**  


Afb. 10 - Inhoud van een contentblok

[Home](#) > Raw Material Collaboration






**Welcome to Ink, Die and Ribbon Advice**

Save time and costs - take advantage of the collaborative knowledge base of Avery Dennison Roll Materials Europe and well-established Ink, Die and Ribbon manufacturers to access print, die and ribbon compatibility-test reports based on pre-tested materials.

First select the segment you would like to use by clicking the radio buttons on the bottom of the page. Then choose a manufacturer. You can then select Fasson Facematerial, and depending on the segment the printing process, the die type or the print head type. Click on search and access the outlined test reports. Click [here](#) to read our Disclaimer.





**Latest additions to Ink, Die and Ribbon Advice:**

-  [New test result](#) Segment: Ink, Fasson Material: FASSON PP NG TOP WHITE, Printing Process: UV Rotary Screen
-  [New test result](#) Segment: Ink, Fasson Material: FASSON PP NG TOP WHITE, Printing Process: UV Flexo
-  [New test result](#) Segment: Ink, Fasson Material: FASSON PP NG TOP WHITE, Printing Process: UV Flat-bed Screen
-  [New test result](#) Segment: Ink, Fasson Material: FASSON PP TOP TRANS, Printing Process: UV Rotary Screen
-  [New test result](#) Segment: Ink, Fasson Material: FASSON PP TOP TRANS, Printing Process: UV Flexo

**Choose a segment and search for a test result**

☐ Die ☐ Ink ☐ Ribbon

[Reset](#)  [Search](#) 

Afb. 11 - Inhoud van een contentblok

Op het eerste gezicht lijken er grote verschillen te zijn tussen de lay-outs op afbeelding 9, 10 en 11. In feite laten afbeeldingen 9 en 10 dezelfde lay-out zien. Natuurlijk zijn er verschillen maar de basislay-out is voor beide afbeeldingen hetzelfde; boven zien we eerst de 'bread crumbs' met daaronder de titel van de pagina. Daaronder staat een inleidend stukje tekst, tussen twee horizontale rode lijnen. Als laatste staat hieronder de werkelijke content, in de vorm van tabellen, alleen tekst, input velden, afbeeldingen of een combinatie hiervan.

Afbeelding 11 toont een andere lay-out, hoewel deze vergelijkbaar is met de lay-out van afbeelding 7 en 8. Bovenaan zijn weer de 'bread crumbs' te vinden. Daaronder een kader met introducerende tekst en een eventuele afbeelding, deze afbeelding is dus niet altijd aanwezig, zoals in het geval van de lay-out van afbeelding 7 en 8. Dit gedeelte heeft weer een eigen titel en meestal een link naar een pagina met meer informatie. Onder het eerste kaders kunnen meerdere kaders voorkomen met content in de vorm van tekst, input velden, afbeeldingen, tabellen of een combinatie hiervan. In dit geval (afb. 11) zijn het twee kaders met content; dit kan variëren.

- Lay-out 1 – Afb. 7 en afb. 8
- Lay-out 2 – Afb. 9 en 10
- Lay-out 3 – Afb. 11

Deze lay-outstructuren worden willekeurig door elkaar gebruikt op de website. Er wordt dus geen onderscheid gemaakt in de lay-outstructuur voor bepaalde delen van de website; het is niet zo dat bepaalde onderdelen van de site een eigen herkenbare lay-out hebben.

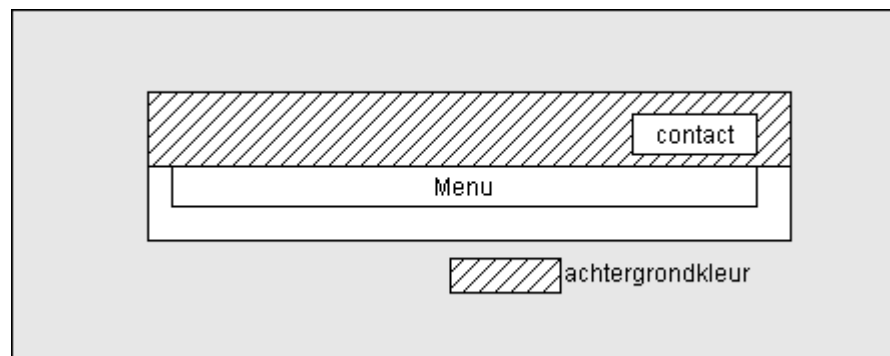
Het feit dat er voor het contentblok drie verschillende lay-outs gebruikt worden, betekent dus dat er voor dit blok drie lay-outstructuren opgesteld zullen worden. Voor de overige blokken (sidebar, header en footer) zal steeds één lay-outstructuur per blok voldoende zijn om de lay-out van het betreffende te beschrijven. Totaal zullen er dus 6 verschillende lay-outstructuren opgesteld worden, wat als gevolg heeft dat er uiteindelijk ook 6 document trees ontwikkeld zullen worden. (deze kunnen samengevoegd worden tot 3 verschillende document trees, waarvan een gedeelte steeds hetzelfde is. Dit is nl het gedeelte dat de header, footer en sidebar beschrijft.) De structuren per blok worden in het volgende hoofdstuk beschreven.

### 3.3 Structuren per blok

In de volgende paragrafen zullen de verschillende lay-out structuren per blok beschreven worden. Er zal steeds een schematisch overzicht van de structuur gegeven worden, met een beschrijving van de inhoud.

#### 3.3.1 Lay-outstructuur header

Op de volgende afbeelding is een schematische weergave te zien van de lay-out van de header. Dit is in feite een schematische weergave van afbeelding 2.

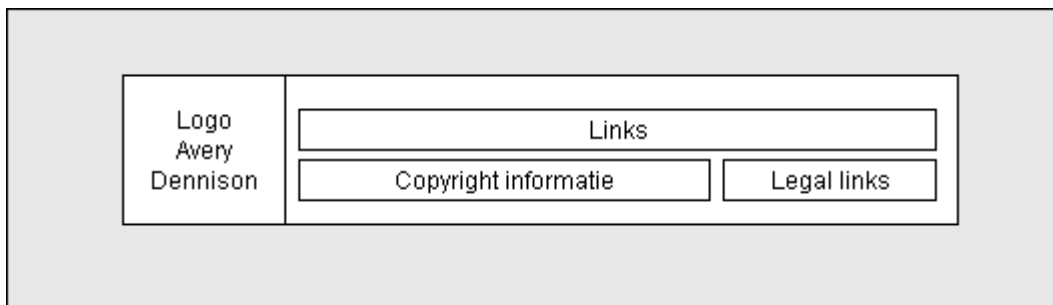


Afb. 12 - Schematische weergave van de header

De indeling van de header is over de gehele website hetzelfde. Het menu is overal hetzelfde en staat rechts, in het onderste gedeelte van de header. Het bovenste rechter deel van de header heeft een rode achtergrondkleur. Helemaal rechtsboven is een 'contact-link' geplaatst.

### 3.3.2 Lay-outstructuur footer

Op de volgende afbeelding is een schematische weergave te zien van de lay-out van de footer. Dit is in feite een schematische weergave van afbeelding 4.

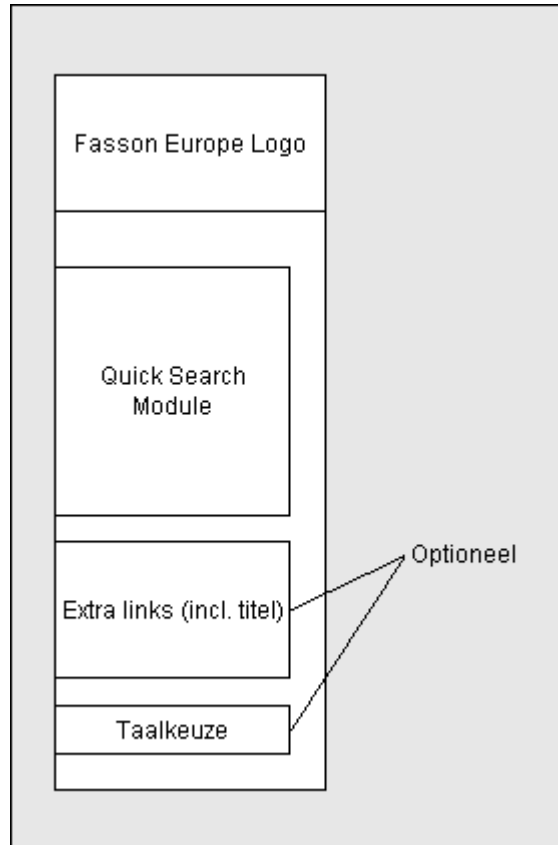


Afb. 13 - Schematische weergave van de footer

De indeling van de footer is over de gehele website hetzelfde; links staat weer een logo, dit keer van Avery Dennison. Rechts in de bovenste helft van de footer staan links naar verschillende pagina's binnen en buiten de website. Daaronder, in het rechter deel staat informatie over copyright met daarnaast een aantal links naar informatie over juridische aspecten (disclaimer e.d.).

### 3.3.3 Lay-outstructuur sidebar

Op de volgende afbeelding is een schematische weergave te zien van de lay-out van de sidebar. Dit is in feite een schematische weergave van afbeelding 6.



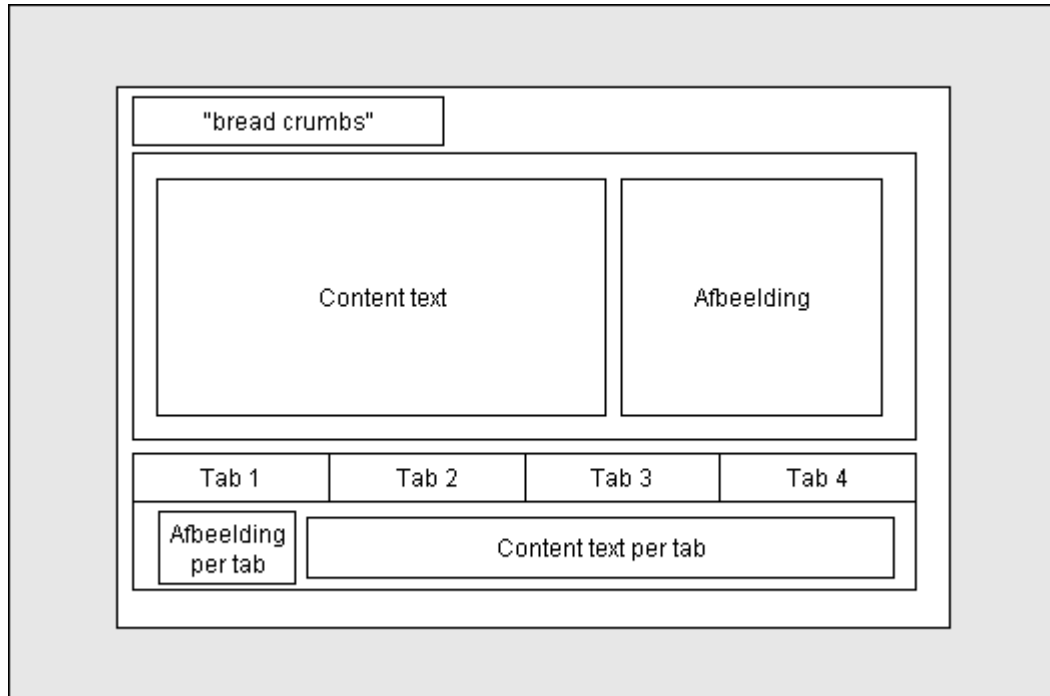
Afb. 14 - Schematische weergave van de sidebar

De invulling van de sidebar kan variëren, echter de indeling is wel hetzelfde over de hele site. Bovenin staat het Fasson Europe logo afgebeeld met daaronder de Quick Search module, deze zijn beiden op iedere pagina aanwezig. Daaronder zijn eventueel extra links te vinden; deze kunnen ook variëren maar hebben wel altijd dezelfde lay-out (zie afb. 6, rechts). Daaronder is eventueel nog een taalkeuzemenu in de vorm van een drop-down-list-box. Deze is terug te vinden op de homepage (zie afb. 6, links).



### 3.3.4 Lay-outstructuur 1 content

Op de volgende afbeelding is een schematische weergave te zien van lay-out 1 van de content. Dit is in feite een schematische weergave van afbeelding 7 en 8.

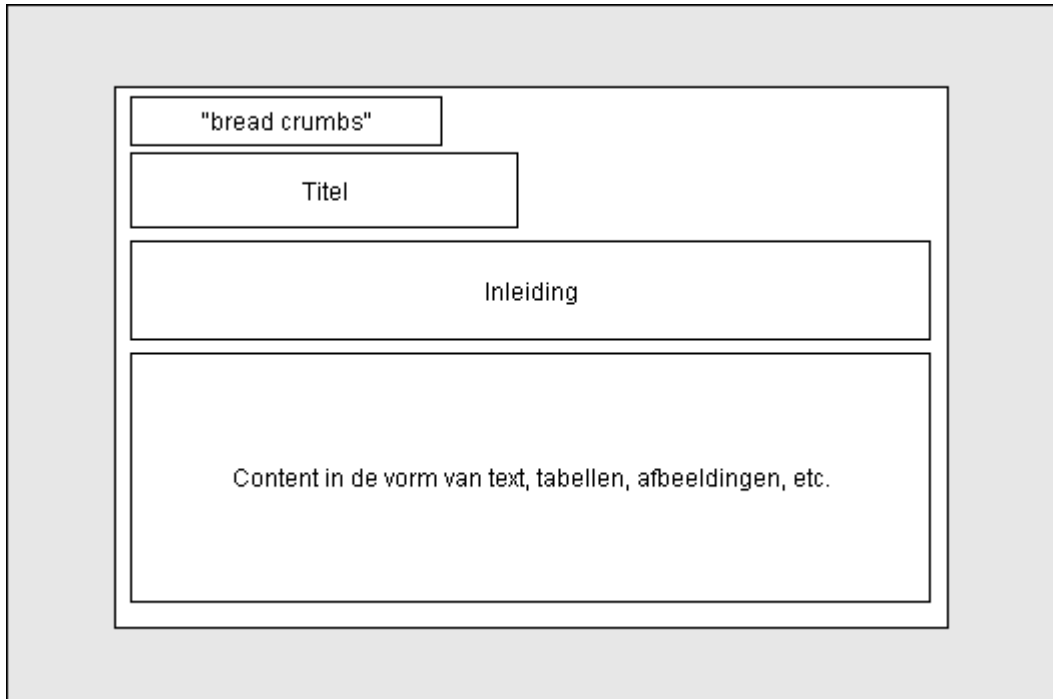


Afb. 15 - Schematische weergave van de content – lay-out 1

Zoals gezegd is het content blok één van de blokken waarvan de inhoud kan variëren. De indeling is echter altijd volgens één van de drie opgestelde lay-out structuren, waarvan afbeelding 15 een schematische weergave vertoont van de eerste lay-out structuur. Bovenaan zijn weer de 'bread crumbs' te zien. Daaronder bevindt zich een kader; deze is altijd op dezelfde manier ingevuld. Links in dit kader staat content in de vorm van tekst, met een eigen titel en vaak ook een link naar een pagina met meer informatie. Rechts in dit kader staat een afbeelding. Het tweede kader is ingedeeld als tabbladen. Elke tab is klikbaar en toont in de ruimte eronder de bijbehorende content. (op afbeelding 7 en 8 is de content te zien horend bij de eerste tab) Deze content heeft ook steeds weer dezelfde indeling; links een afbeelding, rechts content in de vorm van tekst.

### 3.3.5 Lay-outstructuur 2 content

Op de volgende afbeelding is een schematische weergave te zien van lay-out 2 van de content. Dit is in feite een schematische weergave van afbeelding 9 en 10.

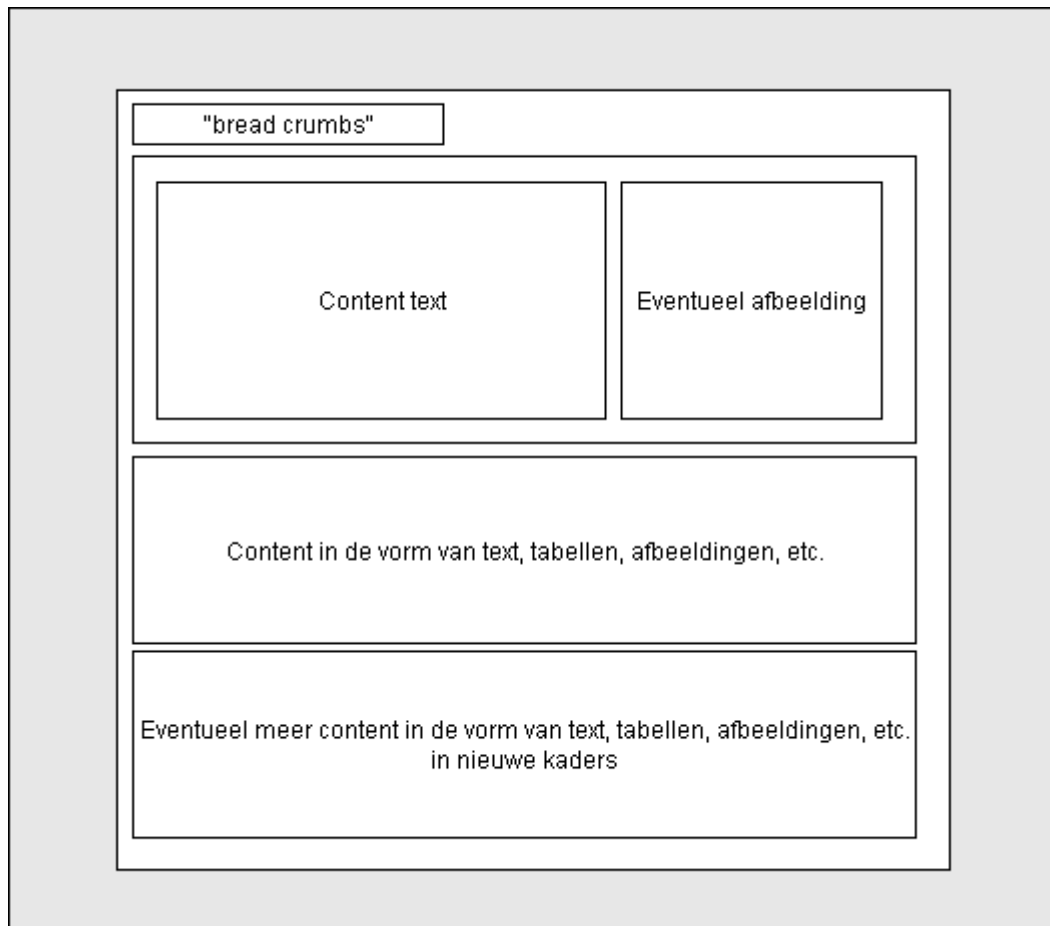


Afb. 16 - Schematische weergave van de content – lay-out 2

Lay-out 2 heeft wederom de 'bread crumbs' bovenaan staan. Daaronder staat de titel van de pagina, met daaronder een introducerende tekst. Deze tekst staat, in deze lay-out, altijd tussen de twee rode horizontale lijnen. Het onderste deel bevat de eigenlijke content van de betreffende pagina, in de vorm van tabellen, alleen tekst, afbeeldingen of een combinatie hiervan.

### 3.3.6 Lay-outstructuur 3 content

Op de volgende afbeelding is een schematische weergave te zien van lay-out 3 van de content. Dit is in feite een schematische weergave van afbeelding 11.



Afb. 17 - Schematische weergave van de content – lay-out 3

Zoals eerder gezegd heeft lay-out 3 veel overeenkomsten met lay-out 1. Ook dit is goed terug te zien in de schematische weergave. Ook hier staan de 'bread crumbs' weer bovenaan. Daaronder staat weer een kader net als bij lay-out 1. De invulling van dit kader kan echter variëren; het gaat hier in ieder geval om een introducerende tekst. Eventueel kan hier nog een afbeelding of een tabel bij staan.

Onder dit kader kunnen meerdere kaders geplaatst zijn; in het geval van afbeelding 11 en de schematische weergave zijn dit twee extra kaders. Dit aantal kan dus variëren, evenals de invulling ervan. Dit kan bestaan uit tekst, tabellen en afbeeldingen of combinaties hiervan.

### 3.4 Van lay-out structuur naar document tree

De volgende belangrijke stap in het toewerken naar de document trees, heeft als doel de HTML-elementen te ontdekken die verantwoordelijk zijn voor de lay-out van de verschillende structuren. Voor elke structuur zal bekeken worden welke elementen dit zijn en hoe deze met elkaar samenhangen. Van deze elementen kan dan per lay-out structuur een document tree opgesteld worden. Alle trees samen vormen dan de document tree voor een lay-out structuur. Aangezien er drie verschillende lay-outs zijn, zullen er uiteindelijk ook drie document trees opgesteld worden. Delen van deze trees zullen overeenkomen, aangezien de header, footer en sidebar op elke pagina dezelfde lay-out hebben.

Het ontdekken van de HTML elementen verantwoordelijk voor de lay-out kan op twee manieren;

- 'van binnen uit' - de code van de pagina's wordt geanalyseerd. Aan de hand van deze code wordt vastgelegd welke elementen hieruit verantwoordelijk zijn voor de layout.
- 'van buiten af' - In dit geval worden de pagina's geanalyseerd zoals ze op het scherm te zien zijn. Vervolgens wordt bepaald welke HTML elementen nodig zijn om dit te bereiken.

Er is gekozen voor de tweede manier, aangezien de pagina's in hun huidige vorm zeer complex zijn opgebouwd. Het is in eerste instantie heel moeilijk te zien hoe de pagina's zijn opgebouwd. Aangezien het hier alleen gaat om stijkenmerken (uiterlijk van de pagina's) is het heel goed mogelijk om de pagina's 'van buiten af' te analyseren. Er zijn ook veel elementen aanwezig die niet noodzakelijk zijn voor om de huidige lay-out te bereiken, als gevolg van vele aanpassingen en toevoegingen die steeds gedaan zijn aan de code. Op deze manier wordt bereikt dat die elementen die voor dit project niet van belang zijn uit de code verdwijnen. (de doelstelling is uiteindelijk ook een 'schonere' makkelijker onderhoudbare code)

Om een overzicht te krijgen van de betreffende HTML-elementen, is van elke lay-out structuur een pagina opnieuw in HTML uitgewerkt. Deze pagina's maken dus alleen gebruik van HTML en bevatten verder geen verschillende scripts. Op deze manier is dus zeker, dat het volledige uiterlijk het resultaat is van enkel HTML. Nu zijn dus ook de HTML-elementen verantwoordelijk voor die lay-out aan te wijzen.

Er zijn drie pagina's uitgewerkt in HTML;

- pagina met header, footer, sidebar en content lay-out 1
- pagina met header, footer, sidebar en content lay-out 2
- pagina met header, footer, sidebar en content lay-out 3

Uit de in HTML uitgewerkte pagina's is de volgende 'hoofdstructuur' af te lijden;

```

<html>
<body>
<form>
  <div>
    //DIV voor hoofdmenu-items
  </div>
  <div>
    //DIV voor submenu-items
  </div>
  <table>
  <!-- start of search menu -->
  <tr>
    <td rowspan=2 valign=top width=200>
      <table>
        <tr>
          <td>
            //hier wordt de sidebar weergegeven
          </td>
        </tr>
      </table>
    </td>
  <!-- end language select -->
  <!-- header -->
    <td>
      <table>
        <tr>
          <td>
            //hier wordt de header weergegeven
          </td>
        </tr>
      </table>
    </td>
  </tr>
  <!-- end header -->
  <!-- start content -->
  <tr>
    <td>
      <table>
        <tr>
          <td>
            //hier wordt een contentblok
            weergegeven
          </td>
        </tr>
      </table>
    </td>
  </tr>
</table>
  <!-- footer -->
  <table>
    <tr>
      <td>
        //hier wordt de footer weergegeven
      </td>
    </tr>
  </table>
  <!-- end footer -->
</form>
</body>
</html>

```

De document trees van alle delen samengevoegd levert de document tree op die geldig is voor een complete pagina. De volgende tree is geldig voor een pagina met lay-out 1;

```

<html>
  <body>
    <form>
      <div id=mainmenu>
        <table id=header_menu>
          <tr>
            <td class=menu_item_yellow>
              <div class=menu_item>
                <a class=menu_link>
            <td class=menu_item_yellow>
              <div class=menu_item>
                <a class=menu_link>
            <td class=menu_item_blue>
              <div class=menu_item>
                <a class=menu_link>
            <td class=menu_item_blue>
              <div class=menu_item>
                <a class=menu_link>
            <td class=menu_item_blue>
              <div class=menu_item>
                <a class=menu_link>
            <td class=menu_item_blue>
              <div class=menu_item>
                <a class=menu_link>
          </tr>
        </table>
      </div>
      <div id=submenu>
        <div class=submenu_list>
          <a class=submenu_link>
            <div class=submenu_item>
          <a class=submenu_link>
            <div class=submenu_item>
          ...
          etc.
        </div>
        <div class=submemenu_list>
          <a class=submenu_link>
            <div class=submenu_item>
          <a class=submenu_link>
            <div class=submenu_item>
          ...
          etc.      (voor elk submenu item is er een link)
        </div>
      </div>
      etc      (voor elk hoofdmenu item zijn er 1 of meer submenu items,
                gegroepeerd in een div)
    </form>
  </body>
</html>

```

```

        <td id=header>
            <div id=header_bg>
                <a class=img_link>
                    <img id=contact>

        <tr>
            <td id=content>
                <div id=content>
                    <div id=br_cr>
                        <a id=br_cr>
                            <font id=br_cr>
                    <div id=cont1_intro>
                        <h2 class=cont1_h>
                            <font class=cont1_txt>
                                <a class=cont1_link>
                                    <img class=cont1_img>
                    <div id=cont1_tabs>
                        <a class=tab_link>
                            <img class=tab_img>
                                <img class=cont1_tab_img>
                                    <font class=cont1_tab_txt>
                                        <a class=cont1_tab_link>

    <div id=footer>
        <a class=img_link>
            <img id=avery_logo>
        <font class=foot_links>
        <a class=foot_links>
        <font class=copyright>
        <a class=legal_links>

```

Bovenstaande document tree geldt voor een pagina met lay-out 1. Het vetgedrukte deel uit bovenstaande tree kan vervangen worden door de tabel van content van lay-out 2 of lay-out 3. De rest van de tree is voor deze lay-outs uiteraard hetzelfde, aangezien header, footer en sidebar voor alle pagina's dezelfde lay-out hebben.

Voor de document tree van lay-out 2 wordt het vetgedrukte deel vervangen door de document tree van het content deel van lay-out 2;

```
<div id=content>
  <div id=br_cr>
    <a id=br_cr>
      <font id=br_cr>
    <div id=cont2>
      <h2 class=cont2_h>
      <font class=cont2_intro>
      <font class=cont2_txt>
      <font class=cont2_txt_bld>
      <img class=cont2_img>
      <a class=cont2_link>
      <a class=cont2_but_link>
        <img class=cont2_but>
      <input class=cont2_inp>
      <select class=cont2_sel_in>
        <option class=cont2_sel_opt>
      <table class=cont2_table>
        <div>
          <font class=cont2_intro>
          <font class=cont2_txt>
          <font class=cont2_txt_bld>
          <img class=cont2_img>
          <a class=cont2_link>
          <a class=cont2_but_link>
            <img class=cont2_but>
          <input class=cont2_inp>
          <select class=cont2_sel_in>
            <option class=cont2_sel_opt>
```



Voor de document tree van lay-out 3 wordt het vetgedrukte deel vervangen door de document tree van het content deel van lay-out 3;

```

<div id=content>
  <div id=br_cr>
    <a id=br_cr>
    <font id=br_cr>
  <div class=cont3>
    <h2 class=cont3_h>
    <font class=cont3_txt>
    <a class=cont3_link>
    <img class=cont3_img>
  <div class=cont3>
    <h2 class=cont3_h>
    <font class=cont3_txt>
    <a class=cont3_link>
    <img class=cont3_img>
    <a class=cont3_but_link>
      <img class=cont3_but>
    <input class=cont3_inp>
    <select class=cont3_sel_in>
      <option class=cont3_sel_opt>
  <div class=cont3>
    <h2 class=cont3_h>
    ...
    etc.      (dit gedeelte kan dezelfde elementen bevatten als de
               voorgaande <div>)

  ...
  etc.      (afhankelijk van het aantal contentblokken kunnen hier meerdere
            divs volgen)

```

Voor een gedetailleerde omschrijving van de ontwikkeling van bovenstaande document trees wordt verwezen naar de volledige documentatie van de systeemanalyse.

### 3.5 Beschrijving management systeem voor menu

Voor het onderhouden van het menu van de website wordt een management systeem gebruikt in Lotus Notes. Dit is een applicatie die in de Lotus Notes omgeving draait en het mogelijk maakt het menu te van de hoofd- en sub-items te voorzien. In de volgende paragrafen zal omschreven worden wat de mogelijkheden van het management systeem zijn in een korte functionele omschrijving. Vervolgens zal in een technische omschrijving uitgelegd worden wat de globale technische structuur is van het systeem.

#### 3.5.1 Globale functionele omschrijving

Alle gegevens die nodig zijn voor de opbouw van het menu zijn opgeslagen in een database. Dit zijn bijvoorbeeld gegevens zoals teksten voor de hoofd- en submenu-items en paden voor links naar de verschillende pagina's

De volgende afbeelding is een screenshot van het startscherm van de applicatie.

	engels	fr	ge	it	sp	ch	pl
1 0	Home	Home	Home	Home	Inicio	Home	Home
2 0	Site Map	Plan	Übersicht	Mappa	Mapa	Mapa	Mapa
3 0	Products	Produits	Produkte	Prodotti	Productos	Výrobky	Produkty
3 1	Standard Products	Les Produits Star	Standardprodukte	Prodotti Standard	Productos Estándar	Standardní produk	Produkty
3 2	Specialty Products	Les Produits Spé	Spezialitäten	Prodotti Speciali	Productos Especit	Speciální produkty	Produkty
3 3	Non-adhesive Products	Les Produits non-	Nicht selbstkleben	Prodotti non adesivi	Productos No Auto	Nesamolepící výrob	Produkty
3 4	Product Datasheets	Fiches Techniques	Datenblätter	Guida ai Prodotti	Fichas Técnicas	Technický datalist	Dane techn
3 5	Product Samples	Les Echantillons	Produktmuster	Campionature	Muestras de Prodi	Vzorky výrobk	Próbki pro
3 6	Product Advice	Conseils Produits	Produkt-Ratgeber	Product Advice	Asesoramiento Pri	Doporučení výrobk	Product A
4 0	Services	Services	Services	Servizi	Servicios	Služby	Serwis
4 1	Service Programme	Le Programme d	Service-Programm	Programmi di serv	Programas del ser	Programy služeb	Program
5 0	News & Information	Les Informations	Informationen	Informazioni	Información	Zprávy & Informa	Informacja
5 1	Industry News	Les Informations	Neues aus der Ind	Notizie dall'Industri	Noticias de la Indu	Novinky z odvětví	Nowości p
5 2	Events	Les Evénements	Veranstaltungen	Eventi	Acontecimientos	Události	Targi&wy
5 3	Brochures	Les Brochures	Broschüren	Brochures	Folletos	Brojury	Broszury
5 4	Glossary of Terms	Glossaire	Facheausdrücke	Glossario dei Tern	Glosario de Térmi	Slovníček termín	Sownik te
5 5	About us	Concernant Fass	Über uns	A proposito di Noi	Sobre nosotros	O nás	O nas
5 6	Company Locations	Adresses de la S	Unternehmensstar	Società e Sedi	Company Locatior	Adresy společností	Lokalizac
5 7	Jobs	Carrières	Karriere	Lavoro	Trabajos	Zaměstnání	Praca
6 0	Transactions	Les Transactions	Transaktionen	Acquisti	Transacciones	Transakce	Transakc
6 1	Price List	Tarifs	Preisliste	Elenco prezzi	Tarifa	Ceník	Lista cen
6 2	Exact / Exact Plus List	Exact / Exact Plus	Liste Exact / Exact	Lista Prodotti Exac	Lista de Exact / Ex	Seznam Exact/Exe	Lista Exac
6 3	Ordering	Commander	Bestellung	Ordinazioni	Pedidos	Objednávka	Zamawia
6 4	Order Tracking	Etat des commar	Bestellungsverfolg	Rintracciabilità	Orc Consulta de Pedid	Sledování objedná	ledzenie
6 5	Mix & Match	Mix & Match	Mix & Match	Mix & Match	Mix & Match	Mix & Match	Mix & Mat
6 6	Fasson Outlet	Fasson Outlet	Fasson Outlet	Fasson Outlet	Fasson Outlet	Prodejna Fasson	Fasson O
6 7	Fasson Rapid-Roll	Fasson Rapid-Rv	Fasson Rapid-Rol	Fasson Rapid-Rol	Fasson Rapid-Rol	Fasson Rapid-Rol	Fasson R
6 8	RollXChange	RollXChange	RollXChange	RollXChange	RollXChange	RollXChange	RollXCha
6 9	Product Code Cross-ref	Les Codes Prodi	Produkt-Code	Qu Referenze e Codic	Referencias Cruza	Kód produktu/kilo	Kod wra
6 10	Product Code Changes	Modifications du	Änderungen der Pi	Product Code Cha	Cambios de Codic	Změny produktové	Zmiana k
6 11	Download Fasson Conne	Télécharger Fas	Download Fasson	Download di Fass	Descargar Fasson	Stáhnout spojení	Pobierz F

Afb. 18 – 'Startscherm' van de applicatie

In de linker kolom kunnen verschillende 'views' op de database gekozen worden. Rechts wordt het resultaat van de gekozen view getoond; een overzicht van een deel van de database. Via deze views kunnen verschillende elementen van de database bekeken worden. Door in het rechter scherm op een regel te dubbelklikken, kunnen details bekeken of bewerkt worden.

Een aantal belangrijke views zijn;

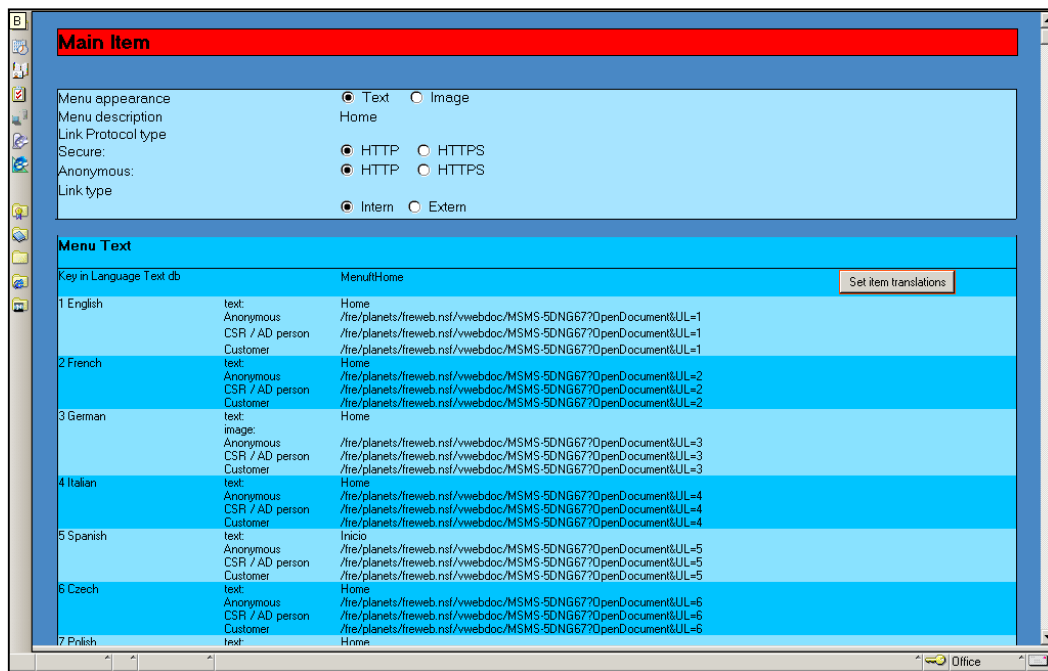
1. Menu, deze toont een compleet overzicht van alle menu-items met hun positie in het menu en de vertalingen van de tekst van de verschillende menu-items.
2. Files, dit is een verzameling van alle mogelijke JavaScript bestanden die gegenereerd kunnen worden en worden gebruikt bij de opbouw van het menu. Deze view toont een overzicht van deze bestanden, die vervolgens bekeken en bewerkt kunnen worden door te dubbelklikken op een van de bestanden.
3. JSFiles. Het systeem kan menu's genereren die compatible zijn met drie verschillende browsers (NS4, NS7 en IE). Deze view toont een overzicht van de 3 standaard JavaScript bestanden die gebruikt worden bij de opbouw van de menu's specifiek per browser. Voor elke browser wordt een apart script gebruikt.
4. GraphicFiles, een overzicht van alle afbeeldingen die mogelijk gebruikt kunnen worden in het menu.

In het screenshot is de eerste view geselecteerd en deze toont rechts een volledig overzicht van het huidige menu; elke regel vertegenwoordigt een hoofd- of submenu-item. In de eerste twee kolommen is aangegeven op welke plek het item in het menu moet komen te staan; de eerste kolom staat hierbij voor het hoofdmenu-item waar dit betreffende item bij hoort, de tweede kolom geeft aan op welke plek het betreffende item onder het hoofdmenu-item moet komen te staan. Positie 0 betekent hier bijvoorbeeld dat het om een hoofdmenu-item gaat, positie 1 geeft aan dat het item het eerste submenu-item onder het hoofdmenu-item is. In de kolommen ernaast staan alle gebruikte vertalingen van de tekst van het betreffende menu-item.

In de verschillende views van de database kan niets aangepast worden. Het geldt alleen als overzicht; vanaf hier kan verder geklikt worden naar schermen waar wel aanpassingen gedaan kunnen worden.

Om aanpassingen te doen voor een menu-item dubbelklikt men op de betreffende regel in de bovenstaande view. Om bijvoorbeeld het eerste menu-item aan te passen, dubbelklikt men op de eerste regel in de bovenstaande view. Het gaat in dit geval om de button 'Home'.

.....Men komt dan bij het volgende scherm terecht;



Afb. 19 – Interface voor het aanpassen van menu-items

Binnen dit overzicht kunnen alle aanpassingen voor een menu-item gedaan worden, in dit geval voor het menu-item 'Home'. In het bovenste blok kan men een aantal eigenschappen voor het menu-item opgeven;

- Menu appearance; is het een afbeelding of een tekstlink
- Menu description; een omschrijving van het menu-item
- Link Protocol type; HTTP of HTTPS
- Link type; interne of externe link

In het overzicht daaronder staan alle vertalingen van de tekst van het menu-item, in dit geval vertalingen van 'Home', met bijbehorende links. Voor elk item zijn drie links gedefinieerd; er worden namelijk drie verschillende gebruikers onderscheiden. (Anonieme gebruikers, interne gebruikers en klanten) Voor sommige menu-items is het zo, dat de links per gebruiker kunnen verschillen, vandaar dat elk menu-item drie links bevat.

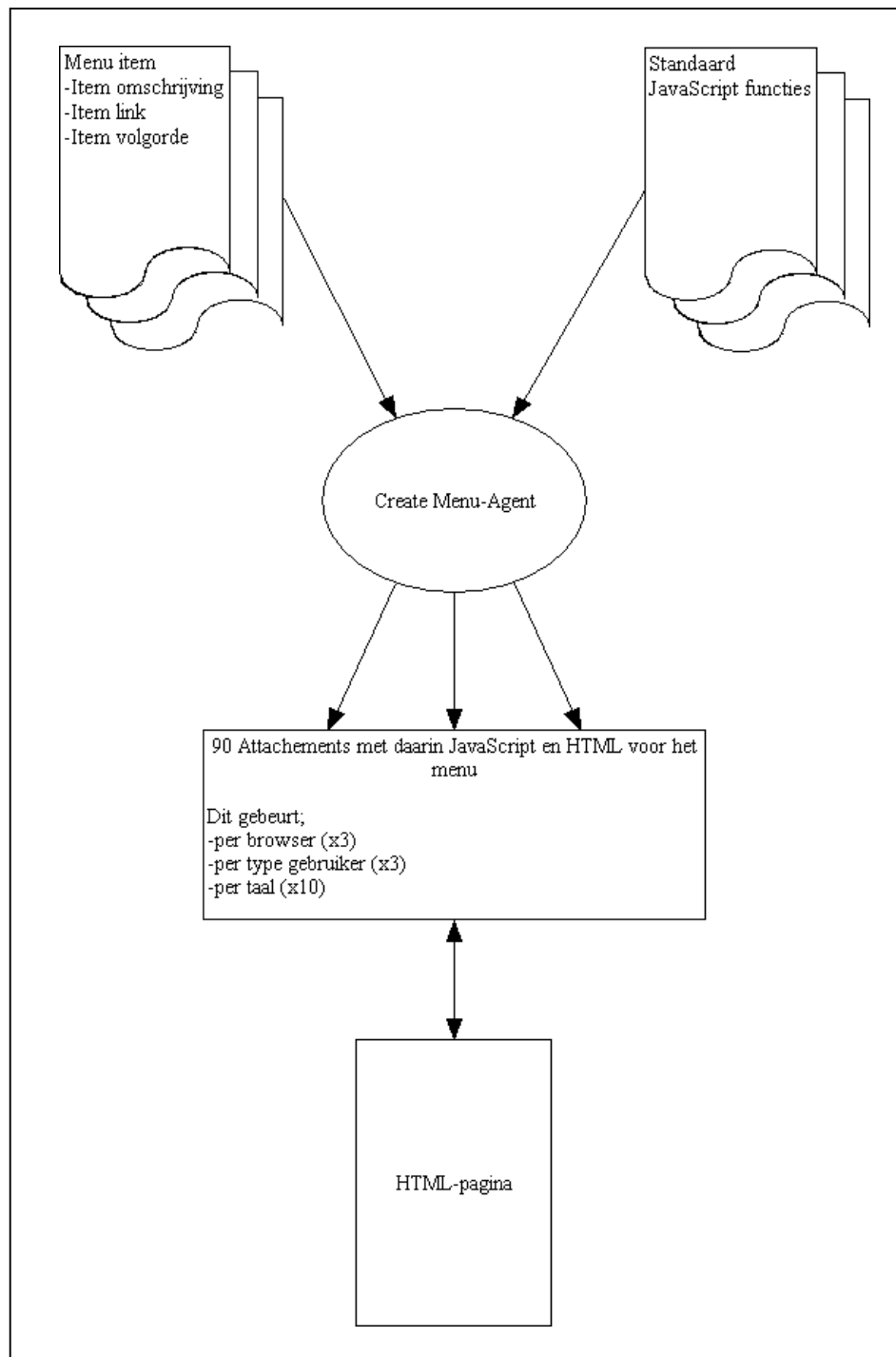
Bovenin het tweede overzicht kan een tekst ingevoerd worden voor het menu-item. Door op de button 'Set item translations' te klikken, worden alle vertalingen voor dat item uit een andere database gehaald en ingevuld.

Op deze manier kunnen dus heel snel menu-items aangepast worden. Opgemerkt moet worden, dat alleen teksten van menu-items, links en een aantal niet zichtbare eigenschappen aangepast kunnen worden. Stijlelementen, zoals bijvoorbeeld de kleur van de buttons, kunnen niet m.b.v. dit systeem worden aangepast. Via een 'omweg' kunnen enkele van deze zaken wel aangepast worden binnen het systeem, door de JavaScript bestanden die gebruikt worden aan te passen. Deze bestanden genereren namelijk de HTML-code, waaruit het menu is opgebouwd.

De applicatie bevat verder geen andere interfaces, zoals die op afbeelding 19. Op een vergelijkbare manier kunnen met de andere views wel andere records uit de database bekeken worden. Door weer te dubbelklikken op een record kunnen gegevens worden aangepast. Zo wordt bijvoorbeeld het JavaScript bestand getoond, wanneer men dubbelklikt op een record in de view JSFiles; alleen de code van het script wordt getoond en deze kan aangepast worden.

### 3.5.2 Globale technische omschrijving

De volgende afbeelding is een schematisch overzicht van de globale technische structuur van het management systeem voor het menu;



Afb. 20 – Schematisch overzicht technische structuur

De verschillende data die nodig is voor de opbouw van het menu is opgeslagen in een database, RMEPlanetsMenu.nsf. De HTML-code waaruit het menu opgebouwd is, wordt door een *agent* geschreven. Een agent is vergelijkbaar met een macro en kan gezien worden als een stand-alone applicatie, die een specifieke taak uitvoert in een of meer databases. Deze taak is in dit geval het verzamelen van de juiste gegevens uit de database en deze op een correcte manier te combineren in een document; het attachement met daarin JavaScript en HTML voor het menu. Er worden totaal 90 van deze attachements gegenereerd door de agent, welke weer worden opgeslagen in de database. Op de HTML-pagina wordt het juiste attachement aangeroepen, deze wordt ge-include in de betreffende pagina.

In het schema is te zien dat de agent twee soorten gegevens combineert.

Ten eerste wordt er gebruik gemaakt van de documenten die bij de view 'menu' horen (afbeelding 1). Dit zijn de gegevens, die de gebruiker van het systeem kan aanpassen voor een menu-item, zoals beschreven is in paragraaf 3.1. Hier gaat het om de inhoud van het menu, in het schema is af te lezen welke eigenschappen aangepast kunnen worden;

- Item omschrijving; dit is de tekst voor het menu-item
- Item link; dit is het pad waar het menu-item naar linkt. Dit kunnen verschillende paden zijn voor een enkel menu-item, als gevolg van verschillende gebruikersniveaus.
- Item volgorde; hiermee kan aangegeven worden op welke positie in het menu het betreffende item moet komen te staan.

De tweede soort gegevens die nodig zijn, zijn de standaard JavaScripts. Deze scripts zorgen ervoor, dat menu's voor drie verschillende browsers geschreven kunnen worden. Er zijn scripts voor de volgende browsers;

- Internet Explorer (versie 4 en hoger)
- Netscape 4.x
- Netscape 7

Op basis van deze twee soorten gegevens bouwt de agent alle mogelijke menu's en slaat de code hiervoor op in aparte documenten, welke opgeslagen worden in de database. Deze documenten worden vervolgens ge-include in de HTML-code van de website. Er worden menu's gemaakt;

- per browser (x3)                      (data standaard uit JavaScripts)
- per gebruiker (x3)                    (data uit documenten menu-item)
- per taal (x10)                         (data uit documenten menu-item)

## 4 Systeemeisen

In de volgende paragrafen worden de systeemeisen voor het nieuwe systeem besproken. Voor het nieuwe systeem zijn twee delen te onderscheiden; de nieuwe style sheet met de aangepaste HTML-pagina's, met daarnaast het bijbehorende management systeem. Bij de beschrijving van de eisen zullen beide onderdelen apart besproken worden. Per onderdeel zullen kort de volgende aspecten besproken worden;

- Doelgroepbeschrijving
- Basis systeemeisen
- Performance eisen
- Interface eisen
- Operationele eisen

### 4.1 Doelgroepbeschrijving

#### **Style sheet**

Doelgroep van dit systeem zijn de bezoekers van de website europe.fasson.com. Dit zijn zowel geregistreerde klanten als bezoekers. De geregistreerde klanten zijn regelmatige bezoekers van de site; zij zijn bekend met de huidige look & feel van de site.

#### **Management systeem**

De doelgroep van het management systeem bestaat uit medewerkers van de afdeling e-Business, de webcontent manager in het specifiek. Deze groep mensen heeft een gedegen kennis van het vakgebied en zijn dus bekend met verschillende aspecten van m.b.t. webdesign en de mogelijkheden en beperkingen van CSS en HTML. Het management systeem geldt voor de doelgroep als een tijdsbesparend middel en moet meer flexibiliteit bieden bij het aanpassen van uiterlijke kenmerken van de website.



## 4.2 Basis systeemeisen

### **Style sheet**

De invoering van de nieuwe style sheet zal de volgende gevolgen hebben;

- De huidige look & feel van de website blijft in eerste instantie onveranderd. Dit moet bereikt worden met een combinatie van HTML en CSS.
- Deze nieuwe structuur zal een zo 'schoon' mogelijke code opleveren. Zo zal bijvoorbeeld het grote aantal geneste tabellen, eerder aan bod gekomen bij de systeemanalyse, zoveel mogelijk verdwijnen. Er wordt gestreefd naar een enkele tabel; deze tabel is verantwoordelijk voor de verdeling op het scherm van de verschillende blokken. Het doel hiervan is het creëren van flexibiliteit m.b.t. de onderhoudbaarheid van het uiterlijk van de website.
- Ook de tweede verbeterde style sheet die ontwikkeld zal worden, zal diezelfde look & feel hebben. Hierbij zal echter gestreefd worden naar een meer consistent uiterlijk van de website.
- De ontwikkeling van de tweede style sheet richt zich vooral op het optimaliseren van het gebruik van CSS op de website. De geleerde CSS-technieken (zelfstudie tijdens dit project) zullen hiervoor zo efficiënt en effectief mogelijk toegepast worden, zodat optimaal gebruik gemaakt wordt van de mogelijkheden van CSS.
- Voor alle pagina's zal een printlay-out ontwikkeld worden. Hiervoor zullen aparte style sheets gebruikt worden.
- Alle style sheets zullen over de gehele website gebruikt worden.

### **Management systeem**

De pagina's van de website zijn opgebouwd uit verschillende HTML-elementen die bepalend zijn voor het uiterlijk van de website, zoals besproken is bij de systeemanalyse. Al deze elementen zullen te identificeren zijn aan de hand van toegekende classes en of unieke id's. Voor elk element zullen een aantal stijleigenschappen gedefinieerd worden in de style sheets.

Het management systeem kent de volgende basis systeemeisen;

- Het systeem moet tijd besparen bij het wijzigen van stijlenmerken
- Alle eigenschappen gedefinieerd in de style sheet voor de verschillende HTML-elementen, moeten d.m.v. het management systeem aan te passen zijn.
- Om de pagina's zo consistent mogelijk te houden wat stijl betreft, moet de keuze van op te geven waardes voor stijlattributen een zekere beperking krijgen. Wanneer een waarde gekozen kan worden, zal deze gekozen worden uit een beperkte lijst. Wanneer een waarde door de gebruiker ingevuld wordt, zal dit beperkt worden door een maximum en minimum waarde.
- Het systeem moet complete style sheets genereren, opgemaakt volgens de standaarden van het W3C, optimaal gebruik makend van de mogelijkheden van CSS.

## **4.3 Performance eisen**

### **Style sheet**

De ontwikkeling van de nieuwe style sheets is niet gericht op het verbeteren van de performance. Als enige performance eis geldt dan ook, dat de website in de nieuwe situatie geen tragere response tijden heeft dan in de huidige situatie. Dus om de performance van de website zo hoog mogelijk te houden zal gestreefd worden naar zo beknopt mogelijke style sheets.

### **Management systeem**

Hier kan gekeken worden naar het doel van het managementsysteem; tijd besparen bij het bewerken van de stijlenmerken van de website. De performance moet dus minimaal van zodanig niveau zijn, dat het uitvoeren van die taak sneller gaat dan in de huidige situatie.

Verder kan voor de performance m.b.t. responsetijden e.d. het managementsysteem voor het menu van de website als referentie gebruikt worden. Een beschrijving van dit systeem is gegeven in *par. 3.5 Beschrijving management systeem voor menu*.

#### 4.4 Interface eisen

##### **Style sheet**

De volgende interface eisen worden gedefinieerd voor de style sheet;

- De huidige look & feel blijft in eerste instantie onveranderd. De drie vastgestelde lay-outs zullen in de nieuwe versie ook nog bestaan.
- De verbeterde versie van de style sheet zal tot gevolg hebben, dat de huidige look & feel van de site nog steeds gewaarborgd blijft, alleen zal deze look & feel op een meer consistente manier toegepast worden.

##### **Management systeem**

Er moet rekening gehouden worden met het feit dat deze applicatie zal draaien binnen de Lotus Notes omgeving. De interface van deze applicatie zal zich als het ware binnen de interface van Lotus Notes bevinden.

De volgende interface eisen worden gedefinieerd voor het management systeem;

- De interface moet zo overzichtelijk mogelijk gehouden worden. Om dit te bereiken zal de informatie in delen worden aangeboden (er zal bijvoorbeeld gewerkt worden met categorieën) Verder zal de informatie niet te dicht op elkaar moeten staan, er moet gezorgd worden voor een goede spreiding.
- Er moet gebruik gemaakt worden van een heldere, eenvoudige navigatie.
- Navigatie moet verlopen via een stap-voor-stap methode. Geleidelijk navigeert de gebruiker naar het element dat bewerkt moet worden.
- Bij de ontwikkeling van de interface zal zoveel mogelijk rekening gehouden worden met de richtlijnen die gesteld worden door Guide (Graphical User Interface Design and Evaluation), zoals de regels opgesteld door Schneidermann.

#### 4.5 Operationele eisen

##### Style sheet

De website zal aan de volgende operationele eisen moeten voldoen;

- Als gevolg van het behoud van de look & feel zal ook de optimale schermresolutie bij de ontwikkeling van de eerste style sheet niet veranderen; de website is geoptimaliseerd voor een resolutie van 800 \* 600. Bij de tweede versie van de style sheet kan dit eventueel veranderen.
- De website dient compatible te zijn met de volgende browsers;
  - ✓ Microsoft Internet Explorer 5
  - ✓ Microsoft Internet Explorer 6
  - ✓ Netscape 7
- Alle pagina's moeten zowel door de CSS-validator als de HTML-validator van het W3C goedgekeurd worden. Dit zijn tools die valideren of de HTML- en CSS-documenten volgens de standaarden van het W3C zijn opgesteld.
- Alle pagina's van de site zullen gebruik maken van het DOCTYPE voor HTML 4.01, strict \*

\*Door de vele verschillen in de manier van renderen die door de verschillende oude browsers werd toegepast, is het heel lang niet mogelijk geweest websites te maken volgens één standaard. (Attributen zoals margins en paddings worden door verschillende browsers anders geïnterpreteerd) Nieuwere browsers (vanaf IE6 en Netscape 6) renderen wel volgens de standaarden die het W3C voorschrijft. Dit heeft als gevolg dat pagina's er in verschillende browsers compleet anders uit kunnen zien; de nieuwe browsers renderen strict volgens de regels die het W3C opgesteld heeft (*strict rendering*), oudere browsers zullen in veel gevallen een andere lay-out van de pagina laten zien, deze maken gebruik van *loose rendering*. De gekozen oplossing maakt gebruik van het DOCTYPE dat ieder document moet bevatten. Hierin kan aangegeven worden welke renderingmode door de browser gehanteerd moet worden. Een *loose rendering* wil zeggen dat de pagina gerendered wordt, rekening houdende met de bugs en eigenaardigheden van de oudere (versie 4) browsers. Een *strict rendering* wil zeggen dat de specificaties van het W3C voor HTML, CSS en andere lay-out talen zo goed mogelijk gevolgd wordt. Een declaratie van een DOCTYPE is in feite alleen een link naar een online document waarin het gedrag voor het renderen is vastgelegd. De volgende declaratie zal gebruikt worden;

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/REC-html40/strict.dtd">
```

##### **Management systeem**

Het management systeem zal aan de volgende operationele eis moeten voldoen;

- Het systeem zal draaien binnen de Lotus Notes omgeving, binnen de interface van Lotus Notes.

## 5 Systeemconcept

In de volgende paragrafen wordt het concept voor het nieuwe systeem besproken. Voor het nieuwe systeem zijn twee delen te onderscheiden; de nieuwe style sheets met de aangepaste HTML-pagina's en het bijbehorende management systeem. Voor beide delen zal apart een systeemconcept besproken worden. In de laatste paragraaf zal voor het gehele systeem een globaal objectmodel en globale technische structuur gegeven worden.

### 5.1 Systeemconcept style sheet

In de huidige situatie wordt de lay-out van een pagina voor het grootste deel bepaald door de code van die pagina. Er is wel een style sheet aanwezig maar deze wordt lang niet optimaal gebruikt. De meeste stijkenmerken zijn 'hard' in de HTML gecodeerd. Ook is de code erg rommelig geworden door de vele ad-hoc toevoegingen die er gedaan zijn in de loop der jaren.

In de nieuwe situatie zullen alle stijlelementen in de style sheets gedefinieerd zijn. De nieuwe pagina's zullen veel 'schonere' HTML bevatten. Zo zal het grote aantal geneste tabellen verdwijnen; uiteindelijk wordt er alleen nog maar gebruik gemaakt van de hoofdtabelstructuur, welke eerder besproken is bij de systeemanalyse (hoofdstuk 2). De verschillende blokken op de pagina (header, sidebar, footer en content) zullen ingesloten zijn door een <div>.

Het concept bestaat in feite uit twee stappen;

- Het schrijven van de stijlen in de style sheets.
- Het aanpassen van de HTML op de pagina's. Hierbij gaat het op het filteren van de overbodige HTML (zoals de geneste tabelstructuren en de hard gecodeerde stijkenmerken). Daarnaast worden aan de HTML-elementen classes en/of id's toegekend.

#### 5.1.1 Systeemconcept style sheet – een voorbeeld

Nu volgt een voorbeeld van de wijzigingen die gedaan zouden kunnen worden voor een deel van een pagina, om aan te tonen op welke manier de pagina's aangepast zullen worden. De bestaande HTML-code zal gewijzigd worden, zodat deze geen attributen meer bevat, die stijkenmerken definiëren. Ook zal de bestaande tabelstructuur verdwijnen; in plaats van deze structuur zal gebruik gemaakt worden van DIVs. Vervolgens zal een deel van een (mogelijke) style sheet gegeven worden, om aan te tonen welke oplossingen gekozen zouden kunnen worden, om dezelfde lay-out als in de oude situatie te bereiken.

Volgende code is een fragment van een code uit een van de pagina's van de website en toont de HTML, die in de huidige situatie gebruikt wordt om de footer weer te geven;

```
<TABLE cellSpacing=0 cellPadding=0 width=600 align=left border=0>
<TBODY>
<TR>
<TD vAlign=top width=153><A href="http://www.averydennison.com/">
<IMG height=48 alt=Avery src="Avery Dennison Roll Materials Europe
Homepage_files/ADRMLogo.gif" width=153 border=0></A>
</TD>
<TD vAlign=top width="90%">
<TABLE cellSpacing=0 cellPadding=0 width=600 align=left border=0>
<TBODY>
<TR>
<TD class=link10px vAlign=top align=left width=600>

<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Home</A> |
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Products &
Services</A> |
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Market Segments</A> |
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>News &
Information</A> |
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Transactions</A> |
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Contact Us</A> |
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Search</A> |
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Site Map</A> |
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Investor Relations</A> |
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>AveryDennison.com</A>

</TD>
</TR>
<TR>
<TD vAlign=top align=left width=600 height=10>
<IMG height=10 src="Avery Dennison Roll Materials Europe
Homepage_files/space.gif" width=1>
</TD>
</TR>

<TR>
<TD class=copyright vAlign=top align=left width=600>
© 1996-2003 Avery Dennison Corporation. All rights reserved.&nbsp;
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Disclaimer</A>&nbsp;
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Legal Notice</A>&nbsp;
<A class=link10px
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Privacy Statement</A>
<BR><BR>
</TD>
</TR>
</TBODY>
</TABLE>
</td>
</tr>
</tbody>
</table>
```

Alle vetgedrukte code in bovenstaand fragment kan in de nieuwe versie weggelaten worden; dit zijn allemaal stijlelementen die in de nieuwe situatie in de style sheet vastgelegd zijn. Ook zal de tabelstructuur voor dit deel niet meer nodig zijn, evenals de pixelgrote afbeelding space.gif, waarmee ruimte gecreëerd wordt tussen de elementen. De code in de nieuwe situatie zal vergelijkbaar zijn met het volgende fragment;

```
<div id=footer>
  <A href="http://www.averydennison.com/" class=img_link>
  <IMG alt=Avery src="Avery Dennison Roll Materials Europe
Homepage_files/ADRMElogo.gif" id=avery_logo>
</A>
  <font class=foot_links>
    <A class=foot_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Home</A> |
    <A class=foot_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Products &
Services</A> |
    <A class=foot_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Market Segments</A> |
    <A class=foot_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>News &
Information</A> |
    <A class=foot_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Transactions</A> |
    <A class=foot_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Contact Us</A> |
    <A class=foot_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Search</A> |
    <A class=foot_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Site Map</A> |
    <A class=foot_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>Investor Relations</A> |
    <A class=foot_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc>AveryDennison.com</A>
  </font>
  <font class=copyright>
  <br>© 1996-2003 Avery Dennison Corporation. All rights reserved.&nbsp;
  </font>
  <A class=legal_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Disclaimer</A>&nbsp;
  <A class=legal_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Legal Notice</A>&nbsp;
  <A class=legal_links
href="http://www.europe.fasson.com/FRE/planets/freweb.nsf/etc">Privacy Statement</A>
</div>
```

In bovenstaande code is te zien, dat de tabelstructuur vervangen is door een <div>, waarbinnen zich alle elementen voor de footer bevinden. Er zijn geen attributen meer in de HTML-code, die betrekking hebben op lay-out van de pagina; deze attributen worden allemaal in de style sheet gedefinieerd. Door de juiste stijlen te schrijven voor de verschillende elementen van het fragment, kan dezelfde lay-out bereikt worden.

De nieuwe code in combinatie met bijvoorbeeld de volgende stijlen, leveren dezelfde lay-out van de footer op;

```

Div#footer {
    width: 753px;}

Img#avery_logo {
    width: 153px;
    height: 48px;
    vertical-align: top;
    border: 0;
    float: left;}

Font.foot_links{
    font-size: 10px;
    color: #3366cc;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

Font.copyright {
    line-height: 3.5em;
    font-size: 10px;
    color: #000000;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

A.foot_links:link {
    font-size: 10px;
    color: #3366cc;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

A.foot_links:hover {
    font-size: 10px;
    color: #ff0000;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

A.foot_links:visited {
    font-size: 10px;
    color: #3366cc;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

A.legal_links:link {
    font-size: 10px;
    color: #3366cc;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

A.legal_links:hover {
    font-size: 10px;
    color: #ff0000;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

A.legal_links:visited {
    font-size: 10px;
    color: #3366cc;
    font-family: Verdana, Arial, Helvetica, sans-serif;}

```

Bovenstaand fragment zou onderdeel kunnen zijn van de nieuwe style sheet. In de nieuwe situatie zal deze style sheet gegenereerd worden door het management systeem. Op de pagina zelf wordt vervolgens de style sheet ge-include.



In eerste instantie lijkt dit een omslachtige manier om uiteindelijk op dezelfde lay-out terug te komen. Met de nieuwe structuur gaan echter wel een aantal belangrijke voordelen gepaard. De belangrijkste voordelen zijn;

- De structuur van het document wordt schoner en dus overzichtelijker. Het gevolg hiervan is, dat het document makkelijker aan te passen is.
- Wijzigingen in stijlenmerken kunnen op deze manier veel sneller en makkelijker doorgevoerd worden. In de oude situatie is het nodig om alle pagina's afzonderlijk aan te passen, wanneer het gaat om een stijlenmerk dat hard is gecodeerd op de pagina. Deze stijlenmerken zullen in de nieuwe situatie gedefinieerd zijn in de style sheet; nu hoeft alleen de stijl voor de betreffende class veranderd te worden in de style sheet.

### *5.1.2 Style sheets voor printlay-out*

Onderdeel van de opdracht is het ontwikkelen van een printlay-out voor elke pagina. Het is echter niet nodig om van elke pagina een tweede versie te ontwikkelen, welke dan als printversie zou kunnen voldoen. CSS biedt de mogelijkheid om stijlen te declareren, die beschrijven hoe de HTML-elementen eruit moeten zien op papier. Op een style sheet kan nl. aangegeven worden, voor welk medium de stijldeclaraties bedoeld zijn. Er zullen dan ook twee style sheets ontwikkeld worden; een style sheet die beschrijft hoe de elementen gerendered moeten worden voor presentatie op het scherm en een style sheet die beschrijft hoe deze elementen eruit moeten zien op papier. Beide style sheets zullen op de pagina's aangeroepen worden. Er zijn verder geen extra aanpassingen in de HTML nodig; een aparte style sheet voor het printen is voldoende om een printlay-out te verwezenlijken.

De stijldeclaraties van de style sheet voor het printen hebben geen effect op de presentatie van de elementen op het scherm. Andersom, wanneer een pagina afgedrukt wordt, hebben de stijldeclaraties van de style sheet voor het scherm geen invloed op de presentatie van de elementen op papier.

## 5.2 *Systeemconcept management systeem*

Om te bereiken dat de nieuwe style sheets gemakkelijk onderhouden kunnen worden, wordt een management systeem voor de style sheets ontwikkeld. Met dit systeem zal het mogelijk zijn wijzigingen aan te brengen in de gebruikte style sheets.

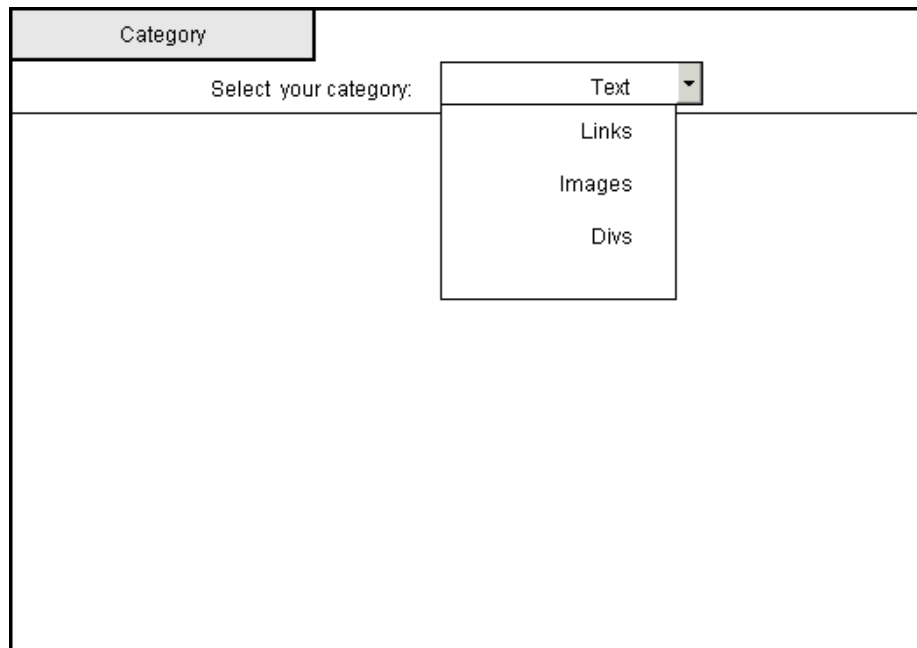
Voor de verschillende HTML-elementen zijn meerdere stijlattributen gedefinieerd in de style sheets. Met het systeem kunnen de waarden van deze stijlattributen gewijzigd worden. Vervolgens wordt door een agent (onderdeel van het management systeem) een nieuwe style sheet geschreven met de nieuw gedefinieerde waarden. Het systeem is vooral bedoeld als tijdbesparend middel voor de medewerkers van de afdeling e-Business bij het onderhoud van de website. Verder zal het systeem de flexibiliteit m.b.t. het aanpassen van uiterlijke kenmerken van de website verhogen. Het systeem zal complete style sheet-documenten genereren, die op de pagina ge-include worden.

Het systeemconcept voor het managementsysteem zal besproken worden a.d.h.v. een globaal storyboard, opgesteld in samenwerking met de webcontent-manager van de afdeling e-Business.

### 5.2.1 Globaal storyboard management systeem

Voor het wijzigen van uiterlijke kenmerken zullen globaal de volgende stappen doorlopen moeten worden;

1. De gebruiker kiest een categorie; deze is te selecteren in een drop-down-list-box (ddlb). De verschillende HTML-elementen waarvoor stijlen gedefinieerd kunnen worden, zijn ingedeeld in verschillende categorieën. (bijvoorbeeld tekst, links, afbeeldingen, etc.)



Afb. 21 – Schets van de interface van het management systeem

De gebruiker kiest een categorie, bijvoorbeeld 'links'.

2. De gebruiker krijgt nu twee nieuwe ddlb's te zien, met in de eerste een overzicht van de HTML-elementen, die op de site voorkomen, uit de gekozen categorie. De tweede ddlb bevat de mogelijke media waarvoor stijlen geschreven kunnen worden. In dit geval zal in de eerste ddlb een overzicht gegeven worden van alle soorten links, die voor kunnen komen op de site. Deze links zijn te identificeren a.d.h.v. een class of een id.

Category		
Select your category:	Links	
Links		
Select your type:	Links in sidebar	for: Screen
	Links in footer	Printing
	Links in content 1	
	Links in content 2	

Afb. 22 – Schets van de interface van het management systeem

De gebruiker selecteert de gewenste class of id van het betreffende element uit de ddlb. Wanneer de gebruiker bijvoorbeeld het uiterlijk van de links in de footer wil veranderen, wordt er geklikt op 'Links in footer'. Hiermee wordt de juiste selector geselecteerd, in dit geval a.footer. In de ddlb ernaast kan aangegeven worden voor welk medium stijlen aangepast moeten worden; voor printen of voor het scherm.

3. Er wordt nu een overzicht gegeven van de stijlattributen die bij de gekozen selector en medium horen. In het geval van links, zijn dit vier overzichten, aangezien een link vier statussen kan hebben (link, hover, visited en active)

The interface is a web-based management system for CSS styles. It features a top navigation bar with a 'Category' dropdown set to 'Links'. Below this, a 'Select your category:' dropdown is also set to 'Links'. The next row shows 'Select your type:' set to 'Links in footer' and 'for:' set to 'Screen'. The main content area is titled 'Links in footer for SCREEN' and contains a 'Save' button. Below the title, there are four panels representing different states of the 'a.foot\_links' selector: 'link', 'hover', 'visited', and 'active'. Each panel displays a list of attributes (font-size, color, font-family, background) with their current values and input fields for modification. For example, 'font-size' is set to '10 px' in all panels. The 'color' and 'background' attributes have color pickers. The 'font-family' attribute has a text input field. The 'visited' and 'active' panels only show 'font-size' in this view.

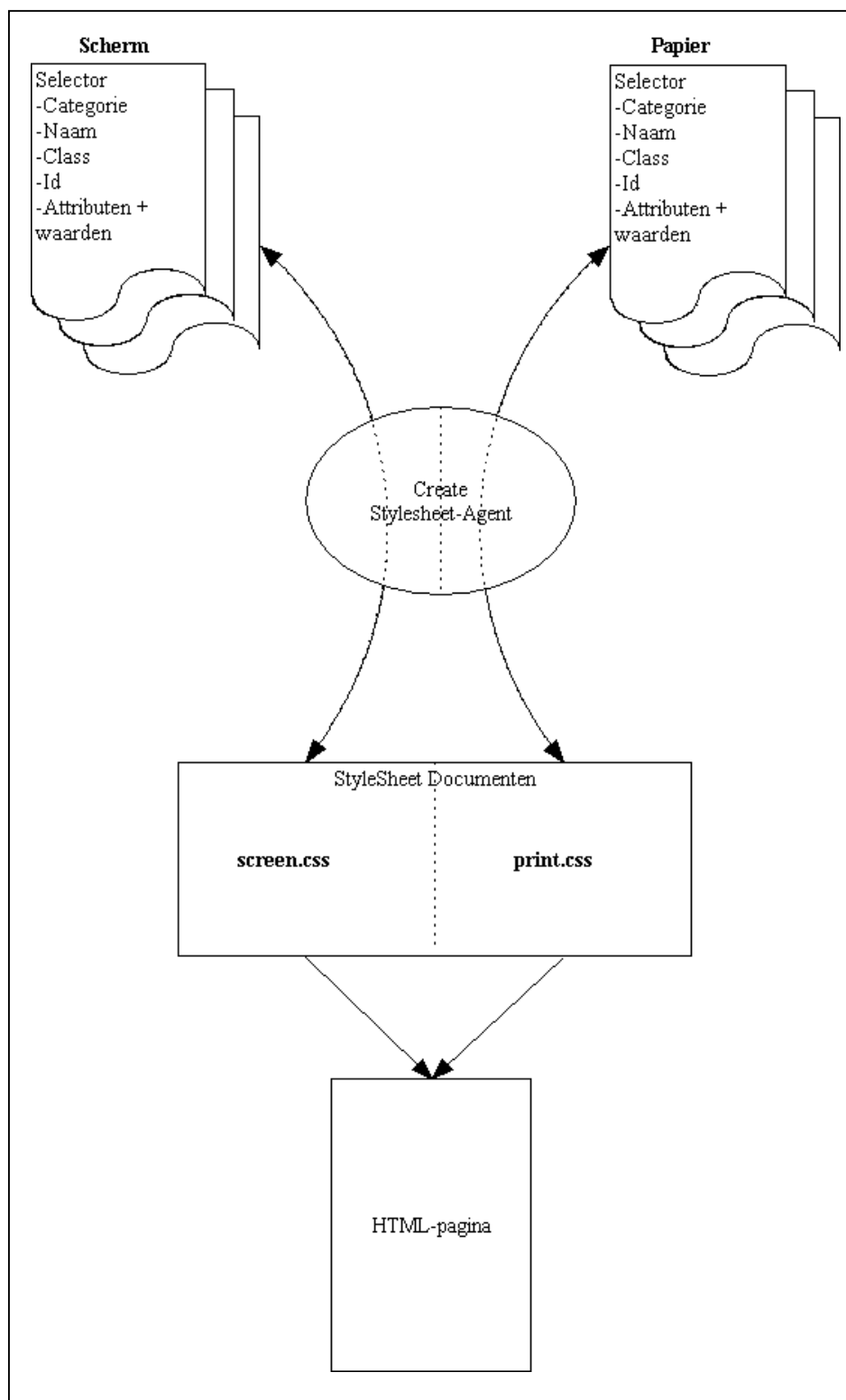
Afb. 23 – Schets van de interface van het management systeem

4. De gebruiker kan nu in bovenstaand scherm voor de stijlattributen van de gekozen selector de verschillende waardes opgeven. Dit gebeurt, afhankelijk van het stijlattribuu, d.m.v. een selectie uit een ddlb, een tekst-invoerveld of eventueel andere invoermogelijkheden.

Wanneer de gebruiker alle waardes heeft opgegeven, kunnen de waardes opgeslagen worden. De gebruiker klikt op de button 'Save'. (Nu wordt door een agent nieuwe style sheets gebouwd, met de zojuist ingevoerde waardes)

### 5.3 Globale technische omschrijving

De volgende afbeelding is een schematisch overzicht van de globale technische structuur van de nieuwe situatie;



Afb. 24 – Schematisch overzicht technische structuur

In de gebruikte database zijn voor elke selector twee documenten opgeslagen met daarin alle gegevens die nodig zijn voor die betreffende selector;

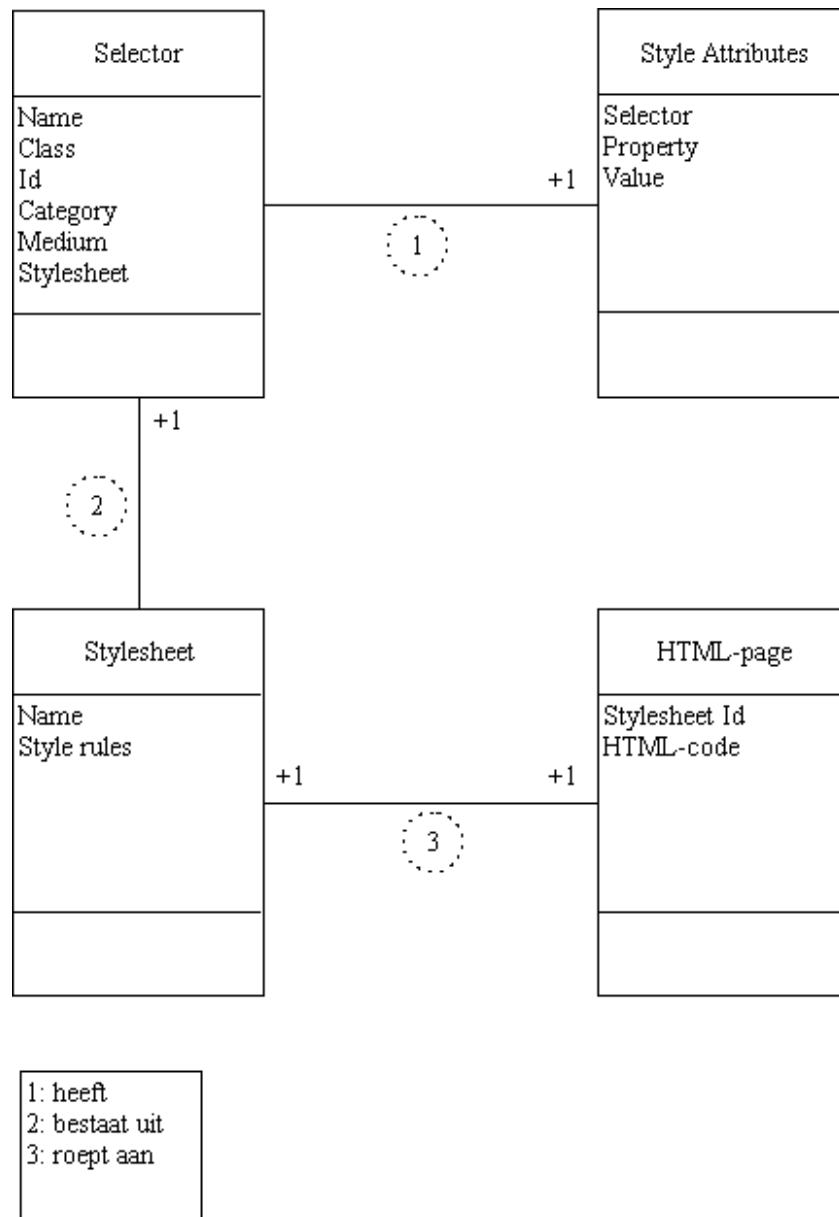
- Categorie; bijvoorbeeld tekst, links, afbeeldingen etc.
- Naam; bijvoorbeeld *font* voor tekst of *a* voor links
- Class en/of id; dit is om de selector uniek te kunnen identificeren (class of id is onderdeel van de selector)
- Attributen en waarden; dit zijn de eigenlijke stijldeclaraties, bijvoorbeeld *font-size: 10px;*.

Er zijn twee documenten per selector nodig; een document met stijldeclaraties voor presentatie op het scherm en een document met stijldeclaraties voor presentatie op papier.

De agent verzamelt de documenten en stelt deze samen tot een kloppend style sheet document. Ook dit gebeurt twee maal; van de documenten voor het scherm wordt een style sheet gemaakt voor het medium *screen*, van de documenten voor papier wordt een style sheet gemaakt voor het medium *print*. Vervolgens worden beide style sheet documenten op de (aangepaste) pagina's aangeroepen.

### 5.4 Globaal objectmodel

Voor het gehele systeem is het volgende globale objectmodel opgesteld;





## 6 Technische structuur

### 6.1 Ontwikkelomgeving

De ontwikkelomgeving van het hele project is Lotus Notes Domino. Het operating system waarop gewerkt zal worden is Microsoft Windows 2000 professional. Verder zal gebruikt gemaakt worden van Lotus Domino servers voor het ontwikkelen en testen van de nieuwe style sheets en het management systeem.

Verder wordt er gebruik gemaakt van de volgende technieken;

- HTML
- CSS (CSS2.0)
- Lotus Script
- Lotus Notes Formula Language

### 6.2 Te gebruiken hard- en software

Er wordt gewerkt op een Dell werkstation (pentium IV, 512 MB SDRAM) met de volgende software;

- Lotus Notes 5
- Microsoft Internet Explorer 6.0
- Netscape Communicator 4.7 / 7.1
- AceHTML 5 Pro

Ook zal er getest worden op een I-Mac, met Microsoft Internet Explorer 5.2.

Er zal niet *ontwikkeld* worden voor Netscape 4.7. Het is echter wel gewenst te weten in hoeverre de pagina's correct getoond worden en waar ongeveer de knelpunten liggen. Dit zal kort gedocumenteerd worden.

## 7 Pilotplan

Het nieuwe systeem bestaat uit twee hoofdonderdelen;

- De style sheets en aangepaste HTML-pagina's
- Het managementsysteem met bijbehorende database

Deze delen zullen opgedeeld worden in pilots die onafhankelijk van elkaar ontwikkeld zullen worden. In de volgende paragrafen zal aangegeven hoe de twee onderdelen opgedeeld zijn in pilots. Ook zal een korte beschrijving gegeven worden per pilot, samen met een schatting van de benodigde tijd en planning voor het ontwikkelen van de betreffende pilot.

### 7.1 Onderverdeling van de pilots

De volgende pilots vormen samen het totale systeem;

- Eerste Style sheet + aangepaste HTML-pagina's (huidige lay-out)
- Eerste Style sheet voor printen (huidige lay-out)
- Tweede Style sheet + aangepaste HTML-pagina's
- Tweede Style sheet voor printen
- Database voor het management systeem
- Management systeem voor de style sheets

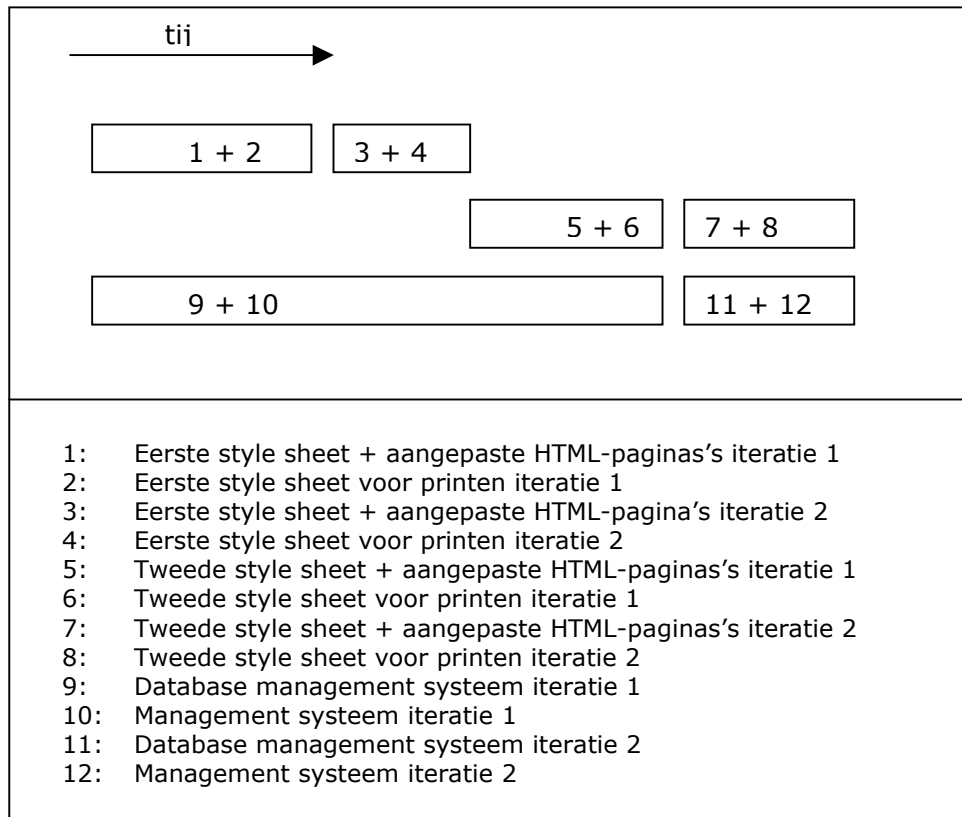
### 7.2 Pilots en iteraties

Elke pilot zal in twee iteraties ontwikkeld worden. Delen zullen serieel ontwikkeld worden; zo zal eerst de eerste versie van de style sheets ontwikkeld worden, voor zowel de presentatie op het scherm als de presentatie op papier (printversie). Vervolgens zullen deze pilots geëvalueerd worden. Deze evaluatie geldt als input voor de ontwikkelingen van deze pilots in de tweede iteratie.

Vervolgens worden de tweede versie van de style sheets ontwikkeld. Voor deze ontwikkeling wordt dezelfde indeling toegepast wat betreft iteraties als bij de ontwikkeling van de eerste versie.

Parallel aan bovenstaande ontwikkelingen zal het management systeem ontwikkeld worden met bijbehorende database. Ook deze ontwikkeling is opgedeeld in twee iteraties, waarbij de uitkomsten van de evaluatie van iteratie 1 weer gelden als input voor de ontwikkelingen in iteratie 2.

In onderstaande overzicht is de verdeling van pilots en iteraties schematisch weergegeven;



### 7.3 Beschrijving per pilot

In de volgende paragrafen zal per pilot een korte beschrijving gegeven worden, evenals een schatting voor de benodigde ontwikkeltijd en een planning.

#### 7.3.1 Eerste style sheet & aangepaste HTML-pagina's

Als eerste pilot zal de eerste style sheet ontwikkeld worden. Deze style sheet zal ervoor zorgen dat de website de huidige lay-out behoudt. Ook zullen de HTML-pagina's aangepast worden, zoals nader besproken is in het systeemconcept voor dit deel van het systeem, zie *par. 5.1 Systeemconcept style sheet – een voorbeeld*.

Er is tijdens de systeemanalyse geconstateerd, dat er 3 verschillende lay-outs zijn. Voor elke lay-out zal een aantal pagina's aangepast worden. De aanpassingen aan de overige pagina's valt buiten dit project; tijdens de implementatiefase zullen alleen de aangepaste pagina's geïmplementeerd worden. Alle andere pagina's zullen binnen een ander project aangepast worden; hiervoor zal documentatie opgesteld worden a.d.h.v. de aanpassingen die gedaan zijn aan de HTML-pagina's tijdens dit project.

De benodigde tijd voor iteratie 1 zal ongeveer 2 weken bedragen. Iteratie 2 zal, afhankelijk van de benodigde hoeveelheid wijzigingen, slechts een maximum van een aantal dagen in gaan nemen.

#### *Iteratie1*

Week 44
<ul style="list-style-type: none"> <li>- Ontwikkelen eerste style sheet</li> <li>- Aanpassen van de HTML-pagina's</li> </ul>

#### *Iteratie2*

Week 46
<ul style="list-style-type: none"> <li>- Finetunen van de style sheet</li> <li>- Finetunen van de HTML-pagina's</li> </ul>

#### *7.3.2 Eerste style sheet voor printen*

De style sheet voor het printen beschrijft hoe bepaald elementen op papier gepresenteerd moeten worden. Ook kan in deze style sheet aangegeven worden welke delen van de pagina afgedrukt worden en welke delen weggelaten kunnen worden. Zo zal bijvoorbeeld het menu niet uitgeprint worden, wanneer een bezoeker een pagina uitprint.

Deze pilot bestaat alleen uit de ontwikkeling van de style sheet; aanpassingen aan de HTML-code zullen niet tot nauwelijks nodig zijn (wellicht enkele toevoegingen van classes).

De benodigde tijd voor iteratie 1 zal maximaal 1 week bedragen. Iteratie 2 zal, afhankelijk van de benodigde hoeveelheid wijzigingen, slechts ongeveer 1 dag in gaan nemen.

*Iteratie1*

Week 45
<ul style="list-style-type: none"> <li>- Ontwikkelen eerste style sheet voor printen</li> <li>- Evt. kleine aanpassen aan de HTML-pagina's</li> </ul>

*Iteratie2*

Week 46
<ul style="list-style-type: none"> <li>- Finetunen van de style sheet</li> <li>- Finetunen van de HTML-pagina's</li> </ul>

*7.3.3 Tweede style sheet & aangepaste HTML-pagina's*

De ontwikkeling van de tweede (verbeterde) style sheet richt zich vooral op het optimaliseren van het gebruik van CSS. De geleerde CSS-technieken (zelfstudie tijdens dit project) zullen hiervoor zo efficiënt mogelijk toegepast worden, zodat optimaal gebruik wordt gemaakt van de mogelijkheden van CSS. Zo zal bijvoorbeeld onderzocht worden hoe *inheritance*-eigenschappen van CSS op de pagina's van de website toegepast kunnen worden.

Bij de ontwikkeling van de tweede style sheet zal gestreefd worden naar behoud van look & feel van de pagina's. In overleg met het e-Business team zal een geschikte lay-out overeengekomen worden.

De benodigde tijd voor iteratie 1 zal maximaal 2 weken bedragen. Iteratie 2 zal, afhankelijk van de benodigde hoeveelheid wijzigingen, maximaal 1 week in gaan nemen.

*Iteratie1*

Week 47/48
<ul style="list-style-type: none"> <li>- Ontwikkelen tweede style sheet</li> <li>- Aanpassen van de HTML-pagina's</li> </ul>

*Iteratie2*

Week 49
<ul style="list-style-type: none"> <li>- Finetunen van de style sheet</li> <li>- Finetunen van de HTML-pagina's</li> </ul>

*7.3.4 Tweede style sheet voor printen*

Deze pilot bestaat alleen uit de ontwikkeling van de verbeterde style sheet voor het printen; aanpassingen aan de HTML-code zullen niet tot nauwelijks nodig zijn (wellicht enkele toevoegingen van classes). Hierbij gelden dezelfde voorwaarden als bij de ontwikkeling van de tweede style sheet voor presentatie op het scherm (*par. 6.3.3*); optimaal gebruik maken van de mogelijkheden van CSS.

De benodigde tijd voor iteratie 1 zal maximaal 1 week bedragen. Iteratie 2 zal, afhankelijk van de benodigde hoeveelheid wijzigingen, slechts ongeveer 1 dag in gaan nemen.

*Iteratie1*

Week 48
<ul style="list-style-type: none"> <li>- Ontwikkelen tweede style sheet voor printen</li> <li>- Evt. kleine aanpassen aan de HTML-pagina's</li> </ul>

*Iteratie2*

Week 49
<ul style="list-style-type: none"> <li>- Finetunen van de style sheet</li> <li>- Finetunen van de HTML-pagina's</li> </ul>

### 7.3.5 Database voor management systeem

Deze pilot bestaat uit de ontwikkeling van de database, die gebruikt zal worden voor het management systeem waarmee de style sheets onderhouden kunnen worden.

De benodigde tijd voor iteratie 1 zal ongeveer 1 week bedragen. Iteratie 2 zal, afhankelijk van de benodigde hoeveelheid wijzigingen, slechts maximaal een aantal dagen in gaan nemen.

#### *Iteratie1*

Week 45
<ul style="list-style-type: none"><li>- Ontwikkelen Databasetabellen</li><li>- Ontwikkelen Relaties</li></ul>

#### *Iteratie2*

Week 46
<ul style="list-style-type: none"><li>- Finetunen Databasetabellen</li></ul>

### 7.3.6 Management systeem voor style sheets

Deze pilot bestaat uit de ontwikkeling van het management systeem, waarmee de style sheets onderhouden kunnen worden. Het systeem moet de mogelijkheid geven om alle stijlen, die gedefinieerd zijn in de style sheets, aan te passen, voor zowel de presentatie op het scherm als de presentatie op papier (style sheet voor printen). Het systeem zal complete kloppende style sheet documenten genereren.

Voor en tijdens deze ontwikkeling zal zelfstudie gedaan worden voor systeem ontwikkeling in Lotus Notes. Vandaar dat de benodigde tijd voor iteratie 1 geschat wordt op ongeveer 4 weken. Iteratie 2 zal, afhankelijk van de benodigde hoeveelheid wijzigingen, maximaal 1 week in gaan nemen.

#### *Iteratie1*

Week 45/46/47/48
<ul style="list-style-type: none"> <li>- Zelfstudie ontwikkelen in Lotus Notes</li> <li>- Ontwikkelen interface</li> <li>- Ontwikkelen van code (agents e.d.)</li> </ul>

#### *Iteratie2*

Week 49
<ul style="list-style-type: none"> <li>- Finetunen interface</li> <li>- Finetunen code</li> </ul>



---

## **Bijlage B – Aanbevelingen en Script**

Volledig gebruik van CSS op europe.fasson.com

Dennis van den Hoek  
Avery Dennison/RME – ASD  
08-01-04

---

## Inhoudsopgave

<b>1 Best practices in using CSS – Recommendations.....</b>	<b>163</b>
Using inheritance.....	163
Working with DIV's .....	163
Table structure VS Positioning of DIV's.....	164
Displaying links as block-level elements .....	164
Fixing bugs in Netscape 4.x .....	164
Adapt to screen size and resolution.....	165
Achieve consistency regarding the use of styles .....	165
<b>2 Script for modifying HTML in Notes databases .....</b>	<b>166</b>
General Directives .....	166
Linking the CSS.....	167
Adding Doc Type .....	169
Modifying the table-structure .....	170
Modifying the header and sidebar .....	171
Modifying the footer.....	180
Modifying the content (layout 2).....	190

Deze bijlage bevat de aanbevelingen met betrekking tot de *best practices* voor de website europe.fasson.com en het script, dat geschreven is voor het wijzigen van de HTML in de overige pagina's, die niet tijdens dit project aan bod gekomen zijn. Deze documentatie is ook in deze vorm aan de opdrachtgever aangeboden en is derhalve in het Engels.

## 1 Best practices in using CSS – Recommendations

### *Using inheritance*

Any HTML page consists of a number of elements - headings, paragraphs, lists, and so on. Every element is contained by another element, and every element itself can contain other elements. This is called the containment hierarchy of a web page. This property is the main reason why document trees were developed for the website. This way it became clear which elements a page consists of and how these elements are related to one another.

At the top of the containment hierarchy is the `<html>` element of the page. Every other element on a web page is contained within the `<html>` element, or one of the elements contained within it, and so on. Similarly, many elements will be contained in paragraphs, while paragraphs are contained in the `<body>`.

With CSS elements often (and with CSS2 can always be forced to) inherit properties from the elements which contain them (also known as their parent elements). This means that if the body of the page has certain style properties written for it in the style sheet (for example font and color), every element within the page (if it is a child of the `<body>` element) will inherit these properties; there is no need to set the font and color again for each element. You can always override the inheritance however. By assigning a property to an element, you override the inherited property.

This method is already used in the first version of the style sheet. For the new version it can be researched how this method can be applied more effectively, so that the website will use inheritance as effective as possible.

### *Working with DIV's*

In the modified HTML the complex table structures which are used to separate different parts of the pages and to position them correctly are omitted. In the new HTML only one simple tablestructure will be used and the different parts of the pages will be enclosed by their own DIV's, e.g. the footer will be enclosed by it's own table cell, within that cell different elements will be enclosed by their own DIV's. By doing this you gain a lot more flexibility in terms of styling different parts of the pages. Now you can actually write styles for specific parts of the pages.

In the first version of the style sheet this method is already used. The different parts of the website (header, footer, sidebar and content) all make use one simple table structure, in which different DIV's are placed. These DIV's can all be styled seperately. It will be usefull to research how this method can be optimized for the website. More DIV's should be used, so that even more flexibility is created. When all elements are enclosed by their own DIV, another 'best practice' will be made possible; the positioning of the different parts (DIV's) of the website. (see next paragraph)

*Table structure VS Positioning of DIV's*

The complex table structure currently used on the website will be omitted in the new version, making use of the new CSS. Only one simple table will be used to position different parts of a page correctly. It is possible to completely omit the table structure. All parts of a page then must be enclosed by a DIV, which will be positioned in the style sheet. This way complete freedom is achieved in terms of where different parts of a page should appear, the size of these parts and all other style attributes. All parts can be positioned independently.

*Displaying links as block-level elements*

In the current situation often maps are used to achieve a bigger clickable area for a link. The same effect can be achieved by displaying links as block-level elements. This way all attributes which can be defined for a block-element (e.g. a DIV) can also be defined for a link. The benefit of using this method, is that when links are being moved, it is no longer necessary to modify the co-ordinates of the maps; the whole link will be moved, and because it is displayed as a block-level element, the clickable area around the link will move with it.

Because of the fact, a link now has the same attributes as any other block-level element, the links can be styled extensively. (e.g. with different sizes, paddings and margins, background-colors or images, etc.)

*Fixing bugs in Netscape 4.x*

Because of Netscape incompatibility regarding DIV's the new version of the website completely breaks in Netscape 4.x. Netscape also ignores media declarations for the style sheets. As a result, the CSS-files won't be interpreted by Netscape 4.x, because there's one where media="screen" and one where media="print", these are just ignored. This makes it possible to apply a work-around to achieve that a page is still readable in Netscape 4.x.

When a style sheet is written without a declaration of media, these styles will be interpreted by Netscape 4.x. On this style sheet can be defined which elements should be shown in Netscape 4.x and which elements should be hidden. This way all the elements that cause a page-layout to break, can be disabled. This style sheet will also be interpreted by the other browsers, but there are two more style sheets (one for printing and one for display on screen) These style sheets will override the first style sheet (the one without a media declaration).

### *Adapt to screen size and resolution*

In the current situation the website is best viewed in a resolution of 1024\*768. In fact, the pages don't adapt to the screen size or resolution. With CSS it is possible to let the pages adapt to size and resolution. By making use of DIV's for the separate parts of the pages, styles can be written for these DIV's, so that they 'grow' with the size and resolution of the screen. By using percentages for the sizes of the DIV's the desired effect can be achieved.

### *Achieve consistency regarding the use of styles*

In the current situation different styles are used for the same elements. E.g. links can look different on different pages, which does not contribute in the consistency of the styles throughout the website. One of the most important causes of this, is the fact that styles are declared in different ways and different locations. (e.g a style sheet is used, but there are a lot of hard-coded style declarations) In the new version all styles are defined in a style sheet. Because all styles are being defined in one place, it is a lot easier to reach consistency in terms of styles used for different elements throughout the website.

The first new version of the style sheet was aimed at the conservation of layout. All styles used in the current situation will also be found in the new version. To reach more consistency, an inventory can be drawn up, of which styles can be used for which element.

## 2 Script for modifying HTML in Notes databases

### General Directives

#### Images

Images are stored in the Graphics Depository database, on the form ObjectStore. This form is viewed and edited from the Notes Client. Make sure that if you don't want the values for attributes like width and height hard-coded in the HTML file, there are no values for these attributes on the form ObjectStore for that image in the Graphics Depository database.

Object Store	
Object Name:	StandardProductsBanner-1.gif
Type:	Graphic banner
Description:	
Alternative text:	**
Width	
Height	
Border	0
Click on button to paste the URL	<a href="http://www.europe.fasson.com/ite/planets/gbfrehome.nsf/b/acgn-5bhh23?opendocument&amp;ul=1&amp;linkid=thun-5b4he&amp;linktype=readabout&amp;ucn=739afd58b43f651fc1256bl200544d07">http://www.europe.fasson.com/ite/planets/gbfrehome.nsf/b/acgn-5bhh23?opendocument&amp;ul=1&amp;linkid=thun-5b4he&amp;linktype=readabout&amp;ucn=739afd58b43f651fc1256bl200544d07</a>
Opentype	<input checked="" type="radio"/> opendocument <input type="radio"/> openform <input type="radio"/> openview <input type="radio"/> readform
Language path	
Db Name	gbfrehome.nsf
View	b
Doc ID	acgn-5bhh23
Banner ID	FREW-5CUA.JX
URL Link	[<a href="/ite/planets/gbfrehome.nsf/b/acgn-5bhh23?opendocument&ul=1&BannerID=FREW-5CUA.JX[placeholder]"><IMG SRC="/ite/planets/deposit.nsf/vObjectStore/StandardProductsBanner-1.gif/\$file/StandardProductsBanner-1.gif" Border=0 ALT="" ></a>]
URL Link II	<a href="/ite/planets/gbfrehome.nsf/b/acgn-5bhh23?opendocument&ul=1&BannerID=FREW-5CUA.JX[placeholder]"><IMG SRC="/ite/planets/deposit.nsf/vObjectStore/StandardProductsBanner-1.gif/\$file/StandardProductsBanner-1.gif" Border=0 ALT="" ></a>
Attach Object Here:	

#### HTML output

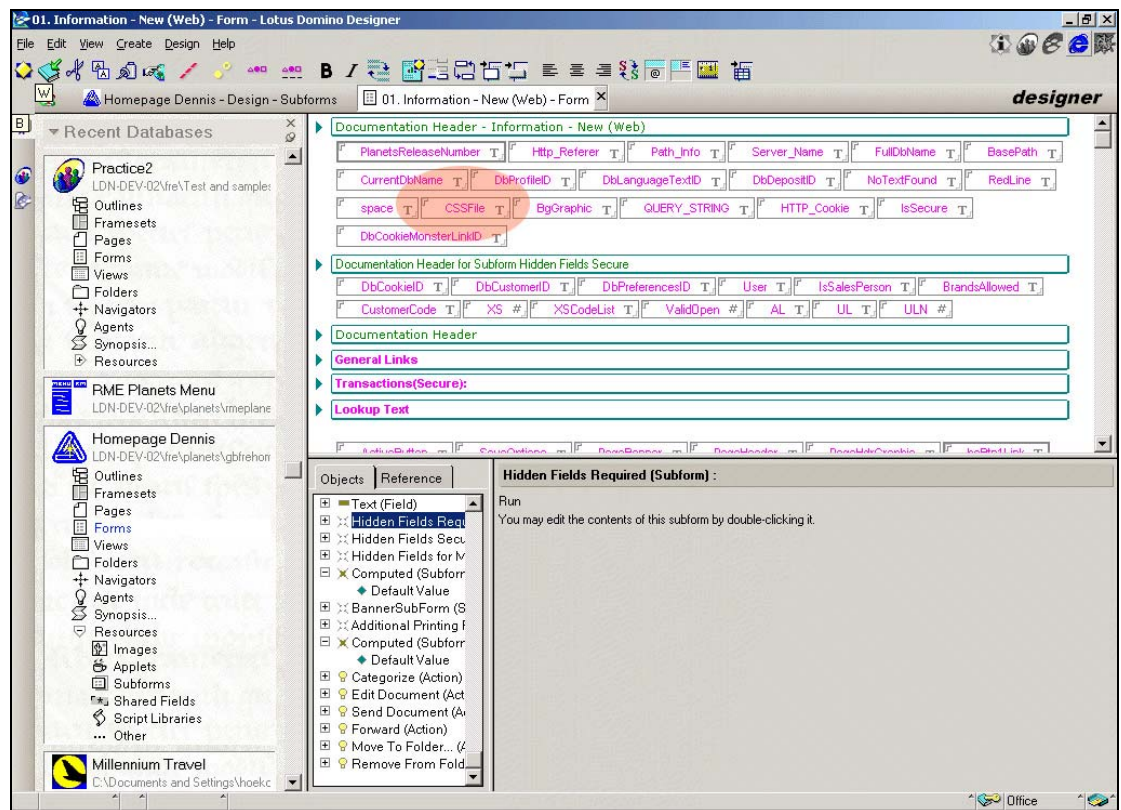
Regularly check the HTML output to detect errors on time. Compare the source of the new page with the source of the local pages.

#### Temporary styles

Use temporary styles to visualise the structure of DIV's, TD's, etc. with borders.

## Linking the CSS

Make sure the HTML is linked to the style sheets. To do this, add a hidden text field CSSFile in the Documentation Header (this is the subform Hidden Fields Required) on the form 01. Information New (Web).



The value of the text field CSSFile should be as follows;

The screenshot displays the Lotus Notes formula editor for the **CSSFile (Field) : Value** property. The left pane shows the **Objects** list with various fields and functions. The right pane shows the formula text.

**Objects List:**

- PlanetsReleaseNumber (Field)
- Http\_Referer (Field)
- Path\_Info (Field)
- Server\_Name (Field)
- FullDbName (Field)
- BasePath (Field)
- CurrentDbName (Field)
- DbProfileID (Field)
- DbLanguageTextID (Field)
- DbDepositID (Field)
- NoTextFound (Field)
- RedLine (Field)
- space (Field)
- CSSFile (Field)
- BgGraphic (Field)
- QUERY\_STRING (Field)
- HTTP\_Cookie (Field)
- IsSecure (Field)
- DbCookieMonsterLinkID (Field)

**Formula Text:**

```

@Char(10) + "<!-- Lotus Notes Team, ASD, ADFRE, Leiden, Holland -->" + @Char(10) + "<!--
All contents copyright (C) 2001, all rights reserved. -->" + @Char(10) + "<!--
----->" +
@Char(10);

LookupText := @DbLookup ( "" ; DbLanguageTextID ; "v0100" ; "Metatags" ; 2 );
Metatagstext := @Implode ( @If ( @IsError ( LookupText ) | LookupText = "" ; NoTextFound ;
LookupText ) ; " " );

Metatagsfield := "<META name=\"description\" content=\"" + Metatagstext + "\">" +
@Char(10) + "<META name=\"keywords\" content=\"" + Metatagstext + "\">" + @Char(10);

tCSSFileScr := @DbLookup ( "" ; DbDepositID ; "vObjectStore" ;
"new_total_screen.css" ; "PastedURLLink" );
tCSSFileScreen := @If ( @IsError ( tCSSFileScr ) ; "" ; tCSSFileScr );

tCSSFilePrint := @DbLookup ( "" ; DbDepositID ; "vObjectStore" ; "new_total_print.css" ;
"PastedURLLink" );
tCSSFilePrint1 := @If ( @IsError ( tCSSFilePrint ) ; "" ; tCSSFilePrint );

copyrighttext + Metatagsfield + "<link rel=\"stylesheet\" href=\"" + tCSSFileScreen + "
media=\"screen\" type=\"text/css\">" + "<link rel=\"stylesheet\" href=\"" + tCSSFilePrint1 + "
media=\"print\" type=\"text/css\">"
    
```



### *Adding Doc Type*

Before making any changes in the HTML, add the correct Doc Type on the HTML pages. The declaration for the Doc Type should be the on the first line of the HTML document. Because it is impossible in Lotus Notes to edit the first line of the HTML output, another structure will be used. This has only been tested for a local page and first should be tested in Lotus Notes. The following structure should fix the problem;

---

```
<html>
<head>
...
</head>
</html>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/REC-html40/strict.dtd">

<html>
<head>
...
</head>

<body>
...
</body>
<html>
```

---

To achieve this, you can add a field on the form 01. Information – New (Web) in the PageHeader

*Modifying the table-structure*

First change the complex overall table-structure. You could also do this along the way (e.g. changing the table-structure of the header, while you working on modifying the header), but as a result of the broken tables you get when you do this, you must implement temporary work-arounds to fix this. Later these workarounds still need to be modified, so it means more work. In the new situation there will be only one overall table structure. It should be like the following structure. (In the end, when all the changes are done, you should check the HTML-output to make sure only one table is used. A single table could occur in the content, but only one table should be used for overall structure.)

```
<table width="100%" border="0" cellspacing="0" cellpadding="0">

  <tr>

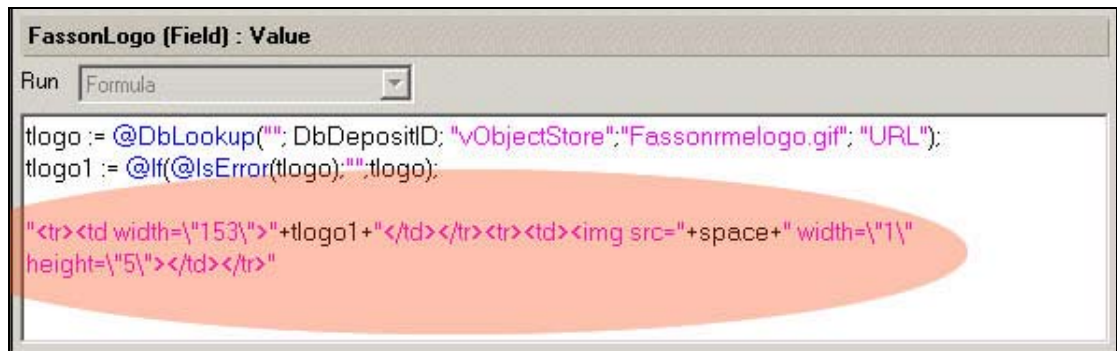
    <td id="sidebar" rowspan="4">
      Sidebar
    </td>

    <td id=header-top>
      top part of the header
    </td>
  </tr>
  <tr>
    <td id=header-middle>
      middle part of the header
    </td>
  </tr>
  <tr>
    <td id=header-bottom>
      bottom part of the header
    </td>
  </tr>
  <tr>
    <td id="content2">
      content
    </td>
  </tr>
  <tr>
    <td id="footer" colspan="2">
      footer
    </td>
  </tr>
</table>
```

The modifications that are needed for every block (header, sidebar, footer and content) regarding to the overall-table structure, is discussed in the section for that specific block. The modifications regarding to the overall table structure is what's first discussed for each block. Header and sidebar will be discussed in one section, because the code for these blocks appear in the same subform.



More HTML regarding to overall table structure of the sidebar is hard coded in the field FassonLogo. The old situation looks like this;



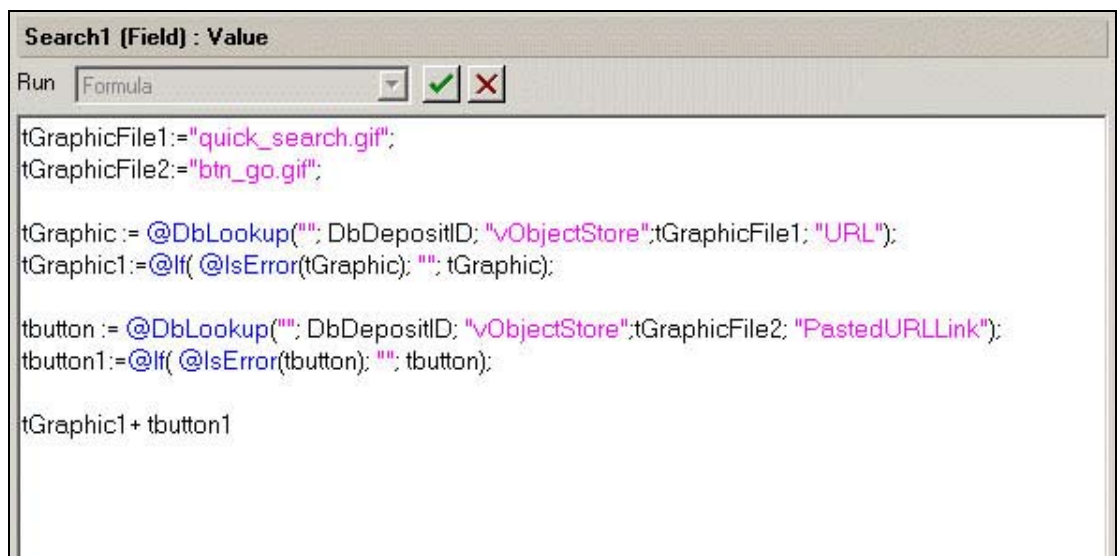
Modify the code to the following situation;



Under the field FassonLogo you find more textfields; SearchTableFMT and further. Most of these fields contain nothing but HTML regarding to the overall table structure and are therefore not needed in the new situation. All fields of this group can be removed, except for the following list;

- Search1
- QueryQuick
- Search3
- Search4
- CustomizeBtn

Make sure the values of these fields don't contain anymore HTML regarding to tables. For example the value for textfield Search1 should be as in the following situation;



The field CustomizeBtn also contains a lot of HTML code regarding to the overall table structure;

CustomizeBtn (Field) : Value
Run Formula

```

REM "tGraphicFile2 has changed into multi-language lookup from the Graphic depository db.";
tGraphicFile2:="btn_customize-"+UL+".gif";

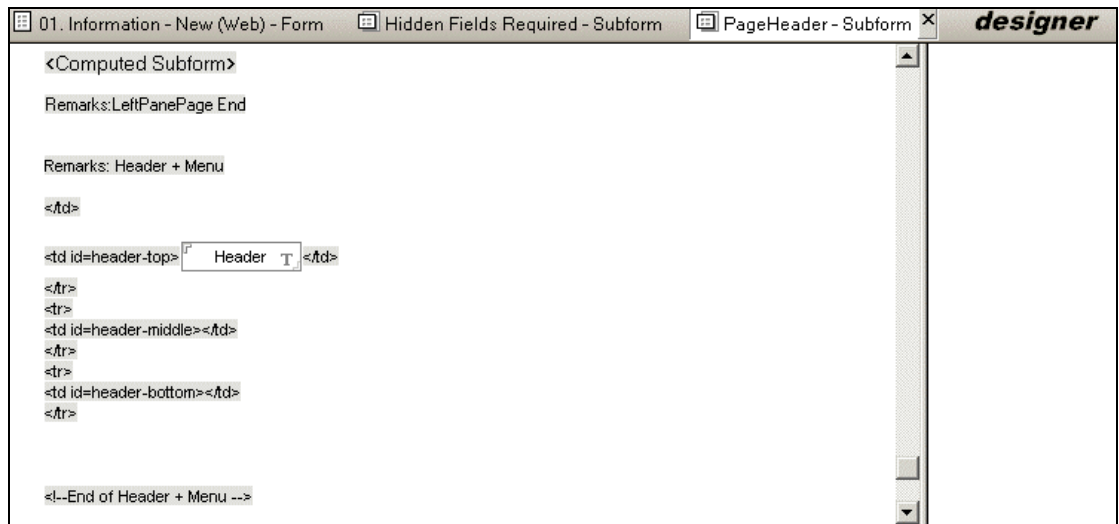
tCustomize1 := @DbLookup(;; DbDepositID; "vObjectStore";tGraphicFile2; "URL2");
tCustomize1a :=@If( @IsError(tCustomize1); ""; tCustomize1);
URL1 := "<tr><td width=120 align=left valign=top bgcolor=#FFFFCC><a href=\""
+CustomizeLinkS+"\">">tCustomize1a+"</a></td></tr>";

tBackgroundPic1 := @DbLookup(;; DbDepositID; "vObjectStore"; "navboxyellow_bottom.gif" ;
"PastedURLLink");
tBackgroundPic1a :=@If( @IsError( tBackgroundPic1 ); ""; tBackgroundPic1 );

tRegister := RegisterLink;
tRegisterText := RegistrationText;

RegisterTableRow := " <tr><td><table width='100%' border='0' cellspacing='0'
cellpadding='0'><tr><td bgcolor='#FFFFCC' colspan='2'></td><td bgcolor='#CCCCCC' width='1'></td></tr><tr><td bgcolor='#FFFFCC'></td><td bgcolor='#FFFFCC' valign='middle' align='left'><a href="
+ tRegister +" class=menuitem10px>">tRegisterText +"</a></td><td bgcolor='#CCCCCC'
width='1'></td></tr></table></td></tr><tr><td width='121' align='left'
valign='top' background="
+ tBackgroundPic1a + "> in screenshot).

Add the HTML, as in the screenshot below;





Adding DIV's and reaming modifications for header and sidebar

The next step consists of adding the DIV's with the correct id's and/or classes. Most DIV's can be added on the subform PageHeader. Modify the code like in the following example;

```
//-->
</script>
<!-- Start of Search menu -->
<form name="queryform">
  <div id="search">
    <div id="QS">
      QuickSearchMap T
    </div>
    FassonLogo T
    Search1 T
    <div id="QS_in">
      QueryQuick T
    </div>
    Search3 T
    Search4 T
    CustomizeBtn T
  </div>
</form>
<!-- End of Search menu -->
```

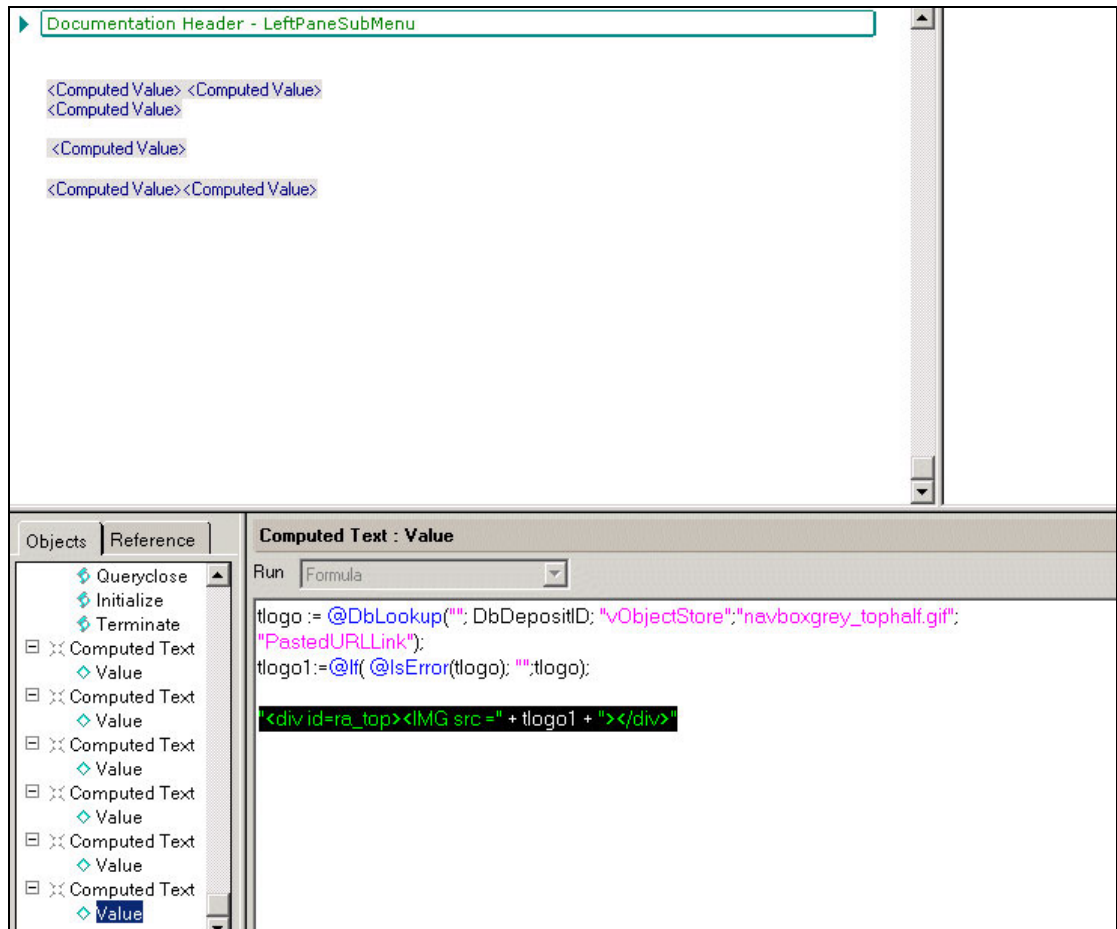
Add all the necessary DIV's like in the following screenshot;

```
FassonLogo T

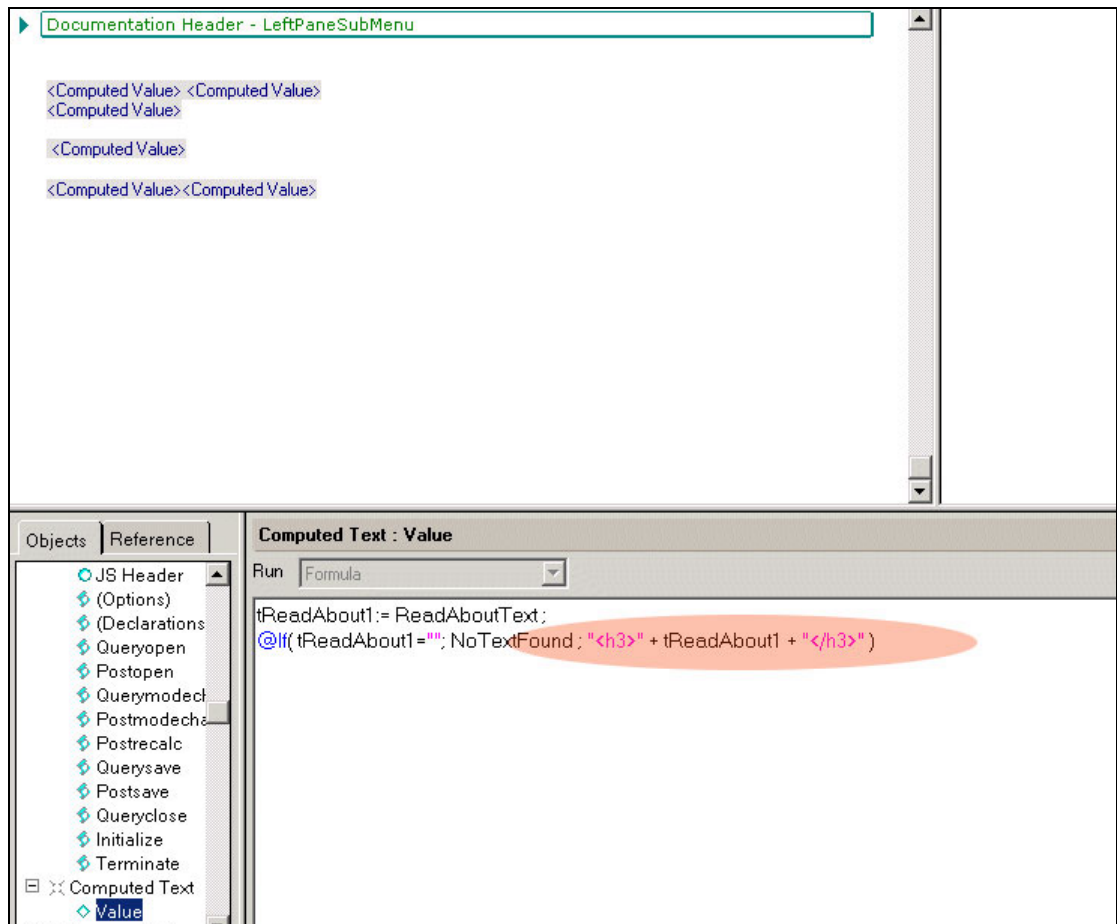
<div id="search">
  <div id="QS">
    Search1 T
  </div>
  <div id="QS_in">
    QueryQuick T
  </div>
  <div id="adv_search">
    Search3 T
    Search4 T
  </div>
  <div id="register">
    CustomizeBtn T
  </div>
</form>
<!-- End of Search menu -->

<Computed Subform>
<Computed Subform>
```

More DIV's need to be added on the subform LeftPaneSubMenu. To accomplish this, the computed values on the subform LeftPaneSubMenu need to be modified. Modify the code as in the following example;



One of the computed values on the subform LeftPaneSubMenu contains the title for the 'Read About' links. The last modification in the sidebar is needs to be done in this computed field. One HTML tag, <H3> needs to be added here, so that the new situation looks like this;



### Modifying the footer

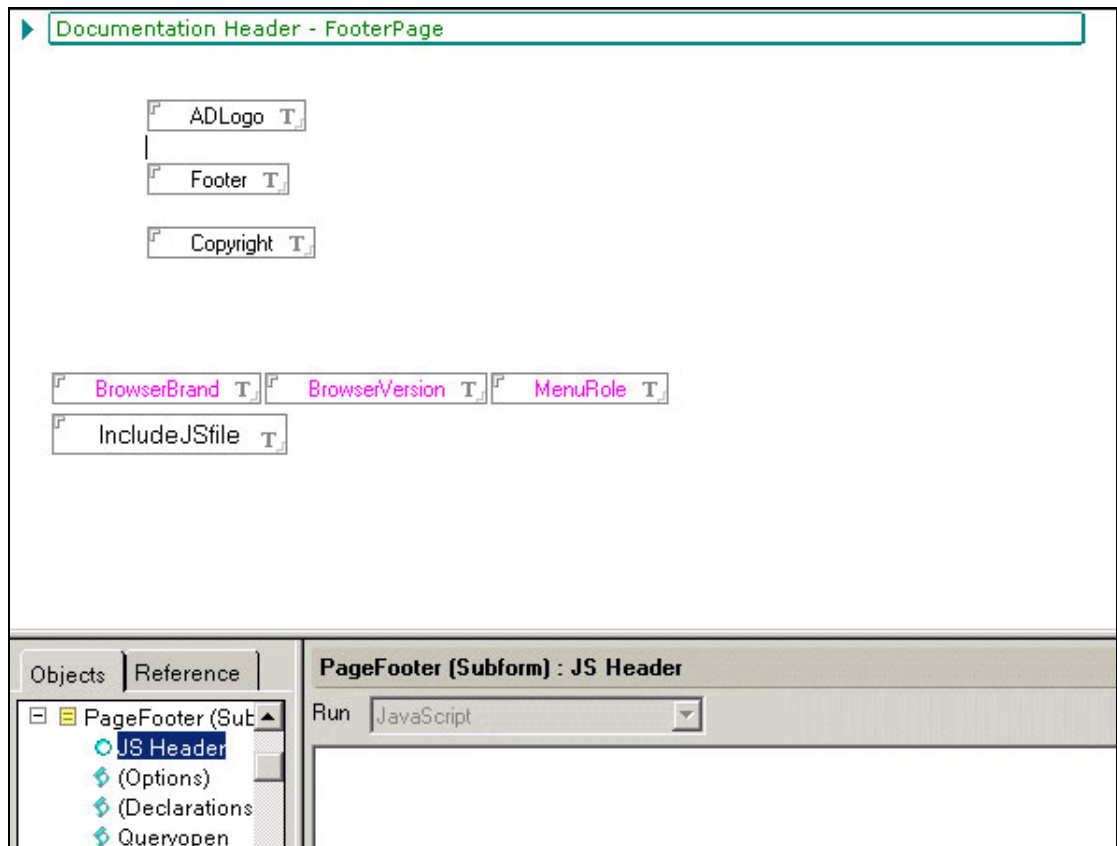
All code for the footer is on the subform PageFooter.

#### Modifying the table structure for the footer

All HTML in the screenshot below (subform PageFooter in old situation) is related to the old table structure.



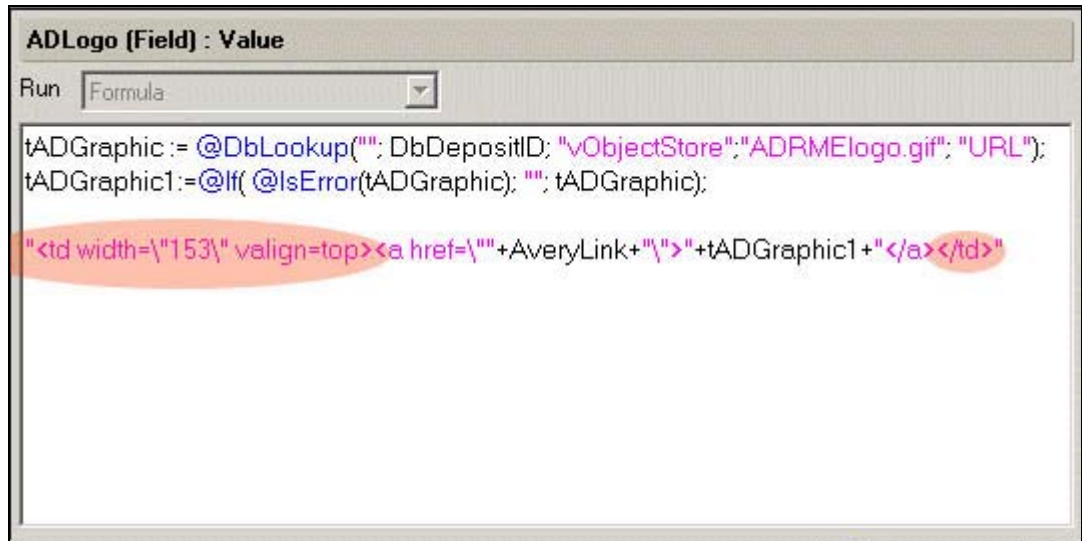
All HTML can be removed, including the computed values; these are for the one-pixel-sized gif's. The result would be the following situation;



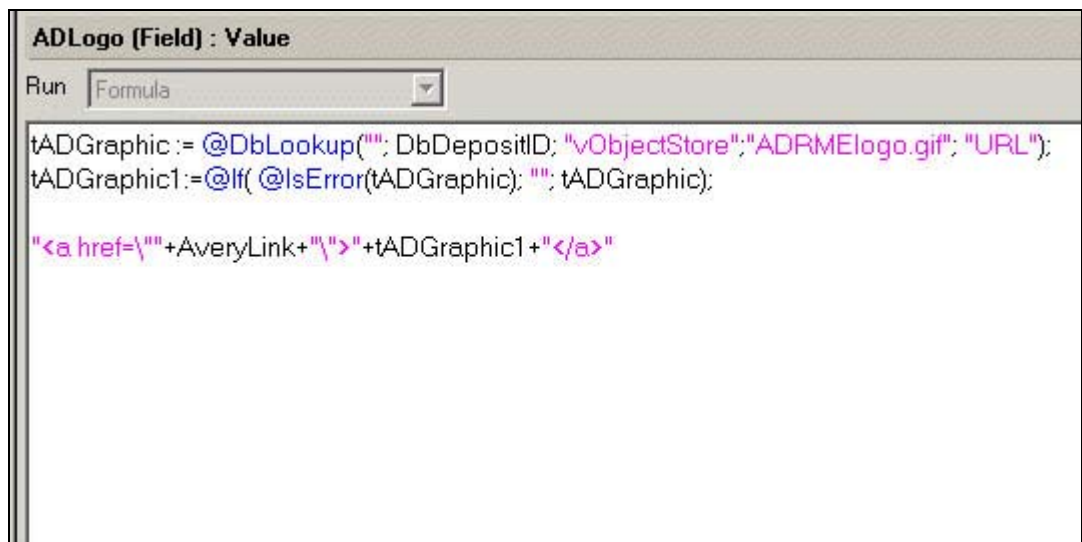
The final modifications regarding to the table structure are done in the three textfields;

- ADLogo
- Footer
- Copyright

The screenshot below is the value for textfield ADLogo in the old situation;



Modify the value for textfield ADLogo;



A similar situation as for textfield ADLogo goes for the textfields Footer and Copyright. Remove the HTML in the values of these fields with the following situations as a result;

New situation for the value for field Footer;

Footer (Field) : Value

Run

Formula

```

CustomizeLookupText := @DbLookup( "" ; DbLanguageTextID; "v0100"; "ftCustomize";
ULN );
tCustomize := @If ( @IsError ( CustomizeLookupText ) | CustomizeLookupText = "" ;
NoTextFound ; CustomizeLookupText );




"<a href=" + HomeLink+ ">" +HomeText+ "</a>" + " | " +
"<a href=" + Btn1Link+ ">" +ProductServiceText+"</a>" + " | " +
"<a href=" + Btn2Link+ ">" +MarketSegmentText+"</a>" + " | " +
"<a href=" + Btn3Link+ ">" +NewsInfoText+"</a>" + " | " +
"<a href=" + Btn4Link+ ">" +TransactionsText+"</a>" + " | " +
"<a href=" + ContactUsLink+ ">" +tContact+"</a>" + "<br>" +
"<a href=" + AdvSearchLink+ ">" +tSearch+"</a>" + " | " +
"<a href=" + SitemapLink+ ">" +SitemapText1+"</a>" + " | " +
@if( User != "Anonymous" ; ("<a href=" +CustomizeLinkS+ ">" +tCustomize+"</a>" + " | ") ; "" )
+
"<a href=" +InvestorLink+ ">Investor Relations</a>" + " | " +
"<a href=" +AveryLink+ ">AveryDennison.com</a>"

```



Office

New situation for the value for field Copyright;

**Copyright (Field) : Value**

Run    

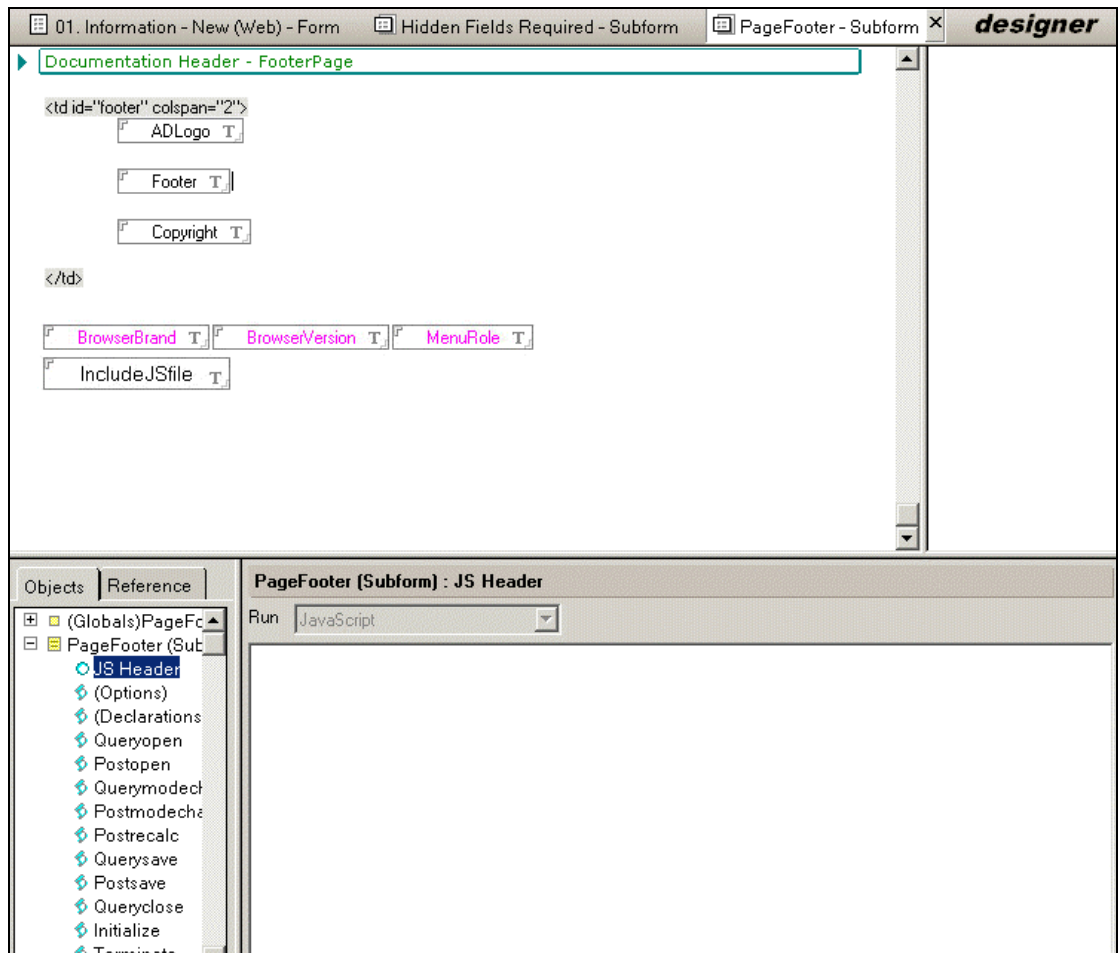
```
tCurrentYear:=@Year(@Now);  
  
" &copy; 1996-" + @Text(tCurrentYear) + " Avery Dennison Corporation. All  
rights reserved.&nbsp;<a href=" + DisclaimerLink + "></a>&nbsp;<a href=" +  
+LegalLink + ">Legal Notice</a>&nbsp;<a href=" + PrivacyLink + ">Privacy  
Statement</a><br><br>"
```

 Office 



### Adding DIV's and reaming modifications for the footer

First fix the table structure for the footer. Add the td with the id=footer on the subform PageFooter;



Also add DIV's in the textfields;

- Footer
- Copyright

DIV's added for the value for textfield Footer;

**Footer (Field) : Value**

Run

```
CustomizeLookupText := @DbLookup( "" ; DbLanguageTextID; "v0100"; "ftCustomize";
ULN );
tCustomize := @If ( @IsError ( CustomizeLookupText ) | CustomizeLookupText = "" ;
NoTextFound ; CustomizeLookupText );

"<div id=\"foot_links\">"+
"<a href=\" + HomeLink+ \">\" + HomeText+ \"</a>\"+ \" | \" +
"<a href=\" + Btn1Link+ \">\" + ProductServiceText+ \"</a>\"+ \" | \" +
"<a href=\" + Btn2Link+ \">\" + MarketSegmentText+ \"</a>\"+ \" | \" +
"<a href=\" + Btn3Link+ \">\" + NewsInfoText+ \"</a>\"+ \" | \" +
"<a href=\" + Btn4Link+ \">\" + TransactionsText+ \"</a>\"+ \" | \" +
"<a href=\" + ContactUsLink+ \">\" + tContact+ \"</a>\"+ \"<br>\" +
"<a href=\" + AdvSearchLink+ \">\" + tSearch+ \"</a>\"+ \" | \" +
"<a href=\" + SitemapLink+ \">\" + SitemapText1+ \"</a>\"+ \" | \" +
@If( User != \"Anonymous\" ; ( \"<a href=\" + CustomizeLinkS+ \">\" + tCustomize+ \"</a>\"+ \" | \" ) ; \" \" )
+
"<a href=\" + InvestorLink+ \">Investor Relations</a>\"+ \" | \" +
"<a href=\" + AveryLink+ \">AveryDennison.com</a>\"+ \"</div>\"
```

Office

DIV's added for the value for textfield Copyright;

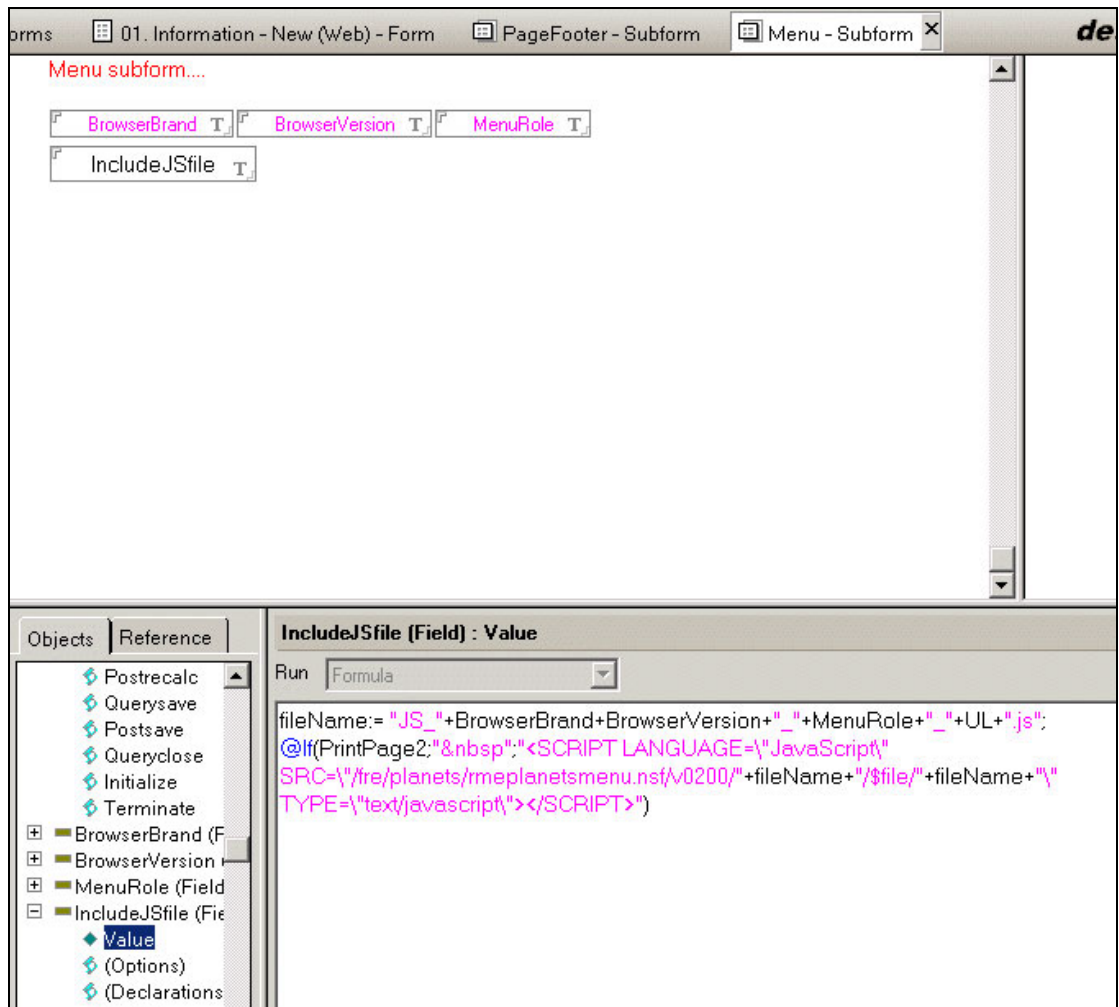
**Copyright (Field) : Value**

Run  ☒ ☐

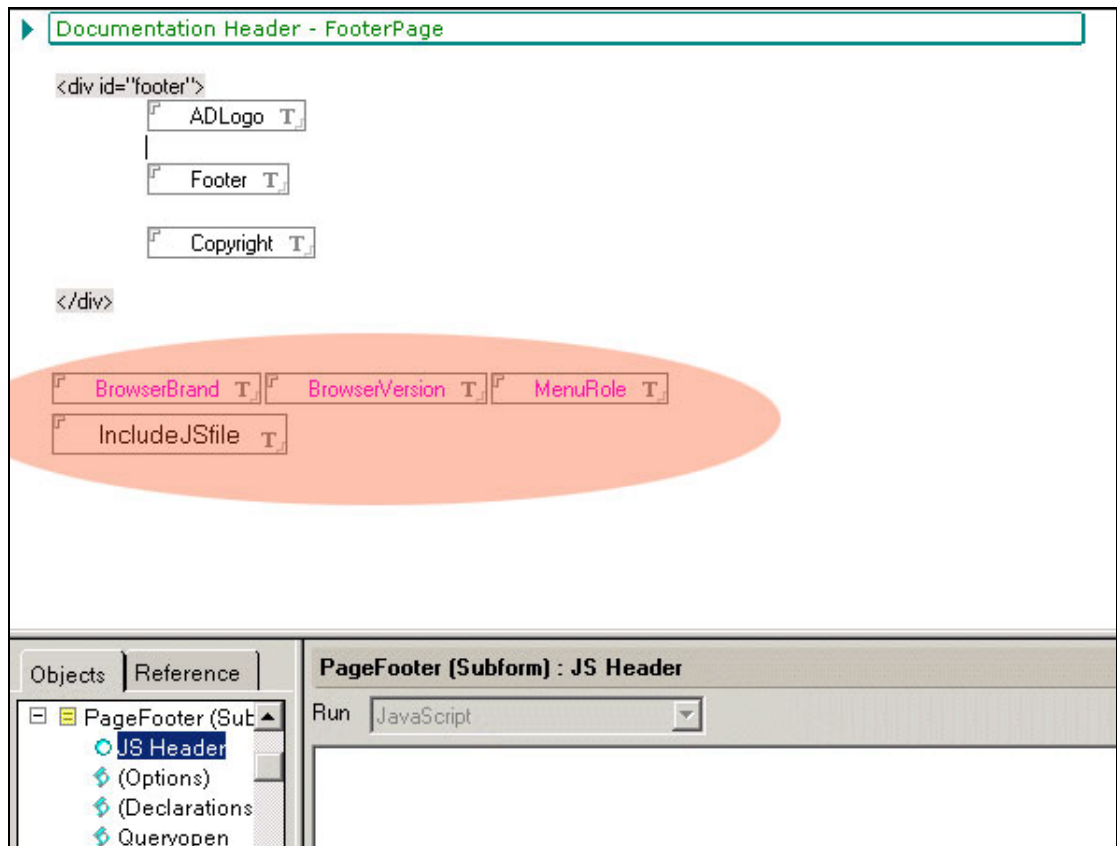
```
tCurrentYear:=@Year(@Now);

"<div id=copyright>&copy; 1996-\"+@Text(tCurrentYear)+\" Avery Dennison Corporation. All
rights reserved.&nbsp;<a href=\" + DisclaimerLink+ \"</a>&nbsp;<a href=\"
+Legallink+ \">Legal Notice</a>&nbsp;<a href=\" +PrivacyLink+ \">Privacy
Statement</a><br><br></div>\"
```

In the new situation the code for the JS-file for the menu on the website is placed in a separate subform Menu;



Remove the marked code from the following screenshot for the JS-file from the subform PageFooter;



To add the code for the menu to the page again, add a computed subform on the subform PageFooter, with the following situation as a result;

<pre> &lt;div id="cont2_intro"&gt;   dPageHeader T &lt;/div&gt;  Text T  &lt;div id="cont2_banner"&gt;   dPageBanner T &lt;/div&gt;  &lt;/div&gt;&lt;/div&gt;  &lt;div id="print"&gt;   &lt;A HREF="#" onClick="window.print();" &gt;Computed Value &lt;/A&gt; &lt;/div&gt;  &lt;/td&gt; &lt;/tr&gt; &lt;/tr&gt; </pre>
<Computed Subform>
</table>
<Computed Subform>

(everything you see above the computed field, is for the content and has already been modified in this screenshot)

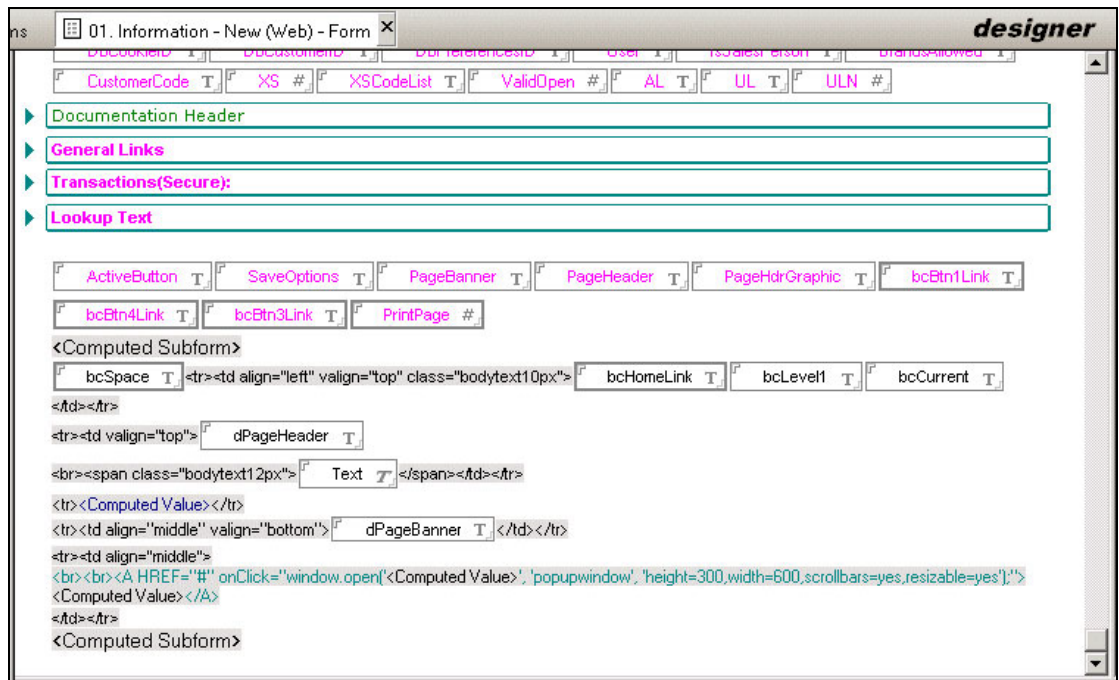
The value for the first computed subform is "PageFooter". The value for the second computed subform is "Menu".

### Modifying the content (layout 2)

All code for the content is on the form 01. Information – New (Web).

### Modifying the table structure for the content

All table-related HTML in the screenshot below (form 01. Information – New (Web) in old situation) is related to the old table structure and can be removed. The textfield bcSpace is used for this gif and can be removed.



The textfield `dPageBanner` and the code around it is part of the subform `BannerSubForm`. The code around it, including the computed value used for the one-pixel-sized gif, is part of the table structure and can be removed.

After removing the table-related HTML the following situation is the result;

The screenshot shows a web form designer interface with a title bar "01. Information - New (Web) - Form" and a "designer" logo. The form layout includes the following elements:

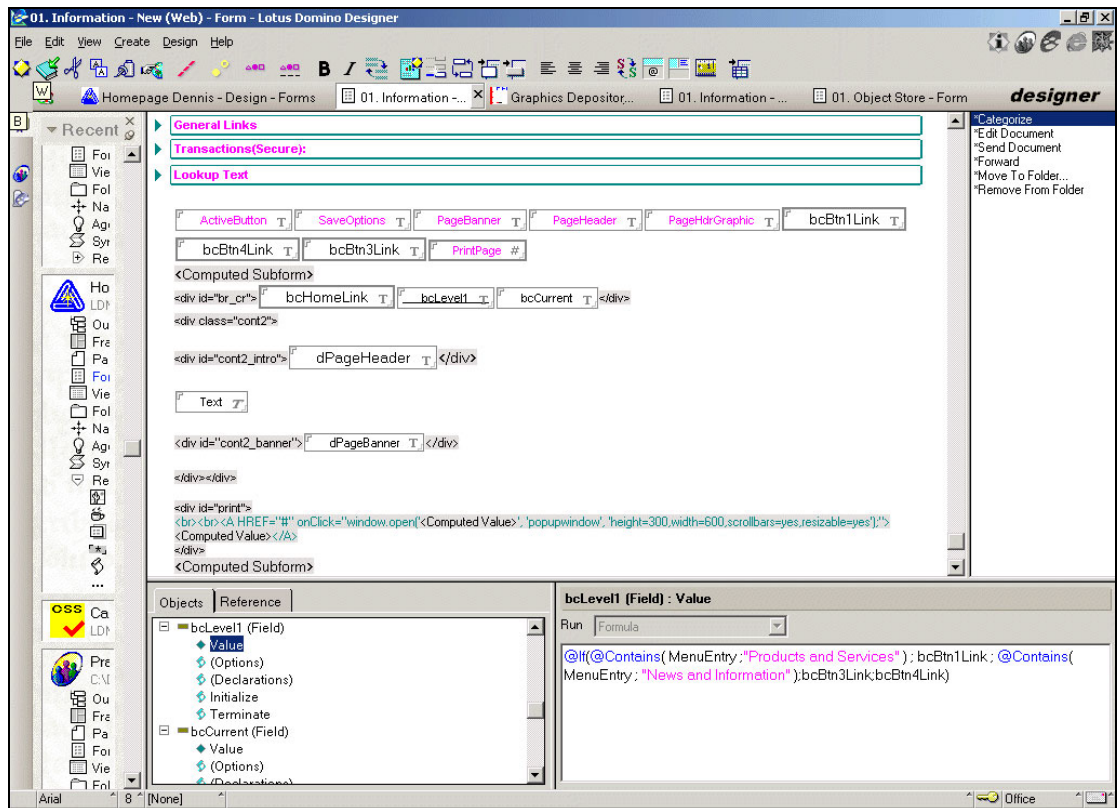
- Fields: `CustomerCode T`, `XS #`, `XSCodeList T`, `ValidOpen #`, `AL T`, `UL T`, `ULN #`.
- Section Headers: `Documentation Header`, `General Links`, `Transactions(Secure):`, `Lookup Text`.
- Fields: `ActiveButton T`, `SaveOptions T`, `PageBanner T`, `PageHeader T`, `PageHdrGraphic T`, `bcBtn1Link T`, `bcBtn4Link T`, `bcBtn3Link T`, `PrintPage #`.
- Subform: `<Computed Subform>` containing `bcHomeLink T`, `bcLevel1 T`, and `bcCurrent T`.
- Fields: `dPageHeader T`, `Text T`, `dPageBanner T`.
- Code Snippets:
 

```
<br><br><A HREF="#" onClick="window.open('<Computed Value>', 'popupwindow', 'height=300,width=600,scrollbars=yes,resizable=yes');">
<Computed Value></A>
```
- Subform: `<Computed Subform>`.

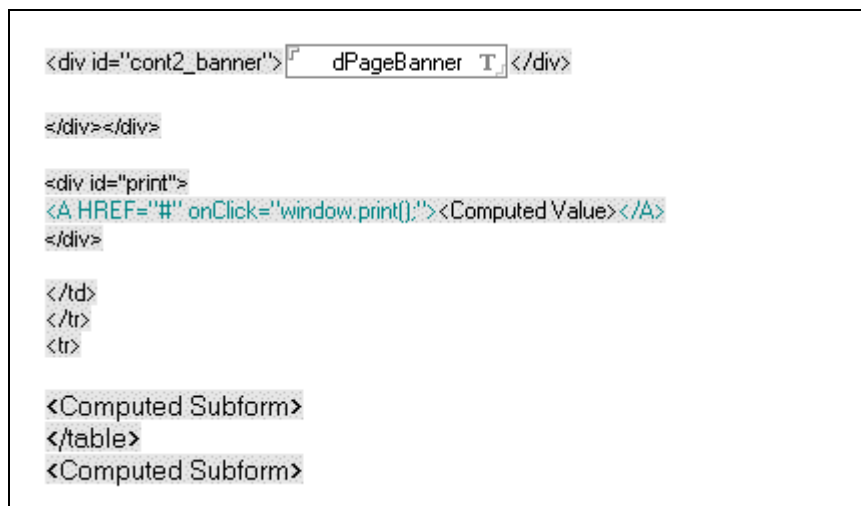


## Adding DIV's and reaming modifications for the content

Add the DIV's with the correct id's or classes like with the following situation as a result;



Also fix the table structure\_by adding the correct table HTML.





## Bread Crumbs

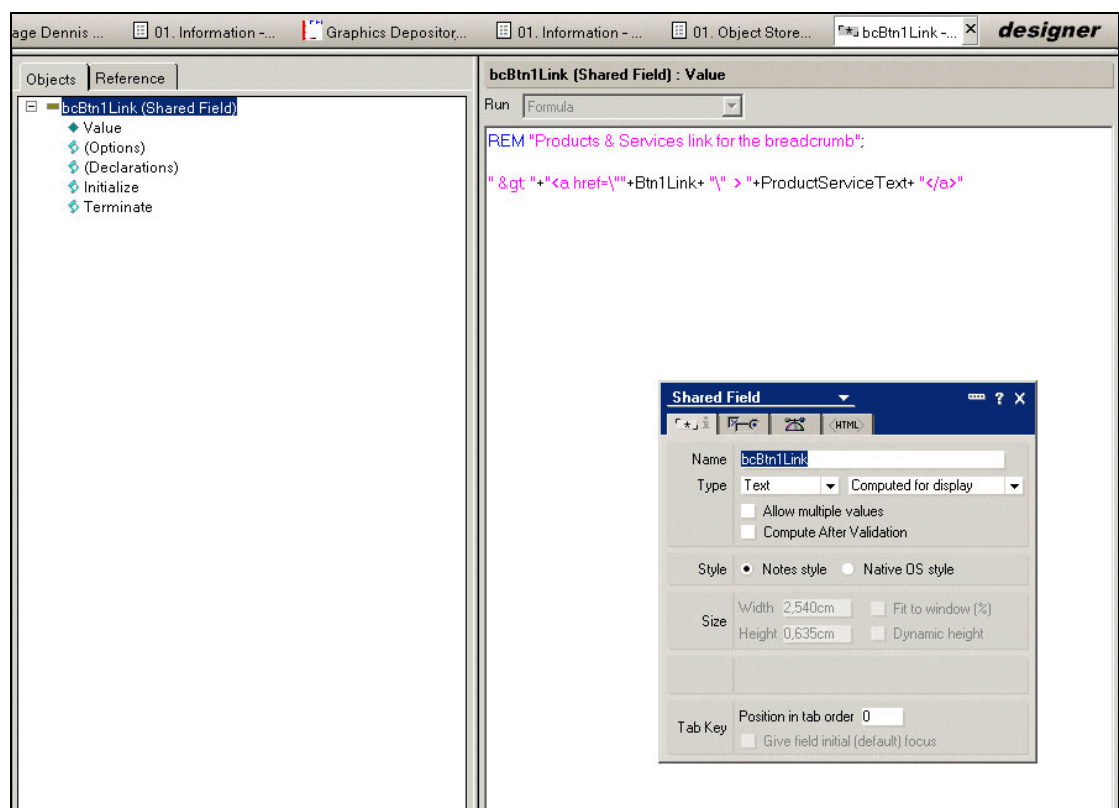
The following fields are used for the bread crumbs;

- bcHomeLink
- bcLevel1
- bcCurrent

Text field bcLevel1 uses three shared text fields;

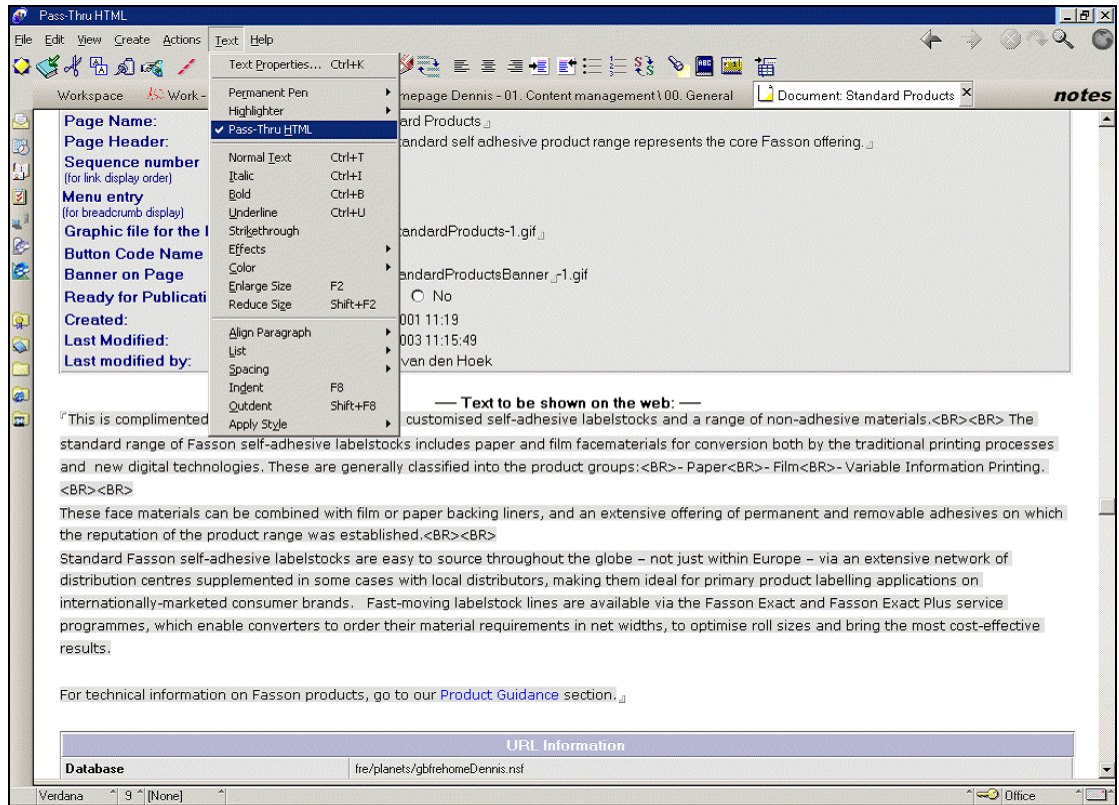
- bcBtn1Link
- bcBtn3Link
- bcBtn4Link

These fields contain declarations for classes. Change the values for these three fields like in the following example for the value for field bcBtn1Link



## Text

The value for this field is the text for the content on the page. The text comes from the form 01. Information – New (Notes) This form is viewed and edited in the Notes Client. To make sure the text can contain HTML, choose Pass-Thru HTML in the menu Text. Otherwise the text will be interpreted as rich text and the font styles used on this form will also appear on the HTML file.

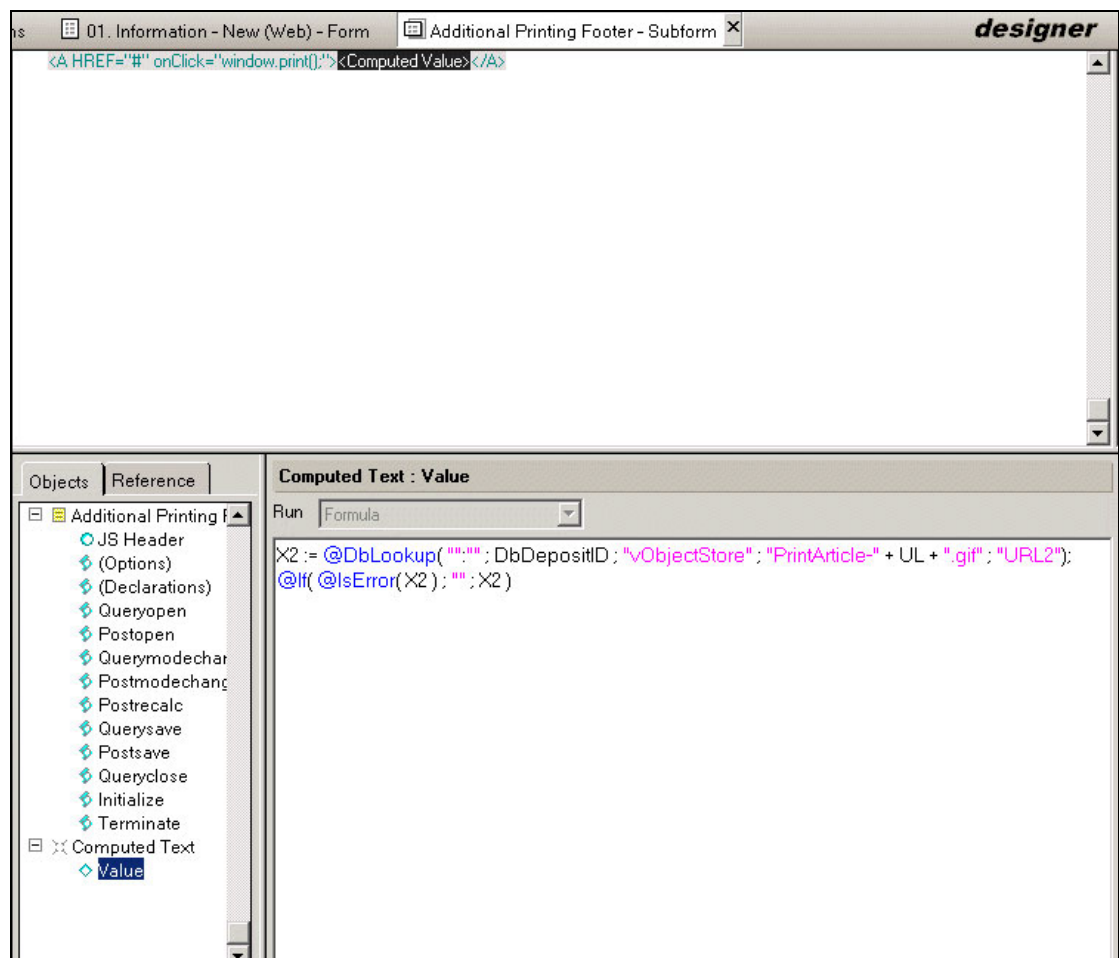


### dPageBanner

The image for the banner used on the page is stored in the Graphics Depository database, on the form ObjectStore. This form is viewed and edited from the Notes Client. Make sure that if you don't want the values for attributes like width and height hard-coded in the HTML file, there are no values for these attributes on the form ObjectStore for that image in the Graphics Depository database.

### Print

Modify the subform Additional Printing Footer like in the following situation;



Only the second computed value from the old situation remains. This is the highlighted value in the screenshot (for retrieving the picture used for the print button).