Goodwill Reklambyrå på Internet - två webbapplikationer

Julia Flodkvist
Hanna Forsberg

2002-08-29
Goodwill Reklambyrå på Internet - två webbapplikationer

Examensarbete utfört i elektronisk publicering vid Linköpings Tekniska Högskola, Campus Norrköping

Julia Flodkvist
Hanna Forsberg

Examinator: Niklas Rönnberg
Norrköping den 29:e augusti 2002
Goodwill Reklambyrå på Internet – två webbapplikationer

Julia Flodkvist och Hanna Forsberg


This degree project is made for Goodwill Reklambyrå in Norrköping. The task was to create a homepage and a business community for the company. By business community we mean an application on the Internet where the company staff are able to read, update, delete and create posts which are stored in databases. Mainly have Flash been used to create the homepage and ASP (Active Server Pages) to build the business community. The final result was an entertaining and informative homepage and a secure and useable business community. In this report the reader gets a detailed statement of our work.

Asp, Flash, webbapplikationer
# Innehållsförteckning

1 Inledande arbete ............................................................................................................. 1
  1.1 Bakgrund ................................................................................................................... 1
    1.1.1 Goodwill Reklambyrå ........................................................................................ 2
    1.1.2 Kravspecifikation ............................................................................................... 2
  1.2 Uppgift .................................................................................................................... 3
    1.2.1 Mål ..................................................................................................................... 3
  1.3 Metod ...................................................................................................................... 4
  1.4 Planering .................................................................................................................. 5
    1.4.1 Tidsplan .............................................................................................................. 5
  1.5 Rapportens uppbyggnad ............................................................................................. 6
    1.5.1 Syfte ................................................................................................................... 6
    1.5.2 Struktur ............................................................................................................... 6

2 Från tanke till färdigt resultat ............................................................................................. 7
  2.1 Teori ...................................................................................................................... 7
    2.1.1 Active Server Pages (ASP) ................................................................................ 7
    2.1.2 Flash ................................................................................................................. 12
    2.1.3 Dreamweaver ................................................................................................... 14
    2.1.4 3ds max ............................................................................................................ 15
  2.2 Förberedelser ............................................................................................................ 18
    2.2.1 Design ............................................................................................................... 18
    2.2.2 Användbarhet ................................................................................................... 25
  2.3 Utförande .................................................................................................................. 30
    2.3.1 Hemsidan .......................................................................................................... 30
    2.3.2 Företagsportalen ............................................................................................... 37

3 Diskussion ................................................................................................................... 47
  3.1 Resultat ................................................................................................................... 47
    3.1.1 Hemsidan .......................................................................................................... 47
    3.1.2 Företagsportalen ............................................................................................... 48
  3.2 Svårigheter och problem .......................................................................................... 50
  3.3 Framtid ..................................................................................................................... 51
    3.3.1 Hemsidan .......................................................................................................... 51
    3.3.2 Företagsportalen ............................................................................................... 51
4 Avslutning .......................................................................................................................... 52
5 Referenser .......................................................................................................................... 53
1 Inledande arbete

I detta kapitel beskrivs bakgrunden till examensarbetet och läsaren får dessutom en kortfattad presentation av Goodwill Reklambyrå. Kapitlet innefattar en ingående beskrivning av uppgiften, vilka kravspecifikationer vi hade och vilka mål vi ville uppfylla. Även metoderna som användes och tidsplanen för arbetet tas upp.

1.1 Bakgrund

Vi har läst högskoleingenjörsutbildningen Medie- och Kommunikationsteknik, båda med inriktning mot medieteknik. När det var dags för vårt examensarbete så ville vi gärna göra det åt ett medieföretag för att få en inblick i ett framtida arbetsliv. Det vi ville få ut av uppdraget var att lära oss mer om multimedia på webben. Till det så behövde vi utnyttja, och självklart fördjupa, våra kunskaper som vi fått under de här tre åren.


Vi ville även arbeta med programmering och då särskilt kopplningar till databaser. Efter lite funderade och diskussioner med vår handledare kom vi fram till att vi skulle göra en applikation i ASP. Vi hade redan bestämt att vi på hemsidan skulle ha en intresseanmälan där uppgifterna som användaren matade in skickas och lagras i en databas. Utifrån detta kom vi på idén att vi skulle skapa en företagsportal där de anställda på Goodwill kunde logga in och hämta information från olika databaser.

Vi skulle alltså skapa två olika sorters applikationer som båda skulle fungera väl på Internet.
1.1.1 Goodwill Reklambyrå


Vår handledare på Goodwill blev Peter Fällman som är VD och ägare av företaget.

1.1.2 Kravspecifikation

Goodwill hade vissa önskemål när det gällde den nya hemsidan, de vänder sig till en speciell kundkrets och ville att sidan skulle vara anpassad efter deras behov och förutsättningar. Hemsidan skulle därför göras i Macromedia Flash istället för Macromedia Director, som vi från början hade föreslagit, då det var vanligare med en Flashspelare än en Shockwavespelare bland deras kunder. Det var även viktigt att inte få för stor filstorlek då de inte ville ha för lång laddningstid, men vi kunde i skapandet av sidan ändå räkna med att de flesta av användarna sitter på ett företagsnät med relativt hög bandbredd.

Företagsportalen utformade vi på egen hand, därför kommer kraven på den från oss själva. Vi ville skapa en säker och användbar portal där användaren hela tiden har kontroll över vad som sker. När det gäller säkerhet var ett krav att ingen obehörig skulle kunna ta sig in på sidorna.

Goodwill ville att vi skulle bibehålla företagets profil, till exempel använda de färger och typsnitt som förknippas med dem, så mycket som möjligt.
1.2 Uppgift

Examensarbetet definierades av vissa mål som sattes upp och sedan skulle uppfyllas. Det är viktigt att avgränsa ett projekt noggrant i början så att uppgiften inte flyter ut och blir ohanterbar. Det är också bra om metoden för att nå målen bestäms i tidigt skede så att idéerna sedan är möjliga att genomföra.

1.2.1 Mål

Examensarbetet hade två mål, dels en hemsida i Flash och dels en företagsportal åt företaget Goodwill Reklambyrå.

Det blev beslutat att hemsidan skulle göras helt i Flash MX. Ingen av oss hade tidigare arbetat med Flash så ett av målen med examensarbetet kan givetvis sägas vara att lära oss så mycket om programmet som möjligt. Med hjälp av Flash gick det att skapa en rörlig sida med snygg design som fångar en potentiell kunds intresse, men stor vikt lades även på att sidan skulle vara överskådlig och ha en god navigering. Användbarhet var ett nyckelord uppbyggnaden av hemsidan.

Målet med företagsportalen var att i ASP skapa en portal, där användaren kan skriva till, ändra och ta bort information i en databas, med hjälp av VBScript och SQL. Inte heller detta var något vi tidigare arbetat med så ett av målen var att vi skulle kunna programmera ASP efter avslutat arbete. Portalen skulle göras säker, det vill säga att ingen kan ta sig in utan att vara behörig. För att göra detta användes en lösenordsskyddad inloggning och en kontroll på varje sida för att kontrollera att användaren verkligen är inloggad och inte kommit dit på något annat sätt.

Företagsportalen var tänkt att vara till för att Goodwill lätt skulle kunna administrera sina uppgifter om deras kunder och anställda via en webbsida på Internet. En fördel med att sköta all databashantering via en webbsida är att alla inblandade kan komma åt det som finns i databasen på vilken dator som helst, enda kravet är att den måste vara uppkopplad mot Internet. Användarna (Goodwills personal) behöver alltså inte ha databasprogrammet eller själva databasen lagrad på datorn. Detta tyckte vi kunde passa Goodwill bra då de jobbar mycket på distans och kanske använder sig av flera olika datorer beroende på var de befinner
sig. Meningen med företagsportalen var att det skulle gå lätt att se vad som finns i databaserna och även att kunna uppdatera dem.

1.3 Metod

Goodwills hemsida skulle som sagt göras helt i Flash MX, endast en grå ram runt Flash-movien skulle vara HTML-baserad. Denna ram skapades i Dreamweaver. Hemsidan skulle innehålla en hel del rörelse, till exempel animerade små intron till varje undersida och animerade knappar. Något som smidigt kunde göras i Flash.

När det gällde innehållet på sidan användes delvis material från Goodwill, bland annat texter som presenterar företaget och en del bilder från Goodwills tidigare kunduppdrag. Övriga bilder och även grafiska inslag skulle främst skapas i Flash men även i 3ds max. Sidan skulle nämligen bland annat innehålla korta 3d-animerade inslag. All eventuell bildbehandling skulle göras i Photoshop.

1.4 Planering

Examensarbetet består av tre delar: hemsidan, företagsportalen och rapporten. Först gjordes hemsidan och efter det företagsportalen. Parallellt med detta arbete fördes anteckningar som sedan användes som underlag till rapporten. Rapportskrivande påbörjades när hemsidan nästan var klar, slutförandet av rapporten skedde dock under sommaren. Examensarbetet beräknades vara klart i slutet av augusti då vår framläggning skulle ske, vilket det också var.

Det första steget var att lära in så mycket som möjligt om de olika tekniker som sedan skulle används. Efter det planerades det hur hemsidan och företagsportalen skulle se ut och fungera. Vi studerade Goodwills dåvarande hemsida noga och tittade även på olika sidor på Internet för att få inspiration till hur den nya hemsidan skulle se ut. Det var även viktigt att både hemsidan och företagsportalen fick ett bra användargränssnitt.

När det förberedande arbetet var klart försökte vi hitta sätt att genomföra våra mål rent praktiskt. Genom böcker, manualer och olika tester kom de bästa lösningarna på problemen fram. Hela arbetsprocessen sammanställdes sedan i rapporten.

Under arbetets gång hade vi fortlöpende kontakt med vår kontaktperson Peter Fällman från Goodwill och vår examinator Niklas Rönnberg från Linköpings universitet.

1.4.1 Tidsplan

Vecka 15-17  Möten med Goodwill och examinator.
             Spåna på idé till hemsidan.
             Göra en tidsplan.
Vecka 18-20  Skapa hemsidans intron.
Vecka 21    Lägga in texter på hemsidan.
             Fundera ut strukturen på rapporten.
Vecka 22    Skapa undersidan *Kunder*.
Vecka 23-25  Skapa företagsportalen.
Vecka 26-33  Skriva rapporten.
1.5 Rapportens uppbyggnad

Det är viktigt att läsaren hela tiden kan hänga med i rapporten och inte behövde stanna upp och bläddra tillbaka för att förstå sammanhanget. Därför är det bra om han eller hon redan från början har fått en överblick över vad rapporten innehåller och hur den är uppbyggd.

1.5.1 Syfte

Syftet med denna rapport var att ge läsaren en klar inblick i hur arbetet fortskred. Hela arbetets gång skulle beskrivas, från planeringsstadiet till själva utförandet. Avslutningsvis skulle det färdiga resultatet sammanfattas och diskuteras.

1.5.2 Struktur

Rapporten består av tre stora kapitel; Inledande arbete, Från tanke till färdigt resultat och Diskussion. Rapporten avslutas med några sammanfattande ord i kapitlet Avslutning.


Från tanke till färdigt resultat består av tre delkapitel, Teori, Förberedelser och Utförande. I Teoridelen beskrivs bakgrundsteori till de program och programmeringsspråk som användes i arbetet, nämligen ASP, Flash, Dreamweaver och 3ds max. Delkapitlet Förberedelser beskriver hur designen och användargränssnittet till hemsidan och företagsportalen planerades fram. I Utförandedelen visas det hur det praktiska arbetet genomfördes steg för steg.

2 Från tanke till färdigt resultat


2.1 Teori

Här följer en beskrivning av de program och programmeringsspråk som användes i examensarbetet.

2.1.1 Active Server Pages (ASP)

I detta kapitel är viss fakta hämtad från boken Active Server Pages 3.0 & databaser på Internet\(^1\) och från webbsidan Idg.se\(^2\).


---

\(^1\) Arvidsson Stefan, Ek Jesper och Eriksson Ulrika (2000)

\(^2\) http://www.idg.se/webstudio

För att servern skall kunna skilja scripten från HTML-koden i ASP-dokumentet måste scripten omges av taggar, <&% och %>. Det behövs inget särskilt program för att programmera ASP utan det räcker med en textredigerare, men det är viktigt att alla webbsidor med ASP måste ha ändelsen .asp och inte .htm eller .html som webbsidor vanligen har.

Rent praktiskt fungerar ASP så här:

1. En användare efterfrågar en sida som innehåller ASP-kod.
2. ASP-dokumentet körs genom serverns programvara. Programvaran läser scripten som finns innanför taggarna och utför det scripten ber om.

2.1.1.1 ASP-objekt

Det finns sju inbyggda objekt i ASP som läses in när en ASP-sida kommer till en webbserver. Tre av objektten förklaras närmare nedan:

- **Request**
- **Response**
- **Session**

Objektet **Request** tar hand om den information som skickas från användaren till servern medan objektet **Response** tar hand om den information som skickas åt andra hållet, det vill säga från server till användaren. Dessa två objekt utgör med andra ord kommunikationslänken för all informationstrafik mellan klienten och servern.

Två olika samlingsobjekt tillhörande **Request** är **Form** och **QueryString**. **Form** utnyttjas bland annat vid olika formulär och **QueryString** används då information som inte kommer från ett formulär behöver skickas med till exempel en länk.

För att få information från en användare används ofta ett formulär som skall tas emot och behandlas på servern. **Form** består av värden från objekt i ett formulär, vilket innebär att det på så sätt går att komma åt information som skickas från formuläret. Formuläret skapas i HTML och i dess tagg anges det vart formuläruppgifterna skall skickas (**Action**) och på vilket sätt (**Method**). Det finns två olika metoder, **Post** och **Get**. Det vanligaste och bästa är **Post**, anledningen till att det är bättre är att vid användning av **Get** läggs formulärinformationen i URL-strängen och är alltså synlig för klienten. En annan anledning är att med **Post** går det skicka obegränsad mängd data vilket det inte gör med **Get**. Formulärets olika objekt (textruttor, radioknappar, kryssruttor och så vidare) innehåller flera olika typer av kontroller som har olika egenskaper. Kontrollen **Name** används till exempel som referens för att hämta ett specifikt värde från formuläret. För att skicka formulärinformationen används oftast en knapp som skickar iväg informationen till sidan den skall och med angiven metod. Mottagarsidan tar emot informationen och **Request** tar hand om informationen. Det som **Request** gör är att den hämtar innehållet från formulärets kontroll **Name** med **Form** och laddar ner denna information i egna variabler. Innehållet är vad som finns i kontollen **Value**. Dessa
variabler är efter det tillgängliga för mottagarsidan att kontrollera, skriva ut eller liknande. 
QueryString användes för att skicka med variabler direkt i URL:en.


2.1.1.2 ASP, databaser och SQL

För att kunna hantera en databas med ASP krävs ett databasobjekt att arbeta mot. I ASP heter detta objekt ActiveX Data Object (ADO) och är kontaktytan mellan ASP-sidan och databasen. ADO innehåller endast tre objekt för att vara så litet som möjligt och på så sätt vara hanterbart på Internet. Två vanliga objekt som tillhör ADO beskrivs nedan:

- Connection
- Recordset

ConnectionString består av information om var databasen ligger lagrad och vad för sorts databas det är, och med metoden Open så öppnas Connection-objektet.


SQL är ett frågespråk för att prata med databaser. De mest grundläggande kommandona är SELECT, FROM, WHERE, INSERT, DELETE och UPDATE. Med SELECT och FROM går det att välja ut poster ur en tabell i en databas. Är det endast vissa av posterna i databasen som är intressanta går det att lägga till kommandot WHERE vilket anger villkor för de poster som skall visas. Kommandot INSERT infogar nya poster i en tabell, DELETE raderar poster i en tabell och UPDATE uppdaterar värdet på en eller flera poster i en tabell.
Det finns många tillämpningar med ASP, här följer några förslag på konkreta användningsområden:

- Databashantering via webbsidor, exempelvis hämta, lagra och ändra information.
- Skapa spel, gästböcker, chattar och annonsmarknader.
- Lagra information om en användare som används varje gång användaren kommer till sidan för att till exempel kunna påminna han eller hon om något.
- Göra mer interaktiva sidor, till exempel visa dagens datum eller låta användaren välja storlek på texten.

2.1.2 Flash

Till detta kapitel är viss fakta hämtad från boken Flash 4 Bible³ och från Macromedias hemsida⁴.


Flash skapar två sorters filer, först en ”arbetsfil”, med filändelsen .fla, som innehåller all information om utseende och uppsyggnad av movien. Det är den filen som användaren utformar i Flash. ”Arbetsfilen” publiceras sedan till en ”filmfil”, med filändelsen .swf. ”Filmfilen” innehåller endast informationen som behövs för att spela upp filen. Själva uppsyggnaden av movien sker på olika nivåer som alla följer samma tidslinje. En movie kan ha en eller flera scener som kan arrangeras för att skapa den spelordning användaren är ute efter. Scenerna i sin tur består av ett oberoende antal lagrar. Hur dessa lagrar visas i scenen bestäms i tidslinjen, de övre lagren i tidslinjen visas över de underliggande. Även lagren kan

---

³ Reinhardt, Robert och Warren Lentz, Jon 2000
⁴ http://www.macromedia.com
på samma sätt ha en nivåskillnad på hur de objekt som ingår i ett lager visas i scenen. De enheter som i tidslinjen skall representera tiden i en animation kallas för frames. Det finns framför allt två viktiga typer av frames, keyframes och statiska frames. Enkelt uttryckt kan man säga att i keyframes placeras eller ändras innehållet (de kan även vara helt tomma) medan de statiska endast repeterar innehållet i förgående frame. (Mer om keyframes och statiska frames i stycket om animation) Den färdiga movien innehåller ofta grafik, text, animation och olika applikationer för hemsidor. Det mesta i en movie är uppbyggt med vektorgrafik, mer om det senare, men de kan även innehålla video, bitmappar och ljud som importerats in till filen. För att en användare skall kunna se en Flash-movie krävs att en Flash-spelare installeras, den går att hämta gratis på Macromedias hemsida. Spelaren visar då movien i webbläsaren eller som en egen fristående applikation.

2.1.2.1 Vektorbaserat ritprogram

2.1.2.2 Animering

2.1.2.3 ActionScript


Några exempel slutligen på vad Flash används till idag:

- Animerade inslag på hemsidor, till exempel rörliga knappar eller logotyper.
- Animerade filmer eller liknande som skall visas över Internet.
- Hela Flash-hemsidor, utan HTML-baserade grafik eller textinslag.

2.1.3 Dreamweaver


Det är som sagt inte alls nödvändigt att kunna HTML, det går att skapa webbapplikationer i Dreamweaver utan att behöva skriva en rad med kod. Är användaren inne i den visuelle editormiljön finns menyer med olika element att lägga in på sidan som till exempel tabeller, knappar eller olika inmatningsfält. Färger, typosnitt och liknande väljs från menyer på samma sätt och det går enkelt att importera bilder med mera. För användare som vill arbeta direkt med koden finns en kod-editor med diverse verktyg och hjälpmedel, till exempel tag

Dreamweaver stödjer serverspråk som till exempel ASP (Active Server Pages) vilket gör att användaren kan skapa dynamiska databasbaserade applikationer och det går även att importera olika objekt skapade i Macromedia Flash.

2.1.4 3ds max

3ds max är ett program skapat för att modellera, animera och rendera tredimensionella modeller. Programmet är utvecklat av Discreet, första versionen presenterades 1996 och i skrivande stund är version 5 av 3ds max snart klart för utgivning. 3ds max är idag en mycket populär objektorienterad plattform som används till exempel för skapa visuella effekter av olika slag, karaktärsanimationer eller grafik till dataspel. För att så enkelt som möjligt förklara hur 3ds max arbetar så går vi nu kortfattat igenom de tre stora bitarna, modellering, animering och rendering.

2.1.4.1 Modellering

Varje objekt tilldelas i 3ds max som default varsin färg, det ligger ingen direkt estetisk tanke bakom detta utan det sker mer för att det skall vara lätt för användaren att skilja objekten åt. Polygonernas ytor, de som är synliga för användaren, täcks då av färgen men det går även att ställa in så att bara polygonernas kantlinjer syns. Detta kallas för objektets *mesh*. Vill användaren att ett objekt skall täckas av ett mönster istället för bara en enda färg används en *textur*. En textur är en bitmap som kan skapas till exempel i ett bildbehandlingsprogram och sedan importeras till 3ds max, användaren tilldelar objektet en textur som då kläs runt objektet.

![Figur 1. Ett objekt uppbryggt av olika antal polygoner.](image1)

![Figur 2. Ett objekt som visas med en färg, med mesh och med textur.](image2)

### 2.1.4.2 Animering

Liksom Flah animerar 3ds max med hjälp av *frames*. Användaren bestämmer hur objekten i scenen skall vara placerade i en viss frame, en såkallad *keyframe*, vill användaren sedan att objekten skall röra sig markerar han eller hon en ny frame som får fungera som keyframe och flyttar objekten till en ny position. 3ds max skapar automatiskt framsen mellan de två keyframsen, vilket då blir själva rörelsen. Denna typ av animering blir självfallet inte speciellt exakt, den kan vara svår att styra över. Det finns därför även möjlighet att programmera en animering, detta är betydligt smidigare om användaren till exempel skall animera ett objekt
som skall cirkla runt ett annat objekt i en exakt bana. Sådana rörelser är bättre att definiera rent matematiskt.


2.1.4.3 Rendering

Under tiden en scen bearbetas i 3ds max ser användaren endast en förenklad version av det färdiga resultatet. Filstorleken måste vara liten annars skulle det bli alldeles för tungt för datorn att arbeta med, därför visas inte texturer, skuggor, reflektioner med mera förrän efter en rendering. Under renderingen tuggas en scen igenom frame för frame, det vill säga är det endast en frame renderas en stillbild och är det en animering renderas flera stillbilder efter varandra. I renderingen läggs alla texturer på objekten och effekterna av ljussättningen in i scenen, 3ds max räknar ut bland annat vilka skuggor och reflektioner det blir i bilden. Alla andra effekter som användaren kan ha ställt in syns nu också, detta blir alltså den färdiga filen. Det finns många inställningar för rendering så att användaren kan få exakt det resultat han eller hon är ute efter. Användaren kan bland annat välja hur hög bildkvaliteten skall vara och vilken sorts fil, videoformat eller bitmappformat till exempel, som det färdiga resultatet skall vara.
2.2 Förberedelser

Innan det praktiska arbetet satte igång skapades hemsidan och företagsportalen på papper. Designen skulle vara klar innan datorn kom in i bilden och även användargränssnitt skulle vara väl genomtänkt.

2.2.1 Design


---

\(^5\) http://www.aldoshoes.com

\(^6\) http://www.nominerad.com
är i den som allt innehåll på hemsidan kommer. De sex knapparna är länkar till de undersidor som finns på hemsidan.

- **Hem**
- **Affärsidé**
- **Organisation**
- **Tjänster**
- **Kunder**
- **Kontakt**

Varje knapp fick bestå av två ringar och namnet på undersidan. För att användaren lätt skall kunna se på vilken undersida han eller hon är på så ändrar ringarna, på den aktuella knappen, färg från grå till röd. Det skapades även en animering (den yttre ringen ändrar storlek) på knappen för att liva upp sidan och den startas när musen är över en knapp. Alla dessa grundlayoutelement lades i en centrerad vit rektangel som blev omgärdad av en ljusgrå färg.

![Figur 3. Grundlayouten på hemsidan.](image-url)

De informerande texterna som Goodwill använde till varje undersida fanns att tillgå och skulle användas även i detta arbete. Texterna presenterades i gråa transparenta rutor med en mörkgrå ram runtom och skrevs i olika toner av mörkgrått. Texten i teextrutorna visas inte förrän teextrutan är på plats och ”rullas” då ner så att användaren kan se den vilket ger en fin effekt. I vissa rutor fick inte all text plats på en gång, det lades då in klickbara pilar så att användaren själv kan mata fram nästa textstycke. Det kändes som den snyggaste lösningen och gav även fördelen att textmängden lätt kan varieras om Goodwill till exempel vill lägga till mer text.
2.2.1.1 Startintro

Ett kort startintro skapades för att en ny användare som kommer in på hemsidan utan tvekan ska veta vilket företags hemsida han eller hon surfat in på. Under introt visas Goodwills logotyp mitt på sidan med deras slogan ”Ny tid Ny byrå” för att presentera Goodwill för användaren. För att få in lite liv och rörelse i introt animerades knappmenyn så att knapparna åker in från sidorna och studsar till sina rätta platser.

2.2.1.2 Hem


![Figur 4. Hem](image-url)
2.2.1.3 Affärsidé

På Affärsidésidan lades sloganen "reklamkrig mediaexplosion ...i begränsningen röjer sig mästaren” tvärs över sidan. Även här fick vissa av orden blinka, ”krig” och ”explosion”, för att användaren särskilt ska lägga märke till dem. Mellan orden lades en 3d-bild på fyra krigande skepp, ett vinrött Goodwillskepp och tre gråa skepp från andra reklamföretag. Detta ska symbolisera att det ensamma Goodwillskeppet är mästaren som röjer sig. På den högra sidan placerades textrutan med dess informationstext.

![Affärsidé](image)

**Figur 5. Affärsidé**

2.2.1.4 Organisation

Lägst upp på Organisationssidan placerades sloganen "Framgång genom att underhålla en relation –inte genom att försörja en hel armé". Till vänster på sidan lades en 3d-film på två människor som skakar hand som zoomas in vilken är tänkt att symbolisera relationen i sloganen. För att lyfta fram de två viktigaste orden, ”relation” och ”hel armé”, i sloganen fick de gå in i ett vinrött sjök på höger sida. Textrutan lades i mitten på sidan under sloganen.
2.2.1.5 Tjänster

2.2.1.6 Kunder


![Figur 8. Kunder](image_url)

2.2.1.7 Kontakt

Till vänster på Kontaktsidan lades ett kort på Goodwills VD och ägare Peter Fällman. På höger sida placerades två textrutor under varandra, en för allmän information om Peter Fällman och en för Goodwills kontaktinformation. För att framhäva deras e-postadress och
lätta upp sidan skrevs den i vinrött. Under dessa textutor lades ytterligare en länk, även den i vinrött, ”Intresserad?” vilken går till en intresseanmälan.

Figur 9. Kontakt

På företagsportalen däremot hölls designen enkel då det primära med portalen är funktionen och inte formen. Målet var att Goodwills personal skulle känna sig som ”hemma” och trivas på sidorna. Längst upp på alla sidorna lades därför en vinröd rektangel med Goodwills logotyp inuti som ramar in innehållet på sidorna och även gör så att användarna känner igen sig från sida till sida. För att få informationen från databaserna strukturerad och lättöverskådlig fick den presenteras i tabeller. All text skrevs med Eurostilefonten och färgerna fick bestå av en gråskala och vinrött för att hålla portalen enhetlig med den övriga hemsidan.

2.2.2 Användbarhet

Något som är mycket viktigt när en applikation skall struktureras upp, i detta fall då en hemsida och en företagsportal, är användbarhet. Det gäller att tänka på bland annat vem som kommer att använda applikationen, är det en användare med liten datorvana eller är det någon med extremt god vana att hantera applikationer? Kommer en blivande användare att kunna förstå applikationen direkt eller skall det behövas en manual? Att ha kunskap om den tänkta målgruppen är väldigt viktigt om en applikation skall bli just användbar, liksom även att ha ett hum om hur vi människor tar till oss information och vilka begränsningar vi har. Här följer en
beskrivning över hur hemsidan och företagsportalen planerades upp ur ett användarvänligt perspektiv.

Målgrupperna som applikationerna skulle rikta sig till var olika för hemsidan och företagsportalen, när det gällde hemsidan så var det i första hand olika företag som skall använda sig av den. Användarna hade i det fallet en medelgod till god datorvana, det vill säga de har använt datorer tidigare och har varit inne på diverse andra hemsidor. När det gällde företagsportalen så var de anställda på Goodwill Reklambyrå själva målgruppen och eftersom de arbetar med datorer dagligen och då ofta över nätverk, har de en mycket god datorvana.


Undersidorna heter i tur och ordning:

- **Hem**
  Detta är startsidan, den första sida användaren möter. Därför kommer användaren här att finna en kort intresseväckande företagspresentation och information om de tjänster Goodwill Reklambyrå erbjuder sina kunder.

- **Affärsidé**
  Här presenteras Goodwills affärsidé korthårt.

- **Organisation**
  På sidan kallad Organisation kommer en beskrivning över hur Goodwill arbetar att finnas. Användaren får en inblick i hur en eventuell affärsuppgörelse skulle utformas och hur deras ärende skulle behandlas.
• **Tjänster**
  Här får användaren en mer preciserad beskrivning av de tjänster Goodwill tillhandahåller. Tjänsterna delas upp i tre grupper för att vara mer lättöverskådliga.

• **Kunder**
  Under *Kunder* läggs en förteckning över några av Goodwills tidigare kunduppdrag in. Användaren kan titta på reklambilder och även få en kort beskrivning av uppdragen.

• **Kontakt**
  På denna sida kommer användaren att få information om hur han eller hon skall ta kontakt med företaget, det kommer även att finnas möjlighet att registrera sitt intresse direkt över nätet. Under *Kontakt* kommer det även att finnas en presentation av Peter Fällman som äger Goodwill Reklambyrå.

![Figur 10. Menyraden där Hem-knappen är aktiverad.](image)


![Figur 11. Goodwills logotyp.](image)

Figur 12. Ikonerna för musiken, playknappen och stoppknappen.


När en användare först loggar in på företagsportalen kommer han eller hon till en startsida med en meny där de olika alternativen över de tjänster som företagsportalen erbjuder är

Som avslutning vill vi bara säga att "reglerna" som vi tar upp i detta stycke för hur en användbar applikation skall byggas upp, inte på något sätt är några lagar som måste följas helt och fullt. Däremot tror vi att det kan vara bra att ha kunskap och vara medveten om dem, känner man till reglerna går det ju även att bryta mot dem.
2.3 Utförande

När det var dags att förverkliga målen rent praktiskt var alla de förberedelser vi hade gjort till stor glädje. Det mesta var väl planerat så det var bara att sätta igång.

2.3.1 Hemsidan

Till hemsidan användes i huvudsak tre olika program, Flash, Dreamweaver och 3ds max. Det var intressant att sammanföra olika tekniker och utnyttja det som lärt in i skolan tillsammans med nyvunna kunskaper.

2.3.1.1 Upplägg


När grunden var gjord var det tid att skapa själva hemsidan i Flash. Eftersom vi inte hade använt oss av Flash särskilt mycket innan började vi med att läsa in oss på programmet i olika handböcker och följde även vissa inbyggda lektioner för att lära oss grunderna. Hemsidan var sedan tidigare, se stycket om Planering, noggrant utformad rent designmässigt och även hur den skulle fungera var bestämt. Nu var det dags att lista ut det smidigaste sättet att genomföra det som planerats vilket gjordes utifrån de nyvunna kunskaperna.

För att inte filen sedan skulle bli för tung för att kunna köras utan problem över Internet testades hur många bilder som kunde ingå i filen, och framför allt hur stora filstorlekar dessa bilder kunde ha. Långa laddningstider skulle undvikas, något som Goodwill själva poängterat vikten av. Undersidan Kunder, vilken skulle innehålla många bilder skapades som en fil för sig och länkades in till den stora ”grundfilen”, som då innehåller alla de andra undersidorna inklusive startsidan. Detta gjorde att laddningstiden blev uppdelad och inte så stor. Första
tanken var att låta alla undersidor med tillhörande intron fungera på samma sätt men efter några tester visade det sig att det inte behövdes. Filstorleken höll sig ändå väl inom rimliga gränser.

2.3.1.2 Hemsidan tar form

Det var viktigt att lägga upp en bra strukturering av sidan i Flash. En god ordning skulle underlätta om det eventuellt skulle behöva göras några ändringar eller liknande i senare skeenden. Varje undersida fick en mapp i tidslinjen där allt som var specifikt för just den undersidan, såsom textrutor och bilder, kunde läggas i olika lager. I det understa lagret, under alla andra mappar, placerades en mapp kallad Grund innehållande det som skulle finnas med på alla sidor. Till exempel menyraden, ramen och rektangeln där logotypen och musikknapparna placerades.

Vissa avbilderna som behövdes till hemsidan skapades direkt i Flash och var då vektoruppbryggda. Andra bilder hade företaget bidragit med och ytterligare några andra skapades i Photoshop. De både senare typerna av bilder importerades in till Flash och hamnade då i filens bibliotek, liksom i tidslinjen fick varje undersida en mapp där bilderna som skulle användas till enbart den undersidan placerades.

För att filstorleken på en Flash-movie inte skall bli onödigt stor är det bra att göra grafiska inslag till symboler. Det innebär att användaren kan använda sig av samma bilder och annat flera gånger utan att för den skulle behöva importera objektet igen. Det finns ett grundobjekt lagrat och sedan kan flera instanser av grundobjektet göras. Det finns i Flash tre typer av symboler; movieclip, button och graphic. Animerade objekt får bli movieclip och tilldelas då en egen tidslinje, ett objekt som skall bli klickbar får bli en button-symbol och stillbilder och annat låter man vara graphic. Symboler användes väldigt mycket i arbetet, till exempel när knappmenyn skapades. Två olika varianter på knappmenyn gjordes, i det inledande introt användes graphic-symboler som animerades direkt i scenen. Efter startintrot blir knapparna klickbara, då användes button-symboler. En button-symbol har fyra lägen (states); Up, Down, Over och Hit. I Up-läget, där användaren bestämmer hur knappen kommer att se ut i sitt ursprungliga läge, lät vi knapparna vara grå liksom i Down-läget, alltså då användaren klickar på knappen. I Over-läget, när muspekaren befinner sig över knappen lades en animering in och knapparna byter även färg från grått till vinrött. I Hit-läget bestämmer man träffytan för
knappen, träffytan gjordes lite större än själva knappen, detta för att underlätta för användaren.

Alla undersidorna skulle följa samma struktur, vilket bestämts redan på planeringsstadiet. De skulle börja med ett intro och för att sedan avslutas som en statisk sida. Grunden, det vill säga ramen och knapparna skulle följa med på alla sidor och inte ändra utseende något. Allt innehåll som är specifikt för undersidan, all information, skulle visas i ramen. All skriven text presenteras i gråa textrutor och vi lade även in bilder till varje undersida. (Mer information i kapitlet Från tanke till färdigt resultat under stycket Förberedelser)


Till introna animerades olika stora färgade sjok, som åker fram och tillbaka i olika hastigheter, detta justerade vi med hjälp av keyframes i tidslinjen. En start- och en stopppoint bestämdes och sedan skapade Flash själv de mellanliggande framsen. Sjoken skapades direktt i Flash som graphic-symboler. Förutom sjoken innehåller introna tredimensionella inslag, vilket vi tar upp mer om i nästa stycke. För att movien skall stanna och bli statisk vid varje undersida lades ett ActionScript in som ser till att filmen stannar på rätt ställen och även sätter igång igen på rätt frame när användaren klickar sig vidare.
2.3.1.3 Tredimensionella inslag


När modellerna var färdiga, allt som allt fyra rymdskepp och en människomodell, började skapandet av de fyra scenerna:

- Till Hem skulle det vara ett vinrött rymdskepp, det skepp som fick symbolisera Goodwill, som vrider sig som för en uppvisning.
• Till **Affärsidé** skulle det vinröda skeppet igen användas igen och dessutom tre grå rymdskepp av samma typ som skulle symbolisera "fienden". Skeppen skulle stå stilla men skjuta på varandra.

![3D-modellen till Affärsidé.](http://www.3dcafe.com)

• Till **Organisation** behövdes två människor, en grå modell och en vinröd modell, som skulle skaka hand. Figurerna skall stå stilla men en inzoomning och en mindre vridning skulle ske.

![3D-modellen till Organisation.](http://www.3dcafe.com)
Till *Tjänster* skulle tre statiska bilder på tre olika typer av rymdskepp, det rymdskepp vi använt till *Hem* och *Affärsidé* samt två andra användas.

![3D-modellen till Tjänster.](image)


När scenerna såg bra ut animerades objekten i den rörelse som de skulle ha i de olika introna, även kamerorna animerades för att till exempel få inzoomningar och kameraåkningar. Keyframes placerades ut i tidslinjen och där bestämdes vilken position objekten och kamerorna skulle ha vid exakt den tidpunkten. Sedan skapade 3ds max själv de mellanliggande framesen.

När animeringen och scenerna var färdiga testades några olika slags renderingar. Till att börja med renderades de animerade scenerna som filmer i avi-format, men då blev filerna...
oacceptabelt stora. Därför renderades till slut alla, även de animerade scenerna, som bitmappar som sedan kunde läggas in i Flash, en bitmapp för varje ny frame.

2.3.1.4 Kundsidan
Två av undersidorna, Kunder och Kontakt skiljde sig en del från de övriga, de skulle inte ha samma typ av intron som de andra undersidorna och de skulle inte bestå av enbart de tidigare beskrivna textrutorna och bilder. De skulle vara mer interaktiva.


2.3.1.5 Kontaktsidan
På undersidan Kontakt lades en länk till ett formulär in där användaren kan fylla i bland annat namn och adress för att beställa en broschyr om Goodwill. Uppgifterna som användaren skickar lagras i en databas, mer om detta i nästa stycke där vi går igenom skapandet av företagsportalen.
2.3.2 Företagsportalen


Figur 17. Strukturen på företagsportalen.

2.3.2.1 Databaserna

*Kunder* innehåller information om deras nuvarande kunder. Tabellen i databasen innehåller följande fält:

- Företag
- Kontaktperson
- E-post
- Telefon
- ID
Framtida kunder innehåller information om eventuella kunder som är intresserade av ett uppdrag. Tabellen i databasen innehåller följande fält:

- Namn
- Företag
- Adress
- Postadress
- E-post
- ID

Anställda/Användare innehåller information om de anställda på Goodwill, databasen används även till att lagra inloggningsnamn- och lösenord och behörighet för användarna av företagsportalen. Tabellen i databasen innehåller följande fält:

- Namn
- Användarnamn
- Password
- E-post
- Access
- ID

Av säkerhetsskäl får inte alla användare se alla fält i databaserna och inte heller komma åt alla List- och ändringssidor. Därför är alla användare uppdelade i två behörighetsklasser, A och B, dessa klasser sparas i Anställda/Användare-databasen under fältet Access. De med A-behörighet kommer till en webbsida och de med B-behörighet kommer till en annan, på så sätt går det att kontrollera vem som får komma åt vilka sidor. I Goodwills företagsportal kommer användare med A-behörighet åt alla sidor och kan till exempel både ta bort och ändra i databasen, medan en användare med B-behörighet endast kommer åt de sidor där det går att se vad som finns i databasen. Med den här lösningen gick det att undvika att okunniga användare gör saker som inte går att ångra, till exempel att de råkar ta bort en kund och därmed även viktiga uppgifter om den kunden. Då det även går att bestämma vilka fält som
skall visas på Listsidorna kan känsliga uppgifter som databasen eventuellt innehåller ”gömmas” från B-behöriga, till exempel de övriga användarnas lösenord. På flera av sidorna i portalen används information från databaserna vilket gör att en kommunikation mellan databaserna och portalen måste upprättas. Denna kommunikation fungerar likadant på alla sidor där databasinformation används och skrivs i början på sidan. I portalen användes databasobjektet ADO vid skapandet av en kontaktyta mellan ASP och databasen. Med dess objekt Connection och metoden CreateObject (se bilaga A rad 01) skapas en förbindelse och med metoden Open (se bilaga A rad 02) öppnas den aktuella databasen. På alla sidor som använder sig av information från någon av databaserna använder även ett Recordset. Därför skapas det i detta steg en instans av ett Recordset för fortsatt valfri användning (se bilaga A rad 03). Efter dessa obligatoriska rader kod skrivs koden för vad som ska göras på sidorna in. En förklaring på vad som händer och vad som är skrivet i koden på alla sidorna i portalen följer nedan.

2.3.2.2 Default- och inloggningssidan

Defaultsidan fungerar som en startsida för användarna och det är här inloggningen till företagsportalen sker. Sidan är skriven i HTML och består av ett inloggningsformulär, innehållande ett användarnamnsfält och ett lösenordsfält, och en ”Logga in”-knapp (se bilaga B raderna 06-35).

![Figur 18. Den första sidan användaren kommer till, defaultsidan.](image)

När användaren fyllt i sitt användarnamn och lösenord och tryckt på ”Logga in”-knappen skickas uppgifterna med metoden Post till Inloggningssidan, det vill säga att formulärets action är namnet på Inloggningssidan (se bilaga B rad 05). På denna sida kontrolleras det att de skickade formuläruppgifterna (användarnamnet och lösenordet) är giltiga. För att komma åt uppgifterna används Request och Form. Genom att jämföra de skickade uppgifterna med uppgifterna i fälten ”Användarnamn” och ”Password” från databasen Anställda/Användare
går det att kontrollera om den skickade informationen finns i databasen. Alla posterna i
databasen väljs med SQL-satsen SELECT och FROM och med WHERE går det att få fram
om de skickade uppgifterna stämde med databasuppgifterna (se bilaga C rad 04). Vid felaktigt
användarnamn och/eller lösenord visas ett felmeddelande och användaren får försöka igen (se
bilaga C rad 20). Skriver användaren däremot in ett korrekt användarnamn och lösenord sker
ytterligare en kontroll där det kontrolleras vilken behörighet användaren har innan han eller
hon slussas vidare till rätt behörighetssida. Användaren kommer alltså aldrig logiskt till
Inloggningssidan eftersom den endast består av olika script och ingen HTML, istället skickas
användaren automatiskt till A- eller B-behörighetsidan med metoden Redirect (se bilaga C
raderna 09-17).

2.3.2.3 Behörighetssidorna
Det finns alltså två olika Behörighetssidor i Goodwills företagsportal, en för A-behöriga och
en för B-behöriga. Dessa två sidor innehåller endast länkar som för användaren vidare till de
olika List- och ändringssidorna och är skrivna helt i HTML. De två Behörighetssidorna
innehåller följande länkar:

A-behörighet

Framtida kunder
Lista
Kunddatabas
Lista
Lägga till kund
Ändra kunduppgifter
Ta bort kund
Användare
Lista
Skapa ny användare
Välkommen Peter!

Kunder
  Kundregister
  Lägga till ny kund
  Andra kunduppgifter
    Ta bort kund

Framtida kunder
  Förteckning över framtida kunder

Användare
  Förteckning över användare
  Lägga till ny användare

Logga ut

**Figur 19.** Sidan där alla länkar till list- och ändringssidorna finns för A-behöriga.

**B-behörighet**

Framtida kunder
  Lista

Kunddatabas
  Lista

Anställda
  Lista

---

Du är nu inloggad!

Kunder
  Kundregister:

Framtida kunder
  Förteckning över framtida kunder

Goodwill
  Förteckning över anställda

Logga ut

**Figur 20.** Sidan där alla länkar till list- och ändringssidorna finns för B-behöriga.
2.3.2.4 List- och ändringssidorna

Det finns fem olika List- och ändringsfiler, två för endast A-behörighet (Kunder och Användare), två för endast B-behörighet (Kunder och Anställda) och en gemensam för Framtida kunder-listan eftersom den är lika för de båda behörighetssidorna. Denna uppdelning gjordes för att få en enkel administration och struktur av filerna. Lösningen innebar dock ett litet problem eftersom inte allt i filerna skulle köras vid alla tillfällen. Beroende på vilken länk användaren klickar på så skall endast den delen i filen utföras och inget mer. Därför fick varje del i filen inledas med ett if-villkor som måste uppfyllas för att den delen skall få köras. En förutbestämd och unik textsträng skickas med från varje enskild länk på Behörighetssida vilken har en matchning i de olika if-villkoren. Den medskickade strängen läses in med Request och QueryString och kontrolleras i if-satserna, där villkoret uppfylls där utförs if-satsen. Under till exempel Användare för A-behöriga finns det två länkar, en länk för att se en lista på användarna och en länk för att skapa en ny användare. All information om de båda sidorna ligger alltså i samma fil men i varsin if-sats. Om användaren klickar på länken lista så skickas strängen ”lista” och eftersom if-satsen som lista-informationen ligger i utförs om strängen är lika med just ”lista” så visas Listasidan och inte Skapa ny-sidan (se bilaga D rad 8 och bilaga E rad 9).

2.3.2.4.1 Lista

Flera av List- och ändringssidorna är just Listsidor och de fungerar alla på samma sätt. Med en SQL-sats (SELECT och FROM) väljs de berörda posterna i databasen ut och sparas med ASP ner i ett Recordset. Uppgifterna i Recordsetet loopas med MoveNext igenom rad för rad och visas för användaren i en tabell som är skriven i HTML (se bilaga E raderna 14-40).

Kundregister

<table>
<thead>
<tr>
<th>Företag</th>
<th>Kontaktperson</th>
<th>E-post</th>
<th>Telefon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Glasstruten AB</td>
<td>Hanna Fornberg</td>
<td><a href="mailto:haff0524@student.liu.se">haff0524@student.liu.se</a></td>
<td>011-126196</td>
</tr>
<tr>
<td>God som godis</td>
<td>Julia E. Flockvist</td>
<td><a href="mailto:julia123@student.liu.se">julia123@student.liu.se</a></td>
<td>011-188979</td>
</tr>
<tr>
<td>Manna coffeeshop</td>
<td>Marie Drott</td>
<td><a href="mailto:mard040@student.liu.se">mard040@student.liu.se</a></td>
<td>011-026256</td>
</tr>
<tr>
<td>Stefans plattor</td>
<td>Stefan Parson</td>
<td><a href="mailto:stpaso380@student.liu.se">stpaso380@student.liu.se</a></td>
<td>011-136285</td>
</tr>
<tr>
<td>Peters ställ</td>
<td>Peter Wallin</td>
<td><a href="mailto:petwall205@student.liu.se">petwall205@student.liu.se</a></td>
<td>011-131720</td>
</tr>
<tr>
<td>Martins senen</td>
<td>Martin Flockvist</td>
<td><a href="mailto:marta@hotmail.com">marta@hotmail.com</a></td>
<td>019-360947</td>
</tr>
</tbody>
</table>

Figur 21. Sida där alla kunder visas i en lista.
2.3.2.4.2 Lägga till


![Formulär för att lägga till en ny kund](image-url)

Figur 22. Sida för att lägga till en ny kund.

Formulärsinformation skickas likadant som inloggningsuppgifterna med formulärets action och metod och tas sedan hand om av en Request och Form. Databasen som informationen skall läggas till i väljs med en SQL-satsen SELECT och med ASP (Recordset och AddNew) skapas det en ny post i databasen där den nya informationen läggs in (se bilaga E raderna 111-121). När den nya posten är på plats skickas användaren automatiskt till en ny sida där uppdateringen eller de skickade uppgifterna bekräftas (se bilaga E 107-110). Intresseanmälningen har några extra finesser, en är att personen måste fylla i vissa obligatoriska uppgifter annars visas en varning. Detta kontrolleras via en särskild funktion som kontrollerar att det finns några tecken inskrivna i de obligatoriska fälten annars visas ett felmeddelande. Den andra är att när uppgifterna är skickade till databasen skickas även automatiskt en e-post till en förutbestämd adress för att informera Goodwill om att en ny intresserad har skickat in sin anmälan från hemsidan.
2.3.2.4.3 Ändra

Det finns en sida för att ändra kunduppgifterna i kunddatabasen och endast de A-behöriga kommer åt den. För att användaren enkelt ska kunna ta fram rätt kund så ligger alla kunderna i en valmeny där användaren väljer den berörda kunden. Valmenyn skapas av en ASP-loop som går igenom alla poster i kunddatabasen och lägger dem i en rullgardinslista, för att få en överskådlig lista syns endast namnet på kunden (se bilaga E raderna 129-143).

**Figur 23.** Rullgardin för att välja en kund vars data ska uppdateras.


**Figur 24.** Sida för att ändra data hos en kund.
När de nya uppgifterna är inskrivna får användaren klicka på en "Uppdatera”-knapp. Dessa uppgifter sparas i ett *Recordset* som skickas vidare till själva uppdateringen av databasen. Här väljs den berörda kundposten ut med en SQL-sats och med *Recordset* och *Update* skrivs de gamla uppgifterna över de nya (se bilaga F raderna 6-12). Nu är databasen uppdaterad och användaren skickas till en sida med texten "Kundinformationen är nu uppdaterad”.

2.3.2.4.4 Ta bort


![Radera kund](image)

**Figur 25.** Sida för att radera en kund.

För att verkligen radera kunden måste användaren trycka på en "Ta bort”-knapp, då skickas ID-numret som legat dolt i tabellen vidare till SQL-satsen DELETE som raderar kunden ur databasen (se bilaga G raderna 8-9). Anledningen till att ID-numret måste följa med hela vägen är för att det är det enda unika fältet i databasen, vilket garanterar att det är helt säkert att rätt kund raderas. Användaren skickas därefter till en sida som bekräftar att kunden är raderad.

På alla List- och ändringssidor finns det en "Tillbaka”-knapp som för användaren tillbaka till Behörighetssidan. Det är bra då användarna lätt kan komma till länkarna om de till exempel
klickat på fel länk. För att slippa lägga in detta manuellt på varje List- och ändringssida sparades dessa ”Tillbaka”-knappar i en särskild fil inkluderades på varje sida. Beroende på vilken inloggning användaren har (A eller B) så skall han eller hon komma tillbaka till olika Behörighetssidor. Detta löstes genom att en if-sats kontrollerar vilken inloggning användaren har och på så sätt dirigera användaren till rätt sida (se bilaga H raderna 2-10).

2.3.2.5 Säkerhet

3 Diskussion

I detta kapitel diskuteras det slutgiltiga resultatet och vilka problem som stötdes på under arbetets gång. Även olika framtidsmöjligheter för både hemsidan och företagsportalen diskuteras.

3.1 Resultat

Vi hade som sagt satt upp vissa mål som vi ville uppfylla under arbetets gång. Till exempel att hemsidan skulle vara rörlig och ha en snygg design men samtidigt vara överskådlig samt ha en god navigering. När det gällde företagsportalen var det viktigast att den blev säker och hade ett bra användargränssnitt, själva designen fick komma i andra hand. Här följer en diskussion om hur väl vi har uppfyllt våra olika mål.

3.1.1 Hemsidan

Vi planerade layouten av sidan mycket väl och kunde därför få sidan att se ut på det hela taget precis som vi ville. Eftersom design är något väldigt individuellt är det svårt att bedöma hur bra en design egentligen är, vi tycker dock att hemsidan fyller sin funktion rent stilmässigt. Något som däremot är mätbart är hur användbar en sida är. Vi kommer nu att göra en liten utvärdering av hur användbar vår hemsida är och vi tittar särskilt på följande kriterier:

- Navigering
- Färg och text
- Enhetlighet
- Kognitiv belastning

Att en sida har en god navigering är mycket viktigt. Vi tycker att vi har lyckats bra, hemsidan har en bra struktur och det är enkelt för användaren att se på vilken undersida han eller hon befinner sig. Det är även lätt för användaren att ta sig tillbaka till startsidan eftersom det både finns en textlänk och en logotyp som fungerar som länk till startsidan. Färgerna på sidan, det vinröda och gråskalan, är som vi tidigare nämnt hämtade från Goodwills logotyp. När vi planerade designen bestämde vi att vi ville ha en rätt så ljus sida, även fast grundfärgerna är mörka. Därför använde vi mycket vitt vilket bidrog till att sidan lättades upp och det är lätt för
en användare att se innehållet på sidan då kontrasten mellan bakgrund och förgrund blir skarp. Däremot kunde kontrasten mellan text och textruta vara större då typsnittet vi använder oss av inte är optimalt för att läsas på en dataskärm. Anledningen att vi ändå valde dessa färger är att det blev snyggare rent designmässigt. Vi anser att vi lyckats få sidan enhetlig eftersom själva ramen med menyer och logotyp följer med på alla undersidorna och eftersom introna håller samma stil. Vi såg från början till att inte överbelasta sidan med information. Eftersom det endast finns sex länkar på sidan så tycker inte vi att de rörliga introna stör användaren.

Vi arbetade hela tiden med att hålla nere laddningstiderna på sidan. Vi testade oss fram till hur många bilder, och framför allt hur stora filstorlekar dessa bilder kunde ha, för att inte filen sedan skulle bli för tung för att kunna köras utan problem över Internet. Resultatet blev en relativt lätt sida som alla användare kan titta på utan problem.

Allt som allt är vi nöjda med hur hemsidan blev, vissa detaljer såsom texten i textrutorna skulle dock kunna förbättras. Vi hade i tidigt stadium planerat att lägga in en bakgrundsmusik till hemsidan. På grund av tidsbrist så hann vi tyvärr aldrig att göra denna låt. Däremot testade vi att lägga in en färdig musikslinga bara för att prova våra musikknappar.

3.1.2 Företagsportalen

Även vid skapandet av företagsportalen tyckte vi att det var viktigt att få ett bra användar-gränssnitt. Vi kommer nu att bedöma portalen efter samma kriterier som hemsidan:

- Navigering
- Färg och text
- Enhetlighet
- Kognitiv belastning

På företagsportalen använde vi oss av ett annat navigeringssystem. Varje undersida har en överskrift som talar om vilken undersida användaren är inne på. Detta funkar bra i det här fallet eftersom antalet undersidor är så lågt, hade det handlat om en stor hemsida hade vi använt oss av en annan teknik. På alla undersidor finns även en knapp tillbaka till startsidan. Bakgrundsfärgen på portalen blev vit och eftersom texterna är skrivna i mörkgrått och vinrött så syns de trots typsnittet väl, vilket vi tyckte blev väldigt bra. Portalen är enhetlig då
logotypen följer med på alla undersidor och gör att användaren hela tiden har en känsla av att befinna sig inloggad på portalen. Menyn på startsidan innehåller många objekt och vi delade därför upp länkarna i tre grupper, vilket gör menyn mer överskådlig för användaren.


Det är svårt att veta om vi kodmässigt har gjort en bra programmering eftersom det var vår första ASP-applikation och inte hade något att jämföra med. Vi valde att dela upp scripten i olika filer vilket vi tyckte gav en lättöverskådlig struktur. Det blev även lättare för oss att göra ändringar i efterhand, till exempel lägga in sessioner, på det viset.
3.2 Svårigheter och problem

Vårt första problem var att Goodwill redan hade ett tema och slogans som de ville att även vi skulle använda. Detta gjorde att vi kände att vi inte hade så mycket utrymme för egna idéer utan blev låsta i deras tankesätt, å andra sidan var det ju deras hemsida så självklart fick vi rätta oss efter dem. De första veckorna då vi planerade upp hur sidan skulle se ut utgick vi mycket efter de bilder som vi fått av dem, detta visade sig senare vara något av ett problem då det var oklarheter med upphovsrätten till bilderna. Vi fick då ändra våra planer helt och istället för att använda färdiga bilder skapa egna vilket både drog ut på tiden och tvingade oss att planera om sidan.

De nya bilderna ville vi skapa i 3ds max och vi bestämde oss för att animera korta filmsnuttar som vi kunde använda i introna, vilket ger mer liv och rörelse jämfört med statiska bilder. Dock ville vi som sagt hålla nere filstorleken och då filmer renderade i 3ds max blir väldigt stora hade vi ett problem. Detta löste vi genom att istället för att rendera en hel film renderade vi filmen frame för frame som bitmappar. Bitmaparna lade vi sedan in i serier efter varandra i Flash, det blev då samma effekt som om vi skulle ha importerat filmer. Eftersom det inte handlade om några långa animeringsbitar rent tidmässigt så fungerade detta bra och de animerade sekvenserna blev inte så tunga att köra. Vi drog ner på filstorleken ytterligare genom att vi gjorde om bilderna från tiff till gif i Photoshop. När vi gjorde om dem till gif drog vi samtidigt ner på antalet färger som skulle användas i bilderna och därmed fick vi ner filstorleken till en bra nivå. En annan sak vi gjorde som drog ner på filstorleken var att animera skotten till Affärsidé-introt i Flash istället för i 3ds max. Då kunde vi använda oss av endast en renderad bild och sedan skapa två typer av vektorbaserade skott som symboler i Flash och sedan använda oss av de två för att animera alla skotten.
3.3 Framtid


3.3.1 Hemsidan


Två av undersidorna behöver vidareutvecklas för att vi inte fick tillräckligt med information i tid. Tjänster saknar viss information och på Kunder går det att lägga till fler beskrivningar av kunduppdrag.

3.3.2 Företagsportalen

Det finns många framtidsmöjligheter med denna portal främst eftersom den är en slags prototyp. Då ingen av Goodwills riktiga databaser använts innebär det att om företagsportalen skall användas professionellt så måste det göras en anpassning av innehållet i portalen. Själva grunden i ASP-kodningen finns och beroende på vilka databaser företaget vill lägga ut och vad som skall kunna göras med dem går det utöka företagsportalen relativt lätt. Det innebär även att utökningsmöjligheterna är stora, på vår prototyp går det till exempel att göra det möjligt att ändra och ta bort information om ”Framtida kunder” och ”Användare/Anställda”. En annan framtidsmöjlighet är att lägga in mer information om kunderna, till exempel projektinformation såsom vilka anställda som gjort vilka projekt. Skulle denna företagsportal användas i verkligheten är det, tycker vi, även viktigt att lägga in fler varningstexter om användaren gör något fel eller innan något raderas ur databasen.
4 Avslutning

Vi tycker att det här har varit ett intressant och lärorikt examensarbete, vi är glada att vi valde just de uppgifter som vi gjorde. Att vi valde att lära oss Flash och ASP från grunden har varit en nyttig erfarenhet. Vi känner att det nu är mycket lättare för oss att få grepp om nya tekniker.

Det har varit spännande att arbeta nära ett medieföretag, det har gett oss en fin inblick i arbetslivet. Eftersom vi båda i framtiden tänker oss att arbeta på ett företag som arbetar med media på ett eller annat sätt så var det väldigt bra att se hur ett sådant företag fungerar och hur de arbetar.


Efter genomfört examensarbete känner vi oss faktiskt mer redo för arbetslivet!
5 Referenser

Arvidsson, Stefan; Ek, Jesper och Eriksson, Ulrika
Active Server Pages 3.0 & databaser på Internet (2000)
Pagina Förlags AB

Reinhardt, Robert och Warren Lentz, Jon
Flash 4 Bible (2000)
IDG Books Worldwide, Inc
ISBN 0-7645-3356-8

Active Server Pages (ASP)

Macromedia
http://www.macromedia.com (senast besökt 2002-07-27)

ALDO
http://www.aldoshoes.com (senast besökt 2002-05-08)

Nominerad.com

3d café
http://www.3dscafe.com (senast besökt 2002-05-15)
Bilaga A

Databaskoppling

<%  
1.  Set Connect = Server.CreateObject("ADODB.Connection")  
2.  Connect.Open "provider=Microsoft.Jet.OLEDB.4.0; Data Source=" & server.mappath("users.mdb")  
3.  Set RecSet = Server.CreateObject("ADODB.Recordset")  
%>
Bilaga B

Defaultsidan

1. <html>
2. <title>Inloggning</title>
3. <p><img src="logotyp.jpg" width="400" height="76"><font size="2" face="eurostile, verdana, arial"></font></p>
4. <font size="2" face="eurostile, verdana, arial">
5. <form method="POST" action="login.asp">
6. <table width="399" border="0" cellspacing="0" cellpadding="0">
7. 
8. <tr>
9.   <td width="174">
10.     <div align="right"><font size="2" face="eurostile, verdana, arial">Användarnamn</font></div>
11.   </td>
12.   <td width="225" height="30">
13.     <div align="right"><font size="2" face="eurostile, verdana, arial">
14.       <input type="text" size="30" name="username" style="font-size: 8pt; font-family: verdana, arial"></font></div>
15. 
16. </tr>
17. 
18. <tr>
19.   <td width="174">
20.     <div align="right"><font size="2" face="eurostile, verdana, arial">Lösenord:</font></div>
21.   </td>
22.   <td width="225" height="30">
23.     <div align="right"><font size="2" face="eurostile, verdana, arial">
24.       <input type="password" size="30" name="password" style="font-size: 8pt; font-family: verdana, arial"></font></div>
25. 
26. </tr>
27. 
28. <tr>
29.   <td width="174">&nbsp;</td>
30.   <td width="225" height="40">
31.     <div align="right"><font size="2" face="eurostile, verdana, arial">
32.       <input type="submit" value="Logga in" style="font-size: 8pt; font-family: verdana, arial" name="submit"></font></div>
33. 
34. </tr>
35. </table>
36. <p>&nbsp;</p>
37. <p>&nbsp;</p>
38. </form>
39. </font> </html>
Bilaga C

Inloggninssidan

1. <!-- #include file="adovbs.inc" -->

2. <% Set Connect = Server.CreateObject("ADODB.Connection")
   Connect.Open "provider=Microsoft.Jet.OLEDB.4.0; Data Source=" & 
   server.mappath("users.mdb")
   Set RecSet = Server.CreateObject("ADODB.Recordset") %>

3. <%

4. Visa = "SELECT * FROM users WHERE Username=" & request.form("username") & ""
   AND Password=" & request.form("password") & ""
5. RecSet.Open Visa, Connect, adOpenStatic, adLockOptimistic

6. if RecSet.recordcount > 0 then

7. Session("username") = Request.Form("username")
8. Session("password") = Request.Form("password")

9. 'Inloggning för A-access
10. If RecSet("Access") = "A" Then
11. Session("access") = "A"
12. Response.Redirect "ABehor.asp"

13. End if

14. 'Inloggning för B-access
15. If RecSet("Access") = "B" Then
16. Session("access") = "B"
17. Response.Redirect "BBehor.asp"

18. End if

19. Else

21. End if

22. RecSet.Close
23. set rs = nothing
24. Connect.Close
25. set Connect = nothing
26. %>
Bilaga D

Behörighetssida

1. <!-- #include file="koll1.inc" -->
2. <html>
3. <body link="#666666" alink="#9C0D07" vlink="#9C0D07">
4. <p><img src="logotyp.jpg" width="400" height="76"></p>
5. <p align="left"><font size="4" face="eurostile, verdana, arial">Välkommen Peter!</font></p>
6. </font></p>
7. <p align="left"><font face="eurostile, verdana, arial" size="3"-Kunder<br></font></p>
8.  
9. <font size="2" face="eurostile, verdana, arial" color="#666666">Kundregister</a><br>
10. <a href="kundA.asp?alt=createuser">Lägga till ny kund</a><br>
11. <a href="kundA.asp?alt=alteruser">Ändra kunduppgifter</a><br>
12. <a href="fram.asp?alt=viewusers">Framtida kunder</a><br>
13. <a href="anvA.asp?alt=viewusers">Användare</a><br>
14. <a href="anvA.asp?alt=createuser">Lägga till ny
19. användare<br />
20. <a href="loggaut.asp">Logga ut</a> 
21. </body>
22. </html>
23. <!-- #include file="koll3.inc" -->
Bilaga E

List- och ändringssidor

1. <!-- #include file="adovbs.inc" -->
2. <!-- #include file="koll1.inc" -->
3. <html>
4. <title>Goodwill reklambyrå</title>
5. <% Set Connect = Server.CreateObject("ADODB.Connection")
6. Connect.Open "provider=Microsoft.Jet.OLEDB.4.0; Data Source=server.mappath("kunder.mdb")"
7. Set RecSet = Server.CreateObject("ADODB.Recordset") %>
8. <%'KUNDER - Förteckning%>
9. <% If Request.QueryString("alt") = "viewusers" Then %>
10. <p> <img src="logotyp.jpg"></p>
11. <p> <font size="4" face="eurostile, verdana, arial">Kundregister</b></font> <font size="1" face="eurostile, verdana, arial">%
12. </font>
13. <font size="1" face="eurostile, verdana, arial">%
14. <% Visa = "SELECT * FROM kunder"
15. RecSet.Open Visa, Connect, adOpenStatic, adLockOptimistic %>
16. <table border="0" cellspacing="0" width="565" height="15">
<table>
<thead>
<tr>
<th>Företag</th>
<th>Kontaktperson</th>
<th>E-post</th>
<th>Telefon</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Företag Value]</td>
<td>[Kontaktperson Value]</td>
<td>[E-post Value]</td>
<td>[Telefon Value]</td>
</tr>
</tbody>
</table>

```asp
<% Do While Not RecSet.EOF %>
<tr>
<td width="138"><font size="2" face="eurostile, verdana, arial">Företag</font></td>
<td width="140"><font size="2" face="eurostile, verdana, arial">Kontaktperson</font></td>
<td width="177"><a href="mailto:<%=RecSet("E-post") %>">
<%=RecSet("E-post") %></a></td>
<td width="110"><%=RecSet("Telefon") %></td>
</tr>
<% RecSet.MoveNext %>
Loop %>
</table>

<p>
<ul>
</ul>
</p>
<% RecSet.Close %>
End If %>
```
47. `<%KUNDER - Skapa ny%>`

48. `<% If Request.QueryString("alt") = "createuser" Then %>`

49. `<p><font size="1" face="eurostile, verdana, arial"><b><img src="logotyp.jpg" width="400" height="76"></b></font></p>`

50. `<p><font size="1" face="eurostile, verdana, arial"><b><font size="4">Lägga till ny kund</font></b></font></p>`

51. `<form method="POST" action="kundA.asp?alt=create">`

52. `<table width="402" border="0" cellspacing="0" cellpadding="0">`

53. `<tr>`

54. `<td height="30" width="187">`

55. `<div align="right"><font size="2" face="eurostile, verdana, arial"><b>Företag</b></font></div>`

56. `</td>`

57. `<td width="215" height="30">`

58. `<div align="right"><font size="2" face="eurostile, verdana, arial">`

59. `<input type="text" size="30" name="foretag" style="font-size: 8pt; font-family: verdana, arial">`

60. `</font></div>`

61. `</td>`

62. `<tr>`

63. `<td height="30" width="187">`

64. `<div align="right"><font size="2" face="eurostile, verdana, arial"><b>Kontaktperson</b></font></div>`

65. `</td>`

66. `<tr>`

67. `<td width="215" height="30">`

68. `<div align="right"><font size="2" face="eurostile, verdana, arial">`

69. `<input type="text" size="30" name="kontaktperson" style="font-size: 8pt; font-family: verdana, arial">`

70. `</font></div>`

71. `</td>`

72. `<tr>`

73. `<td height="30" width="187">`

74. `<div align="right"><font size="2" face="eurostile, verdana, arial"><b>E-post</b></font></div>`

75. `</td>`
76. </td>
77. <td width="215" height="30">
78. <div align="right"><font size="2" face="eurostile, verdana, arial">
79. <input type="text" size="30" name="epost" style="font-size: 8pt; font-family: verdana, arial">
80. </font></div>
81. </td>
82. </tr>
83. <tr>
84. <td width="187" height="30">
85. <div align="right"><font size="2" face="eurostile, verdana, arial"><b>Telefon:</b></font></div>
86. </td>
87. <td width="215" height="30">
88. <div align="right"><font size="2" face="eurostile, verdana, arial">
89. <input type="text" size="30" name="telefon" style="font-size: 8pt; font-family: verdana, arial">
90. </font></div>
91. </td>
92. </tr>
93. <tr>
94. <td width="187">
95. <div align="right"></div>
96. </td>
97. <td width="215" height="40">
98. <div align="right"><font size="2" face="eurostile, verdana, arial">
99. <input type="submit" value="Lägg till ny kund" style="font-size: 8pt; font-family: verdana, arial" name="submit">
100. </font></div>
101. </td>
102. </tr>
103. </table>
104. </form>
105. </p>
106. </font>
107. <% ElseIf Request.QueryString("alt") = "create" Then %>
do while not RecSet.EOF

%>

<option value="<%=RecSet("ID")%>"><%=RecSet("Företag")%>

<% RecSet.MoveNext loop RecSet.Close %>

<input type="submit" value="Gå vidare">

</select>

</form>

<% End If %>

<% 'KUNDER - ta bort %>

<% If Request.QueryString("alt") = "deleteuser" Then %>
<p><img src="logotyp.jpg"></p>
<p><font size="4" face="eurostile, verdana, arial">Ta bort kund</font></p><font size="1" face="eurostile, verdana, arial">

<% Visa = "SELECT Företag, ID FROM kunder"
RecSet.Open Visa, Connect, adOpenStatic, adLockOptimistic %>

<form action="delete.asp" method="post">
<select name="val">
<option SELECTED>Välj företag
<% %>
do while not RecSet.EOF

%>
<option value=""><%=RecSet("ID")%>"=""><%=RecSet("Företag")%>

<!-- #include file="koll2.inc" -->
Bilaga F

Huvudändringssida

1. <!-- #include file="koll1.inc" -->
2. <!-- #include file="adovbs.inc" -->
3. <html>
4. <body link="#666666" alink="#9C0D07" vlink="#9C0D07">
5. <% Set Connect = Server.CreateObject("ADODB.Connection")
   Connect.Open "provider=Microsoft.Jet.OLEDB.4.0; Data Source=\& server.mappath("kunder.mdb")
   Set RecSet = Server.CreateObject("ADODB.Recordset")
%>
6. <% If Request.Querystring(\"alt\") = \"alter\" Then %>
7. <p><font size="2" face="eurostile, verdana, arial"> <b><img src="logotyp.jpg" width="400" height="76"></b></font></p>
8. <p><font size="2" face="eurostile, verdana, arial"> <b><font size="4">Kundinformationen är nu uppdaterad!</font></b></font></p>
9. <%  
10. Visa = "SELECT * FROM kunder WHERE ID = " & Request.Form(\"val\")
11. RecSet.Open Visa, Connect, adOpenStatic, adLockOptimistic
   
   RecSet("Företag") = Request.Form("uForetag")
   RecSet("Kontaktperson") = Request.Form("uKontakt")
   RecSet("E-post") = Request.Form("uEpost")
   RecSet("Telefon") = Request.Form("uTelefon")
   RecSet.Update
12. RecSet.Close
13. Else %>
14. <p><img src="logotyp.jpg"></p>
15. <p><font size="4" face="eurostile, verdana, arial">Uppdatera informationen</font></p>
16. <font size="1" face="eurostile, verdana, arial"></font></p>
17. <%= Visa = "SELECT * FROM kunder WHERE ID = " & Request.Form("val")
18. RecSet.Open Visa, Connect, adOpenStatic, adLockOptimistic %>
19. <table border="0" cellpadding="0" cellspacing="0" width="565" height="15">
20. <tr>
21. <td width="159"><font size="2" face="eurostile, verdana, arial">Företag</font></td>
22. <td width="157"><font size="2" face="eurostile, verdana, arial">Kontaktperson</font></td>
23. <td width="156"><font size="2" face="eurostile, verdana, arial">E-post</font></td>
24. <td width="93"><font size="2" face="eurostile, verdana, arial">Telefon</font></td>
25. </tr>
26. </table>
27. </font>
28. <form method="post" action="alter.asp?alt=alter">
29. <input type="text" value="" name="uForetag">
30. <input type="text" value="" name="uKontakt">
31. <input type="text" value="" name="uEpost">
32. <input type="text" value="" name="uTelefon">
33. <input type="hidden" value="" name="val">
34. <input type="submit" name="Submit" value="Uppdatera">
35. </form>
36. </body>
37. </html>
38. <% End if %>
39. <!-- #include file="koll2.inc" -->
Bilaga G

Huvudraderingssidan

1. <!-- #include file="koll1.inc" -->
2. <!-- #include file="adovbs.inc" -->
3. <html>
4. <% Set Connect = Server.CreateObject("ADODB.Connection")
   Connect.Open "provider=Microsoft.Jet.OLEDB.4.0; Data Source=& server.mappath("kunder.mdb")
   Set RecSet = Server.CreateObject("ADODB.Recordset") %>
5. <% If Request.QueryString("alt") = "delete" Then %>
6. <p><font size="2" face="eurostile, verdana, arial">b><img src="logotyp.jpg" width="400" height="76"></b><font size="4">Kunden är nu raderad!</font></p>
7. <% If Request.QueryString("alt") = "delete" Then %>
8. <%= "DELETE * FROM kunder WHERE ID = " & Request.Form("val")
9. Connect.execute(Visa)
10. Else %>
11. <body link="#666666" alink="#9C0D07" vlink="#9C0D07">
12. <p><img src="logotyp.jpg"></p>
13. <%= "SELECT * FROM kunder WHERE ID = " & Request.Form("val")
14. RecSet.Open Visa, Connect, adOpenStatic, adLockOptimistic%>
<form method="post" action="delete.asp?alt=delete">
<table>
<tr>
<td width="117"><font size="2" face="eurostile, verdana, arial"><b>Företag</b></font></td>
<td width="146"><font size="2" face="eurostile, verdana, arial"><b>Kontaktperson</b></font></td>
<td width="146"><font size="2" face="eurostile, verdana, arial"><b>E-post</b></font></td>
<td width="141"><font size="2" face="eurostile, verdana, arial"><b>Telefon</b></font></td>
</tr>
<tr>
<td width="117"><font size="2" face="eurostile, verdana, arial">% =RecSet("Företag") %></td>
<td width="146"><font size="2" face="eurostile, verdana, arial">% =RecSet("Kontaktperson") %></td>
<td width="146"><font size="2" face="eurostile, verdana, arial">% =RecSet("E-post") %></td>
<td width="141"><font size="2" face="eurostile, verdana, arial">% =RecSet("Telefon") %
    a. <input type="hidden" value="% =request.form("val") %" name="val">
</font></td>
</tr>
<tr>
<td width="117"><font size="2" face="eurostile, verdana, arial"></font></td>
<td width="146"><font size="2" face="eurostile, verdana, arial"></font></td>
<td width="146"><font size="2" face="eurostile, verdana, arial"></font></td>
<td width="141"></font></td>
</tr>
<tr>
<td width="117"><font size="2" face="eurostile, verdana, arial"></font></td>
<td width="146"><font size="2" face="eurostile, verdana, arial"></font></td>
<td width="146"><font size="2" face="eurostile, verdana, arial"></font></td>
<td width="141"></font></td>
</tr>
</table>
<input type="submit" name="Submit" value="Ta bort">
</form>
<div align="right"><font size="2" face="eurostile, verdana, arial"> </font></div>
</td>
</tr>
</table>
</form>
</body>
</html>

<% End if %>

<%! #include file="koll2.inc" -->
Bilaga H

Inkluderingsfiler

Koll 1

1. <% if Len(Session("Username")) > 0 Then %>

Koll 2

2. <% if Session("Access")="A" then %>
3. <font size="2" face="eurostile, verdana, arial" color="#666666"><a href="ABehor.asp">Tillbaka</a></font><br>

4. <%
5. else
6. %>
7. <font size="2" face="eurostile, verdana, arial" color="#666666"><a href="BBehor.asp">Tillbaka</a></font><br>

8. <%
9. End if
10. %>

11. <% Else
12. Response.redirect "default.asp"
13. End if %>