

CHALMERS



Interaktivt stöd vid introduktion av nyanställda

*Master of Science Thesis [in the Programme Information
Engineering]*

NIKLAS KIHLM-FORSBERG

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Göteborg, Sweden, november 2009

The Author grants to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet.

The Author warrants that he/she is the author to the Work, and warrants that the Work does not contain text, pictures or other material that violates copyright law.

The Author shall, when transferring the rights of the Work to a third party (for example a publisher or a company), acknowledge the third party about this agreement. If the Author has signed a copyright agreement with a third party regarding the Work, the Author warrants hereby that he/she has obtained any necessary permission from this third party to let Chalmers University of Technology and University of Gothenburg store the Work electronically and make it accessible on the Internet.

Interaktivt stöd vid introduktion av nyanställda

NIKLAS KIHLM-FORSBERG

© NIKLAS KIHLM-FORSBERG, November 2009.

Examiner: OLOF TORGERSSON

Department of Computer Science and Engineering
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000
Department of Computer Science and Engineering
Göteborg, Sweden, November 2009

Sammanfattning

Det är oftast upp till chefen att introducera en nyanställd i företaget och lotsa runt denne bland arbetsplatser och nya arbetskamrater. Det är den här processen som det här arbetet har behandlat, utvecklat och försökt underlätta. Dels för chefen och dels för den nyanställde.

I undersökningen har en litteraturstudie, en teknisk analys, en iterativ prototyputveckling och en implementering av en prototyp ingått. I litteraturstudien har fokus legat på områdena webbdesign och applikationsdesign. Prototyputvecklingen har gått genom tre iterationer, från pappersprototyper till digitala bilder som använts som mall under implementeringen av den slutgiltiga applikationen.

Applikationen har utvecklats i .NET och C# mot en SQL databas. Dessutom används JavaScript för ett par funktioner samt CSS för att underlätta designen.

Resultatet är en fungerande prototyp där användaren kan logga in, utföra de uppgifter som har blivit tilldelade, svara på ett quiz och utvärdera processen. Huvuddelen av arbetet har legat på att ta fram ett så intuitivt gränssnitt som möjligt vilket har inneburit att andra viktiga delar i en komplett applikation har lämnats till framtida utveckling. Arbetet har också lämnat ett par rekommendationer till vad som bör utvecklas vidare om applikationen ska fungera till sin fulla rätt.

Abstract

The job of introducing a new employee is often up to the director of the company. It is his mission to guide the employee through co-workers and new environments and to see that he or she settles in correctly. This master thesis develops an application that is to assist both the employer and the employee in this process.

The project has consisted of a literature study, a technical analysis and both prototyping and a prototype implementation. The literature study has focused on two areas, web design and application design. The prototyping went through three iterations, from paper prototyping to digital images used during the implementation of the web application.

The web application was developed in .NET and C# using a SQL database as storage. In addition to this there are some JavaScript's used as well as CSS to help the design.

The result is a working prototype of the web application where a user is able to login, perform the tasks available, answer a quiz and then evaluate the process. Since this work has concentrated on interactivity and user interface design a lot of other work has been left aside, like security and similar important aspects. This leaves a couple of recommendations for improvement needed to be made before the application is to be complete.

Förord

Det här examensarbetet är den sista delen i civilingenjörsutbildningen inom IT, med inriktning mot interaktionsdesign, vid Institutionen för Data och Informationsteknik (D & IT). Utbildningen som är på 270 högskolepoäng har genomförts vid Chalmers Tekniska Högskola (CTH) och vid IT-universitetet på Lindholmen.

Examensarbetet utfördes hos Koneo & H2Data i Mölndal mellan våren och hösten 2008. Jag skulle speciellt vilja tacka Hans Hansson och min handledare Stefan Klöfver vid Koneo & H2Data för råd och hjälp samt visat tålamod med uppgiften.

Till det här examensarbetet har Olof Torgersson vid D & IT fungerat som examinator.

Niklas Kihl-Forsberg

Innehållsförteckning

1 Inledning	9
1.1 Problemformulering	10
1.2 Syfte och mål	10
1.3 Avgränsning	111
2 Webb- och applikationsdesign	12
2.1 Vad är en webbapplikation?	12
2.2 Design av webbapplikationer	13
2.3 Designdirektiv för webbapplikationer	14
2.4 Gestaltprinciper	15
2.5 Applikationsdesign	17
2.6 Webbdesign	18
3 Teknisk analys	21
3.1 Microsoft SharePoint	21
3.2 Fristående applikation	22
4 Metodbeskrivning	21
4.1 Prototyputveckling	24
4.2 Intervjuer	26
4.3 Användningsfall	28
4.4 Genomförande	30
5 Resultat	31
5.1 Processen	31
5.2 Applikationens delar	33
6 Diskussion	37
6.1 Design	37
6.2 Framtida arbete	38
6.3 Egna tankar	39
Referenser	41
Bilaga A - Intervjufrågor	41
Bilaga B - Processen	41
Bilaga C - Användningsfall	41

1 Inledning

1990 startade H2 med konsultverksamhet och försäljning av datorer och datortillbehör, främst åt flertalet skolor i regionen. Med IT-branschens snabba utveckling så var H2 tvungna att byta fokus med jämna mellanrum, dels för att behålla sin flexibilitet men också för att fortsätta vara konkurrenskraftiga på marknaden. 1997 strukturerad man om företagets hårdvaruverksamhet och inriktade sig även mot större kunder, företag och kommuner. Dessutom ökades tjänstedelen och servicenivån kraftigt.

Idag sysselsätter H2-gruppen ca 30 anställda inom två affärsområden, H2 Solutions och Koneo & H2Data.



Figur 1 och 2 Logotyper för Koneo & H2Data och H2 Solutions.

H2-gruppen är via bolaget H2 Data AB med i Koneokedjan – En av Sveriges största kedjor för försörjning till kontor inom områdena data, dokumenthantering och inredning. H2 Solutions AB hjälper till med IT-strategier, levererar heltäckande IS/IT-lösningar och bistår med utbildning, support och service.

H2 gruppen är en komplett leverantör av IT-tjänster och produkter och kan även titulera sig Microsoft Gold Certified Partner inom flera områden.

1.1 Problemformulering

Introduktionen av en nyanställd inom ett företag är ofta ett chefsansvar. Chefen ser till att den nyanställde tas emot och introduceras på sin arbetsplats och i organisationen. Till sin hjälp har chefen vanligen material som stöd, material som beror på vilken position den nyanställde har och var i organisationen placeringen ska ske.

I större organisationer och företag med en omfattande hierarki av anställda och chefer så kan det uppstå svårigheter med den här sortens introduktioner. Oftast får den nyanställde låna tid och resurser av arbetskamrater och kollegor för att komma in i rutiner. Dessutom kan det ur chefs synpunkt vara svårt att samla all information som ska delges den nyanställde utan att ibland missa viktiga delar.

Min handledare Stefan Klöfver har i en dokumentation om en möjlig medarbetarintroduktion (2003) gjort följande konceptbeskrivning:

"Upplägget i den interaktiva introduktionen bygger på att man skapar en process som tar hand om introduktionen på ett strukturerat sätt. Indirekt så sätter man igång en process som stödjer den nyanställde i den fortsatta utvecklingen i organisationen och som utgör ett stöd i kommande medarbetarsamtal. Dokumentationen från introduktionsperioden kan sedan användas som underlag inför medarbetarsamtalet."

Målsättningen med en interaktiv medarbetarintroduktion är alltså att den nyanställde snabbt ska kunna komma in i rutiner och få tillgång till aktuell och relevant information. Genom att erbjuda en sådan här introduktionskurs så säkerställer man även att samma information lämnas till alla nyanställda, samtidigt som chefen får en process och ett system att luta sig emot.

1.2 Syfte och mål

Enkelt förklarat är målet med det här examensarbetet att framställa en fungerande prototyp till en tänkbar framtida medarbetarintroduktion. Arbetet startas inte från scratch utan kommer att bygga på idéer och dokumentationsmaterial som min handledare Stefan Klöfver har arbetat fram.

Efter önsknings från H2 Data om att implementera programmet i SharePoint så kommer en teknisk analys att genomföras för att se om detta är möjligt. Analysen kommer att utreda om det dels är tekniskt möjligt att genomföra och dels om det ligger inom tidsramen för det här arbetet att implementera programmet i det här systemet.

Vidare så kommer huvudmålet med arbetet att ligga i att ta fram en så användbar applikation som möjligt, vilket medför att interaktionsdesign, grafik och gränssnitt kommer att ha högsta prioritet. Designen kommer dels att baseras på de kurser som har behandlats det senaste året inom linjen interaktionsdesign och dels på en lite djupare studie i webb- och applikationsdesign som kommer att göras under arbetets gång.

Ett personligt mål med det här arbetet är att få en känsla av hur det är att gå från idéstadiet till en färdig prototyp i ett projekt och att få applicera några av de kurser man har läst genom åren i ett skarpt läge och på en riktig arbetsplats. Arbetet kommer dessutom att ge en bättre överblick av utvecklingsprocessen i ett projekt, vilket säkert kommer att bli en värdefull erfarenhet när man väl kommer ut i arbetslivet.

1.3 Avgränsning

Med tanke på tidsramen för examensarbetet så kommer applikationen att behandla den nyanställdes första månad inom företaget och processen som denne ska gå igenom under den här tiden. Applikationen kommer alltså inte att fördjupa sig i vad som händer efter den här introduktionstiden med varken programmet eller den anställda. En kortare diskussion om en framtida utveckling av applikationen tas emellertid upp i kapitel 6.3.

Som nämndes under "Syfte och mål" så kommer fokus i det här examensarbetet att ligga på gränssnitt, interaktion och användarvänlighet hos applikationen. Det kommer att innebära att andra delar hos applikationen inte kommer att ges samma tid och kraft till utveckling. Delar som säkerhet, optimering av funktioner i programmet och andra viktiga egenskaper som bör gälla hos en färdig applikation lämnas till framtida studier.

2 Webb- och applikationsdesign

2.1 Vad är en webbapplikation?

Om man liknar en webbsida med dagens TV-apparater så skulle en applikation kunna vara dagens dator. TV:n sätter man på för att bli tilldelad information men den bryr sig inte om vem som tittar och det går heller inte att interagera med den i någon större utsträckning. Datorn däremot vet om vem du är och det är du själv bestämmer vad som ska hända när du använder den.

En webbapplikation är som det låter, en förening av Internet (webben) och ett program på datorn (applikation). För inte alltför många år sedan så utnyttjades Internet oftast till att söka efter information, t.ex. genom att surfa runt på olika statiska sidor. Utmaningen för webbdesignern låg då framförallt i att strukturera upp informationen med länkar och göra den sökbar och enkel att få fram.

En applikation däremot är ett program på en lokal dator, framtagen för att utföra en specifik uppgift. Exempel på enkla och tidiga applikationer är Kalkylatorn, Paint och WordPad. Kombinera webbsidan med applikationen och du får, enkelt beskrivet, dagens webbapplikationer. Möjligheten att ansluta till webbapplikationen var du än befinner dig samtidigt som du kan interagera och arbeta med den på många olika sätt, allt genom din egen webbläsare.

Enligt Baxley (2003) så har en webbapplikation följande två egenskaper:

1:1- kommunikation

Webbapplikationer etablerar en unik koppling mot var och en av sina användare. En webbsida presenterar bara innehållet till vem som helst och känner inte av vem som läser eller tar del av innehållet.

Permanent ändring av data

Webbapplikationer tillåter användaren att skapa, manipulera och lagra data permanent som t.ex. transaktioner, e-mail eller annan personlig information. Sidor som Google låter användaren skicka data men lagrar inget av det för specifika användare eller till en senare session.

En annan viktig punkt som Baxley berör har att göra med designen av webbapplikationer. Till skillnad från vanliga webbsidor så kommer användaren till en webbapplikation med ett specifikt mål och med särskilda förväntningar. Man vill utföra en uppgift och förväntar sig att det ska gå snabbt och smidigt att genomföra. En viktig egenskap applikationen då bör ha är att få användaren att "glömma" designen och bara koncentrera sig på vad det är som ska utföras och att få arbetet att flyta på så friktionsfritt som möjligt.

Till skillnad från video- och datorspel, interaktiva webbsidor och liknande system för underhållning så är en webbapplikation som bäst när den inte drar uppmärksamhet till sig. Arbetet som ska genomföras i applikationen ska vara så självklart och naturligt att själva applikationen inte syns och designen är paradoxalt nog som bäst när man inte märker den. Det här betyder inte att designen inte ska vara estetiskt tilltalande eller behaglig utan att den ska ligga som stödhyll för applikationens faktiska uppgift och mål.

2.2 Design av webbapplikationer

Internet är en obestridlig mästare när det gäller att sprida information och i takt med att antalet användare växer så gör även kraven detsamma. Med så många sidor som det finns idag så ökar förstås konkurrensen och för att få en så hög trafik till sidan som möjligt så gäller det att erbjuda en positiv upplevelse så att användarna återvänder. Förutom innehållet på sidorna så är designen en hög konkurrensfaktor då det gäller att fånga användare.

För webbapplikationer gäller detsamma, men då det idag inte finns några specifika riktlinjer för hur webbapplikationer och dess gränssnitt ska se ut gäller det att ta från det som redan finns. Med tanke på webbapplikationens liknelse med traditionella program och webbsidor så är det väl inte långsökta att man även designar dem efter dessa redan utarbetade regler och normer.

Samtidigt som applikationen ska erbjuda ett väl fungerande och interaktivt sätt att kommunicera med användaren så ska den även presentera informationen på samma sätt som en webbsida gör. Navigeringen och upplevelsen hos en applikation bör därför överensstämma med en webbsidas. En viktig anledning till detta har att göra med användaren och webbläsaren som används till att läsa applikationen. För när man använder

en webbläsare så förväntar man sig också att det ska fungera på ett visst sätt. Därför är det viktigt att applikationen fungerar på samma sätt som en webbsida, för att användaren lätt ska kunna arbeta i den utan att behöva lära sig nya sätt att interagera.

2.3 Designdirektiv för webbapplikationer

Med tanke på den ofantliga mängd information och riktlinjer som finns för webb- och applikationsdesign så har det varit viktigt att sälla materialet och inrikta sig mot det som känts allra viktigast. En grundpelare inom gränssnittsdesign är fortfarande Ben Shneidermans åtta gyllene regler (1987) som, applicerade på en webbapplikation, kan beskrivas som följande:

Sträva efter konsekvens

Det är viktigt att hålla designen konsekvent genom hela applikationen, dels för att få användarna att "glömma" designen och koncentrera sig på uppgiften, men också för att få en känsla av att man hela tiden befinner sig inom applikationens ramar och funktioner.

Tillåt genvägar för återkommande användare

När användandet ökar så ökar även kraven på applikationen. Framför allt så vill man få ut så mycket som möjligt av applikationen med så lite interaktion som möjligt. Förkortningar, snabbkommandon och genvägar blir då mer attraktiva och efterfrågade.

Ge informativ feedback

För varje handling användaren utför ska applikationen ge en relevant respons. En liten handling kan ge en relativt liten respons medan en större handling bör ge mer omfattande feedback.

Designa för visuell avslutning

Texter, formulär och liknande handlingar bör grupperas så att användaren kan få en känsla av avslutning. Användaren känner belåtenhet när den har klarat av något, även om det bara är en liten del i ett större sammanhang.

Erbjud enkel felhantering

Det ska vara svårt för användaren att begå ett fel och om användaren gör fel så ska systemet på ett enkelt sätt erbjuda olika sätt att korrigera felet. Inget fel får lämna användaren frågande och utan möjlighet att gå vidare.

Ge möjlighet att ångra

Det här gör att användaren slipper oroa sig för att göra "fel", vilket i sin tur uppmuntrar till ytterligare utforskning av applikationen och dess funktioner. Även här bör man kunna ångra en liten handling eller hela moment.

Användaren som styr

Vana användare vill ha en känsla av att det är de som styr för applikationen framåt och att den svarar på deras handlingar. Motsatsen kan innebära att man känner sig styrd av systemet och inte har kontroll över vad som händer.

Reducera användandet av korttidsminnet

Människans korttidsminne kräver att applikationer hålls enkla och kortfattade. Man tar som användare hellre flera korta instruktioner än en enda lång. För applikationen innebär detta fler sidor med begränsad information, enkla displayer och liten rörlighet på bilder och fönster.

2.4 Gestaltprinciper

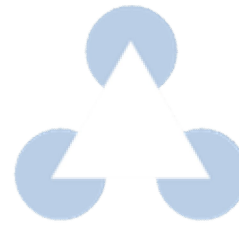
Gestaltprinciper är teorier som uppkom inom filosofin och psykologin i början av 1900-talet. Dessa teorier menar att en av hjärnans grundfunktioner är att isolera, avgränsa och strukturera information; eller enklare beskrivet, att helheten är större än summan av sina delar.

De flesta av principerna kan kännas väldigt lika och t.o.m. se ut att vara variationer av varandra men det beror på att alla behandlar samband i olika former. Människans uppfattning baseras på samband och relationer, likheter och olikheter osv.

En fundamental princip inom Gestalt perception är Prägnanz lag (2008) som i sig består av sex grundbegrepp.

Lagen om avslutning

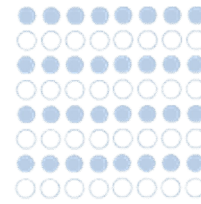
Det här begreppet tillämpas då vi tenderar i att se hela figurer även om det saknas information. Bilden till höger (Figur 3) är ett klassiskt exempel på det här. Vi ser en vit triangel på tre blåa cirkelar trots att det lika gärna kan vara tre ofullständiga cirkelar som ligger bredvid varandra. Men hjärnan fyller i den saknade informationen och triangeln är ett faktum.



Figur 3

Lagen om gemenskap

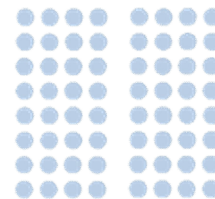
Hjärnan grupperar liknande element och figurer tillsammans. Grupperingen kan bero på olika egenskaper som färg, form, storlek eller kontrast. I figuren till höger (Figur 4) ser man rader av blå och vita cirkelar eftersom de liknar varandra. Hade alla cirkelar varit i samma färg hade man inte sett det som att cirkelarna låg uppordade på samma sätt.



Figur 4

Lagen om närhet

Objekt som ligger nära varandra kan hjärnan tolka som att de hör ihop eller är mer relaterade till varandra. I figur 5 till höger så ser vi två olika grupperingar vilka kan tolkas som att de skiljer varandra åt.



Figur 5

Lagen om symmetri

När vi ser två mer eller mindre likformade objekt nära varandra så föreställer vi oss omedvetet att de hör ihop och bildar något större. I bilden till höger (figur 6) ser vi hellre tre par klamrar som hör ihop än sex stycken olika föremål eller två par och två singlar.



Figur 6

Lagen om kontinuitet

Hjärnan fortsätter mönster och samband där det nödvändigtvis inte behöver finnas något. Hjärnan ser hellre två linjer som korsar varandra än fyra linjer som möts vid mitten i figuren till höger (figur 7).



Figur 7

Lagen om följsamhet

Syftar på att linjer eller mönster som ser ut att röra sig åt samma håll skapar följsamhet i en bild. Tillsammans har de ett gemensamt mål eller en gemensam riktning. I bilden till höger (Figur 8) grupperar man efter vilket håll linjerna lutar.



Figur 8

2.5 Applikationsdesign

Förutom de generella normer och principer ovan, som egentligen gäller för all sorts design, så har även applikationer och program egna specifika egenskaper som man kan dra lärdom av. Två av de mest påtagliga är begreppen direktmanipulation och snabbkommandon.

2.5.1 Direkt manipulation

Direkt manipulation kan enklast beskrivas som en förening av begreppen semantik och syntax. En händelse, t.ex. en förändring i applikationen ska förmedlas grafiskt till användaren (syntax) så att denne förstår vad som har skett (semantik).

Ett enkelt exempel på direkt manipulation är när man tar bort en fil på datorn. Att flytta runt en fil på skärmen till olika mappar innebär även att man flyttar runt filen i datorns filsystem. Flyttar man filen till papperskorgen så försvinner filen och papperskorgen ändrar utseende och visar att filen har slängts.

I sin bok om interaktionsdesign behandlar Cooper (2007) det här ämnet och han listar tre egenskaper som sammanfattar direkt manipulation på ett bra sätt.

- Visuell representation av det manipulerade objektet
- Fysisk handling istället för textbaserade kommandon
- Omedelbar visuell respons vid en utförd handling

En enklare beskrivning av direkt manipulation, enligt Cooper, skulle kunna vara att "klicka och dra omkring saker". Även om det uttrycket bara

behandlar en av punkterna ovan så ger det en god bild av vad begreppet innebär.

2.5.2 Snabbkommandon

Något som alla program utnyttjar flitigt idag är snabbkommandon och genvägar för avancerade användare. Kombinationen av att använda musen och tangentbordet är oslagbar när det gäller att höja effektiviteten och ger användaren ett flyt i sitt arbete som är omöjlig att uppnå med bara en mus.

Snabbkommandon bör vara ett komplement till redan existerande alternativ och ska ligga osynliga för ovana användare. Nybörjare brukar oftast föredra att använda menysystem och enklare metoder medan vana användare håller tangentbordets kommandon som extra hjälpmedel.

Webbapplikationer använder idag inte snabbkommandon i samma utsträckning som vanliga program, även om några har spridit sig och blivit allt mer använda. En av de vanligaste situationerna är när man använder Enterknappen till att söka eller skicka vidare ifylld information på en webbsida eller i ett webbformulär. En annan välanvänd knapp är Tab-knappen som ofta används till att flytta fokus mellan olika delar eller fält på en webbsida.

Det finns såklart nackdelar med kortkommandon hos webbapplikationer som inte påverkar vanliga program. En anledning till att snabbkommandon inte är lika utbrett på webben beror på att många kommandon redan är upptagna av webbläsaren som presenterar applikationen. Kombinationer innehållande Ctrl-, Skift- och Alt-knappar är redan upptagna och gör att valen av kommandon till webbapplikationen blir färre och mindre intuitiva.

2.6 Webbdesign

Om applikationsdesign kan lära oss om direkt feedback och användarvänlighet så kan webbdesign ge viktig information om bl.a. navigation, grafisk layout och strukturering av information.

Nielsen (2000) och många andra webbdesignguruer har kommit fram till att en besökare inte läser ordagrant vad

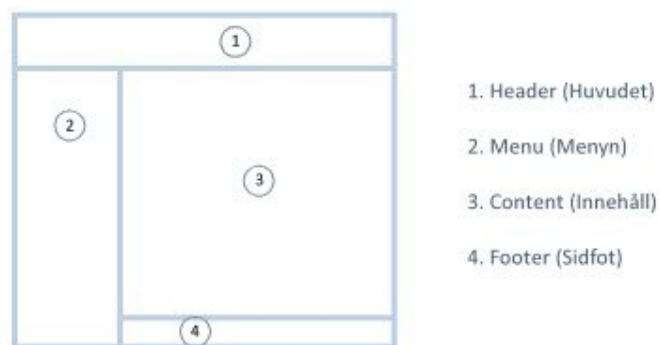
"People rarely read Web pages word by word; instead, they scan the page, picking out individual words and sentences."

- Jacob Nielsen

som står på en webbsida, utan att de skannar av den. De flesta användare kommer till sidan för att leta efter specifik information och skannar då av sidan snabbt efter nyckelord eller relevanta fraser. Hittar de inget så fortsätter de vidare till någon annan sida.

2.6.1 Layout

Något som är avgörande för hur en webbsida kommer att uppfattas vid ett första besök är dess layout. En intressant layout kommer att fånga användarens uppmärksamhet och väcka intresse hos användaren till att fortsätta undersöka sidan och dess innehåll. En klassisk webbsida med god layout innehar fyra grundobjekt som hjälper att strukturera upp sidan, vilket visas i bilden nedan (figur 10).



Figur 9 Grundstrukturen för en webbsida.

I huvudet (1) presenteras företaget eller organisationen, oftast i form av en logga eller en annan bild. Navigeringsfältet (2) ligger vanligast i en meny till vänster men kan också förekomma i huvudet på webbsidan beroende på storleken på hela portalen och mängden information som ska presenteras. En tung portal med många undersidor tjänar på att ha menyn till vänster medan en lättare webbsida kan spara in plats genom att ha en meny i huvudet på sidan. I sidfoten presenterar man allmän information om sidan så som kontaktinfo, företagsinfo, copyright och annan eventuell sekretess.

2.6.2 Navigation

När strukturen är klar så gäller det att på ett enkelt och intuitivt sett kunna navigera mellan all sektioner på sidan. Har man lyckats med en bra layout så kommer navigationen bli just enkel och intuitiv. Användarexperten Steve

Krug (2000) listar tre grundläggande typer av navigation som även om de inte är standardiserade över Internet trots allt är väldigt vanliga.

Global navigation

Navigationselement som återfinns på varje sida genom hela strukturen, t.ex. "Hem" länken som ofta finns längst upp på en sida.

Sekundär navigation

Länkar som gäller specifika sidor klassas som sekundära länkar och utgör ofta undermenyer inom olika teman eller sektioner.

Allmän navigation

Den här kategorin är en slags uppsamlingsplats för länkar som inte passar in hos de andra två. Kontaktsidor, sidor som beskriver företaget och FAQ-sidor (Frequently Asked Questions) är exempel på dessa.

3 Teknisk analys

3.1 Microsoft SharePoint

Microsoft SharePoint, även känd som "Microsoft SharePoint products and technologies" är en samling produkter och program som bl.a. inkluderar process- och dokumenthantering, sökmoduler och diverse samarbetsfunktioner genom webbläsaren. Termen "SharePoint" brukar innefatta allt från plattformen som SharePoint bygger på till alla dess olika funktioner (Pyles, James m.fl. 2007). Microsoft nämner följande produkter som delar i begreppet SharePoint:

- [Microsoft Office SharePoint Server 2007 \(MOSS\)](#)
- [Windows SharePoint Services 3.0 \(WSS\)](#)
- [Search Server 2008 Express](#)
- [Forms Server 2007](#)
- [Microsoft Office SharePoint Designer](#)

Tekniken bakom SharePoint bygger på ASP.NET 2.0 som är ett programmeringsspråk speciellt framtaget för att bygga webbsidor, webbapplikationer och liknande webbtjänster. SharePoint använder sig också av IIS för publiceringen (Microsoft Corporation, 2009) av applikationerna (webbserver) och SQL Server för lagring av data. Eftersom det här examensarbetet bara tillfälligt kom i kontakt med SharePoint så hänvisas intresserade läsare, som vill lära sig mer om det här systemet, till Microsofts hemsida.

Efter en tids arbete med SharePoint och med relativt lite framgång så kändes möjligheten att bygga en hel prototyp i det här systemet avlägset. Anledningen är att SharePoint redan består av egna regler och ramverk som, när man lärt sig dem, antagligen underlättar programmeringen avsevärt. Man får tillgång till redan färdiga moduler och funktioner, vilket säkerligen snabbar upp implementeringen, dvs. om man redan har kunskapen om dessa regler och ramar. Med hänsyn till tidsramen utsatt för det här examensarbetet så blev bedömningen att det skulle ta allt för lång tid att först lära känna SharePoint och sedan utveckla en applikation i det.

Lösningen blev istället att utveckla en fristående applikation, fortfarande med samma teknik men utan SharePoints regelverk och komponenter.

3.2 Fristående applikation

Grundtekniken bakom den prototyp som utvecklades är alltså i princip densamma som den hade varit om SharePoint hade använts. System och mjukvara som användes i det här projektet var sådant som redan fanns tillgänglig hos H2 Data och som de använde sig av. Dessutom, eftersom önskan fanns om att implementera applikationen i SharePoint, så användes mestadels Microsofts egen anpassad teknologi där det var möjligt.

Nedan följer en kort beskrivning av tekniken som användes och vad dess olika delar gör. För mer information om varje del så rekommenderas läsaren att följa länkarna i respektive referens.

IIS

Internet Information Services (IIS) installeras på en dator för att kunna svara på http-anrop på port 80, vilket kort sagt betyder att man installerar en webbserver. Eftersom både SharePoint och IIS är Microsoft-produkter och eftersom H2 Data redan använde sig av den här typen av server så kändes det naturligt att fortsätta med detta (Microsoft Corporation, 2009).

SQL Server

SQL står för Structured Query Language och är ett standardiserat språk för att hantera och modifiera data i en relationsdatabas. Man kan från webbservern säga till SQL-databasen att man vill ha en viss samling data för att bearbeta och visa upp på sidan (Microsoft Corporation, 2009).

HTML och ASP.NET

HyperText Markup Language (HTML) och Active Server Pages .NET (ASP.NET) är ramverk i vilket man bygger webbapplikationer och webbsidor. Med den här koden beskriver man hur sidan är uppbyggd och var text, länkar och bilder ska vara placerade m.m. (Microsoft Corporation, ASP.NET, 2009).

CSS

Cascading Style Sheets (CSS) är en textfil som talar om hur HTML-dokumentet ska se ut. Fördelen med att använda CSS är att man med

bara en fil kan bestämma hur alla webbsidor inom applikationen ska definieras (W3C, 2009).

C#

C-Sharp som det uttalas är ett objektorienterat språk baserat till stor del på C++ men har även en hel del gemensamt med Java. C# är avsett att vara ett enkelt, modernt och objektorienterat programmeringsspråk. Med C# så sköter man logiken när en applikation eller webbsida ska byggas upp tillsammans med data från databasen (Wikipedia, C Sharp, 2008).

JavaScript

JavaScript är en ett scriptspråk som har likheter med programmeringsspråket Java. Man kan skriva det direkt i HTML-koden och man behöver heller inte kompilera det innan det körs. Tillskillnad från C# så körs JavaScript hos klienten och inte på servern (Sun Microsystems, 2009).

4 Metodbeskrivning

4.1 Prototyputveckling

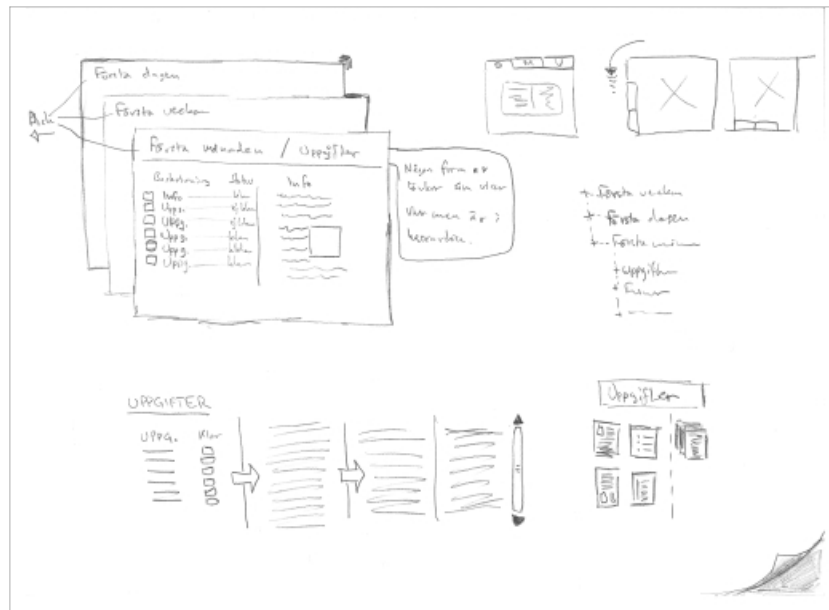
Inom mjukvaruutveckling har det, sedan 1970 då det introducerades, blivit allt vanligare att bygga prototyper eller enkla versioner av den mjukvara som senare ska utvecklas (Wikipedia, 2008). Allmänt gäller för all typ av prototyputveckling att det ska underlätta kommunikationen, både utvecklaren och användaren emellan men också mellan två utvecklare.

Pappersprototyper är en av de mest använda metoderna inom användarcentrerad designutveckling när det gäller det inledande stadiet av en utvecklingsprocess. Processen hjälper utvecklarna att möta användarens förväntningar och behov, framförallt när det kommer till designen och relationen mellan användaren och programvaran.

Alla olika delar av applikationen ritas ned på pappersbitar som sedan används som pusselbitar i en större modell. Pappersbitarna föreställande delar av designen kan sedan flyttas runt, målas om, eller bytas ut helt mot andra delar vilket ger utvecklaren nästan obegränsade möjligheter att testa idéer och designförslag. Alla runtomkring kan dela med sig av sina egna förslag och påverka andras bara genom att flytta runt eller lägga till sina egna delar med händerna.

4.1.1 Fördelar och Nackdelar

Fördelen med pappersprototyper är att de framförallt sparar värdefull tid och pengar eftersom man lätt kan testa många av sina idéer och funktioner med sitt program långt innan någon kod har skrivits. Det är självklart mycket lättare att modifiera både designen och strukturen i applikationen om man har den framför sig i pappersform och man gör detta helt utan att stöta på några buggar eller felmeddelanden. Dessutom minskar man kostnaderna och svårigheterna med att genomföra dessa ändringar sent i utvecklingsfasen vilket i sin tur underlättar supporten av programvaran.



Figur 10 Ett exempel på en av pappersprototyperna som användes under studien.

En av fördelarna med den här typen av modellering är att man skapar och influerar kommunikationen inom ett utvecklingsteam, något som naturligtvis försvinner helt när man bara är en ensam utvecklare.

Vidare så finns det begrepp som blir svårare att beskriva med pappersprototyper gentemot en virtuell applikation. Begrepp som t.ex. hur lång tid det tar att ladda en sida, vilka färger och nyanser som ska finnas på sidan och hur dessa passar ihop med varandra. Generellt gäller detta allt som har med bilder, färger, fonter eller specifik design att göra. Att få en helhetssyn eller en känsla för hur programmet verkligen kommer att se ut är också svårt med den här typen av arbetssätt. Scrollning är en annan detalj som är enkel att visa i ett program men som blir väldigt opraktisk i pappersform.

Trots dessa brister så är fördelarna med pappersprototyper så många och övervägande att det har blivit en självklar del i den inledande utvecklingsfasen av en applikation. Att få ner sin design på papper och kunna omvärdera den medan man tittar på den är utvecklande och gör att man analyserar sina egna idéer på ett mer konstruktivt sätt. Dessutom kan objekten som skapats i den här fasen även användas i senare delar av utvecklingen.

4.2 Intervjuer

Intervjuer i undersökningssyfte brukar oftast delas in i tre större grupper, strukturerade, semi-strukturerade och ostrukturerade intervjuer (Wikipedia, 2009).

Ostrukturerad

Vid en ostrukturerad intervju ställer personen som intervjuar en rad öppna frågor. Med öppna menas att den svarande har möjlighet att styra intervjun och ta upp punkter som han eller hon tycker är intressanta och viktiga. Den här typen av intervjuer används allt som oftast när intervjuaren i fråga har liten förståelse i ämnet som diskuteras och alltså inte vet vad som kan komma att vara intressant eller viktigt för undersökningen.

Semi-strukturerad

Genomför man en semi-strukturerad intervju så bör intervjuaren ha en god uppfattning om ämnet som ska diskuteras och vilka frågor som är relevanta att ta upp. Här kan man begränsa svaranden till att bara ta upp frågor som verkligen är intressanta för undersökningen och som bara rör ämnet i fråga. Det är också lättare att få en mer systematisk analys av problemet jämfört med föregående intervjumetod.

Strukturerad

Den tredje och sista varianten är en helt strukturerad intervju där, till skillnad från de två tidigare intervjuerna, man har här möjligheten att guida svaranden att hålla sig inom ett visst område. Man kan t.ex. ge ett antal förutbestämda svar eller hålla sig till formulär eller liknande. Den här sortens intervju faller mer under kvantitativ analys än de två andra eftersom svaren i det här fallet blir väldigt lika varandra och därmed lättare att analysera.

4.2.1 Fördelar

Intervjuer är mångsidiga på det sätt att de kan användas genom hela utvecklingsfasen och fortfarande inbringa viktig information. Beroende på hur man lägger upp intervjun så kan man undersöka personers attityd och intresse för prototypen lika väl som mot den slutliga produkten.

En av många fördelar med en intervju jämfört med ett statistiskt frågeformulär är hur lätt en intervju kan förändras på plats och vilken fördel det här innebär för insamlandet av information. Om t.ex. en person

som svarar på ett formulär missförstår en fråga, eller om personen ger ett svar med irrelevant information, så kan man ju under en intervju modifiera frågan lite så att man får ett mer meningsfullt svar. Under en intervju kan man även ställa följdfrågor till ett problem som har uppkommit på plats och på så sätt utvinna mer angelägen information, något som ett frågeformulär inte klarar av.

Ett annat exempel gäller hur intresserad personen i fråga är av ämnet som ska undersökas eller hur stor motivationen är för tillfället. Det händer väl sällan att en intervju avbryts på grund av motivationsbrist eller liknande. Har någon väl accepterat att ställa upp på en intervju så följer man allt som oftast igenom med den. Däremot så är det mer vanligt att en person som har börjat fylla i ett frågeformulär tröttnar någonstans i mitten och antingen fyller i meningslös information bara för att bli klar eller helt enkelt bara lämnar den halvfärdigt ifylld.

4.2.2 Nackdelar

Kostnaderna att genomföra en intervju kan bli höga jämfört med att samla in information från formulär eftersom intervjuaren måste vara med på plats, något som normalt inte gäller för de andra undersökningsmetoderna. Kostnaden i både tid och pengar blir högre ju fler personer som ska undersökas.

Ett annat problem med intervjuer är att personen som genomför undersökningen lätt kan tolka eller förvränga svaren på ett eller annat sätt. Detta kan ske oavsiktligt och utan att intervjuaren tänker på det, något som Jordan (2000) nämner i sin bok. Han menar också att när man uttrycker sina åsikter till en annan person så finns det en risk att man inte vågar eller vill delge allt. Man kanske känner sig obekvämt med att uttrycka sig ansikte mot ansikte och föredrar att ge sina svar skriftligt. Detta tror Jordan kan bero på att när två människor talar med varandra så är det naturligt att man oftast vill framstå som en resonlig person och inte komma med obekvämligheter.

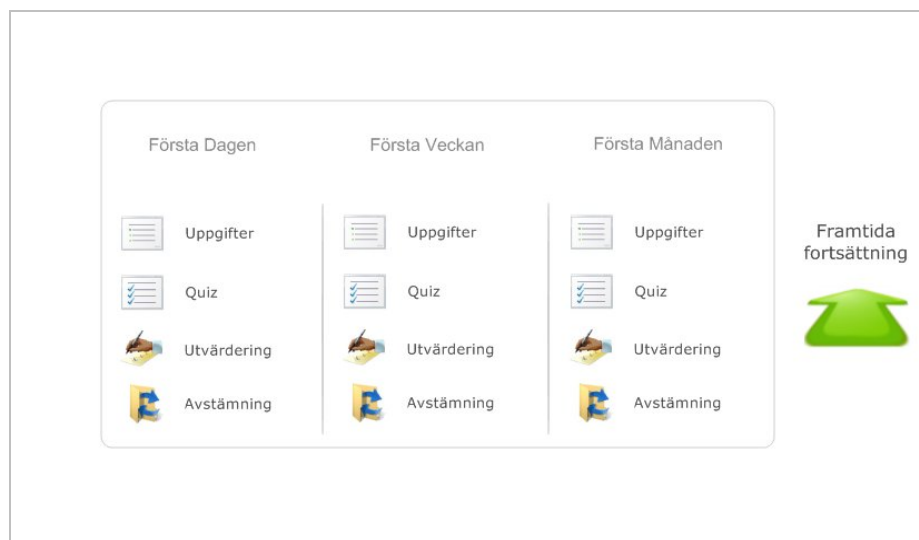
4.2.3 Medierande objekt

Att mediera betyder att förmedla eller "stå i mitten", enligt Svenska Akademiens Ordlista (2006). Ett medierande objekt är något som är tänkt att stimulera reflektion och underlätta dialogen mellan två parter, i det här fallet under en intervju. Enligt Karlsson (2005) så kan ett medierande

verktyg användas i en användarstudie på två sätt, med syfte att identifiera krav eller i syfte att utvärdera designlösningar. Hon säger däremot att huvuddelen av existerande forskning kring valet av medierande objekt har rört produktrepresentationer för utvärdering av designlösningar, vilket ju gäller i det här fallet.

Ju mer detaljerad och omfattande ett medierande objekt är desto mer bör det ju influera dialogen mellan deltagarna. Vidare, så om man har ett detaljerat medierande objekt som kan tänkas likna den slutgiltiga produkten, så bör det ge mer utmärkande input till hur designlösningar och krav kan specificeras än om man bara har en skiss.

Nedan följer en av prototyperna som användes som medierande objekt under intervjun (Figur 9). Bilden är från det andra och sista steget i prototyputvecklingsfasen där skisser utvecklats till detaljerade bilder som praktiskt taget liknar en fullt fungerande applikation. Upplägget och grafiken i den här bilden användes nästan rakt av i den färdiga prototypen.



Figur 11 En av prototyperna som arbetades fram under utvecklingsfasen.

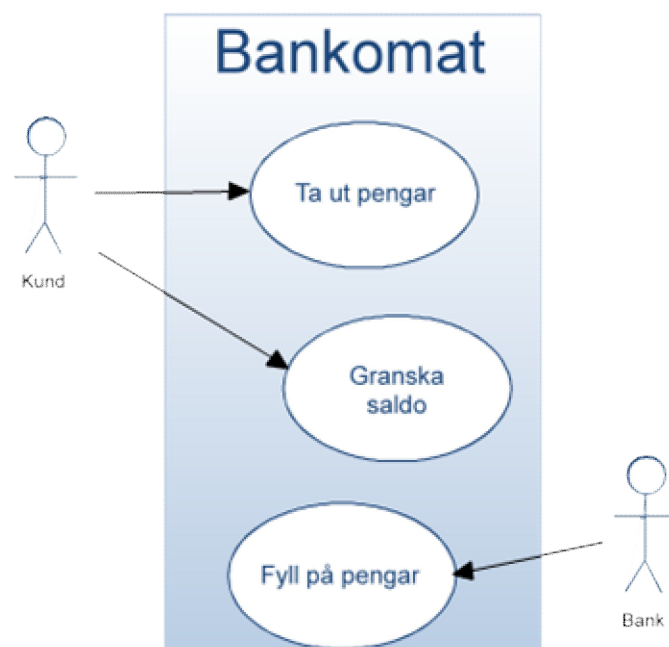
4.3 Användningsfall

Ett användningsfall (Eng. Use Case Diagram) är en grafisk beskrivning av hur ett system ser ut och beter sig när någon utomstående kraft interagerar med det (Jia, 2002). Det är alltså en beskrivning av VAD ett system gör och inte HUR det gör det. Ett användningsfall innehåller *användare* som

interagerar med systemet, *mål* som beskriver vad som ska göras och eventuella *beroenden* som kan finnas mellan de två.

Bankomater brukar tas som ett vanligt exempel när man ska beskriva användningsfall. Där är det kunden och banken som interagerar med systemet och betraktas som *användare*. Ett *mål* med systemet kan vara att ta ut pengar eller att granska sitt saldo.

En viktig detalj med bankomater är att det inte går att ta ut några pengar om det inte finns några i bankomaten, detta kan då skrivas som ett *beroende* i användningsfallet.



Figur 12 Ett enkelt exempel på ett användningsfall för en bankomat.

En väsentlig fördel med användningsfall är att de tillåter utvecklaren ta ett steg tillbaka, analysera systemet och verkligen ställa frågor som "vad är det som händer" och "vad är det vi egentligen behöver göra". Detta utan att man behöver ta hänsyn till implementeringen eller liknande problem. Sådana här frågor leder ofta till att man kan tänka om och omforma systemet eller övergripande koncept tidigt i processen.

Användningsfallet som användes i det här arbetet presenteras i bilaga C.

4.4 Genomförande

Arbetet började med litteraturstudien av webb- och applikationsdesign som beskrivs i kapitel 2. Detta gjordes för att friska upp minnet av det som behandlats under det senaste året på linjen för interaktionsdesign, men också för att specialisera mig mer inom det aktuella området.

Vidare gjordes den tekniska analys beskriven i kapitel 3, som skulle ge klarhet i vilka verktyg och inom vilket system programmet skulle utvecklas.

När alla förutsättningar var bestämda så inleddes arbetet med den prototyputveckling som beskrivs i kapitel 4.1. Prototyputveckling gjordes först eftersom det kändes viktigt att ha med ett medierande objekt i intervjuerna som skulle följa senare.

Intervjuerna skedde sporadiskt från det att den första prototypen var avklarad till det att programmet i stort sett var färdigt. Intervjuerna, beskrivna i kapitel 4.2, hölls semi-strukturerade och gjordes med ett par anställda på H2 Data. Upplägget och frågorna kan ses i bilaga A.

Innan programmeringen påbörjades gjordes en bild av hur processen skulle se ut, beskriven i Bilaga B, och ett användningsfall, beskriven i kapitel 4.3 och visualiserad i Bilaga C.

5 Resultat

5.1 Processen

Som nyanställd gäller det att snabbt komma in i företagets rutiner, förstå företagets mål samt att kunna se kopplingen mellan det egna arbetet och hur företag går. Alla nyanställda borde få en "normlots" in i verksamheten (Norrby 2008).

Prototypen som utvecklades tar vara på det här och ser till att sätta en viss standard för vad som ska täckas i introduktionen. Processen som den nyanställde ska ta sig igenom när den kommer till ett nytt företag är uppdelad i tre moduler: Första dagen, Första veckan och Första månaden. Processen med att introducera en nyanställd är beräknad till att ta en månad och därefter bör en ny handlingsplan upprättas för framtida utveckling. En noggrannare visualisering av processen kan ses i Bilaga B och det kan nog rekommenderas att man tar en snabb koll på den innan man läser vidare. Om inte annat så för att få en översiktlig bild av processen och vilka steg den består av.

Första dagen

Den här dagen räknas som den första dagen då den nyanställde verkligen börjar arbeta. Ofta så är första dagen på en arbetsplats inte en vanlig arbetsdag utan innehåller mer sociala bitar som att gå runt på arbetsplatsen, presentera sig för nya arbetskamrater och äta lunch med chefen.

Under den första arbetsdagen bör man gå igenom den nödvändigaste informationen för att klara av sitt arbete. Här är det tänkt att information som t.ex. larmfunktioner, utrymningsplaner vid brand, fotografering och upprättande av ID-kort m.m. ska ha en klar plats.

Första veckan

Under den första veckan ska man ordna med många praktiska saker och sätta sig in i de vanligaste administrativa göromålen. Exempel på dessa kan vara att lära sig hur telefonen och telefonväxeln fungerar, hur tidrapporter fylls i, upprättande av e-mail och mobiltelefon och andra kontaktuppgifter m.m.

Första månaden

När man har klarat av den första veckan på den nya arbetsplatsen kan man blicka lite längre framåt och hantera mer övergripande uppgifter och områden.

Varje modul som beskrivits ovan (Första dagen, Första veckan, Första månaden) består i sig själva av fyra steg som den nyanställde ska ta sig igenom. De fyra stegen är Uppgifter, Quiz, Utvärdering och Avstämning.

Uppgifter

Här listar chefen eller den ansvarige för applikationen vilka uppgifter och vilken information som den nyanställde ska ta del av. Uppgifterna kryssas av allteftersom de blir avklarade.

Quiz

Efter att uppgifterna har blivit avklarade så går den nyanställde igenom ett frågeformulär. Frågorna är baserade på de gjorda uppgifterna och säkerställer att den nyanställde verkligen har tagit del av informationen som delgivits. Om man svarar rätt på mer än 80 % av frågorna går man vidare till utvärderingen, annars får man gå tillbaka och läsa igenom uppgifterna igen, innan man gör om provet.

Utvärdering

Utvärderingen är till för att ge feedback till företaget och chefen om hur introduktionen har gått och hur den nyanställde upplevde den. Här svara man på frågor om vad som var bra och dåligt och eventuella förslag till förbättring.

Avstämning

När utvärderingen är klar stämmer man av med chefen att man är redo att gå vidare i introduktionen.

Efter att alla tre moduler är avklarade och introduktionen är slut så bör man sammanställa alla svar och utvärderingar på en plats och planera en framtida fortsättning för den nyanställde. Men eftersom den här frågan ligger utanför det här arbetet så kan bara rekommendationer lämnas om hur systemet bör utvecklas vidare.

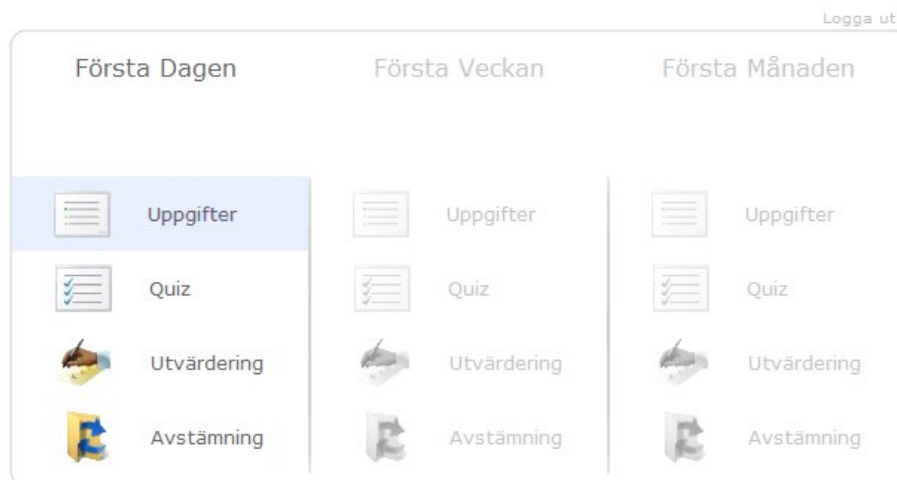
5.2 Applikationens delar

Det första den nyanställda möts av när den använder applikationen är en inloggningsruta med användarnamn och lösenord. Det är upp till chefen eller den systemansvarige att se till att den nyanställda har ett eget konto innan han börjar med introduktionen.

Anledningen till att en inloggning bör göras innan är för att säkerställa att ingen annan tar del av informationen, men också för att behålla en känsla av trygghet för den nyanställda. Det underlättar att lämna information och utvärderingar, om företaget och upplevelserna, om man vet att det stannar mellan chefen och en själv.

5.2.1 Huvudmenyn

Huvudmenyn i applikationen består av de tre modulerna Första dagen, Första veckan och Första månaden, vilka i sig har fyra undermenyer. När man använder sig av applikationen för första gången ligger fokus på första dagen och man klickar sig fram i menyerna med hjälp av musen.



Figur 13 Huvudmenyn i applikationen.

5.2.2 Uppgifter

Här listas alla frågor och all information som den nyanställda ska gå igenom under respektive period. Genom att klicka på en länk till vänster så presenteras informationen till höger. När användaren har gått igenom texten så klickas "klar"-knappen in och uppgiften bockas av som avklarad.

Första Dagens Uppgifter

Uppgift

- Om företaget **Klar**
- Historik **Klar**
- Publiceringsverktyg (CMS) **Klar**
- Kunder & lösningar **Klar**
- Mailkonto
- ID-kort

Om företaget

H2 Solutions arbetar med kunder i flera olika branscher. Därmed överför vi kompetens och erfarenheter mellan de olika branscherna vi är verksamma i. Idag arbetar vi inom kemi, läkemedel, stat och kommun. Inom kemi har vi en stark position och arbetar med såväl leverantörer och operatörer som användare.

Läs mer på: [H2 Solutions / Om företaget](#)

Klar

Figur 14 Exempel på uppgifter från första dagen.

5.2.3 Quiz

Frågeformuläret består av ett flertal frågor som matchar den information som delgetts under uppgifter, där varje fråga har tre svarsalternativ. För att klara provet behöver man svara rätt på minst 75 % av frågorna. Frågorna rättas i realtid och klarar man provet visar applikationen detta och man återvänder till huvudmenyn.

Första Dagens Quiz

För att få godkänt så behöver du ha rätt på minst 75% av frågorna.

1. Vilken gata ligger H2 Solutions på?

- ☒ Göteborgsvägen 91
- ☐ Mölndalsvägen 12
- ☐ Vasagatan 5

2. Vilket årtal startade H2 sin verksamhet

- ☒ 1990
- ☐ 1991
- ☐ 1992

3. Inom vilket område används Athena CMS

Figur 15 Första dagens quiz.

5.2.4 Utvärdering

Här fyller användaren i ett antal frågor i fritext om hur tiden har varit på arbetsplatsen. Vad var bra första månaden? Vad var dåligt första månaden? Vad har du för önskemål eller synpunkter?



Figur 16 Utvärderingen från första dagen.

5.2.5 Avstämning

När utvärderingen är slutförd meddelas chefen att användaren är klar. Då kan chefen och den nyanställda träffas och stämma av hur tiden har varit och vad som har tagits upp i utvärderingen.

5.2.5 Avklarade steg

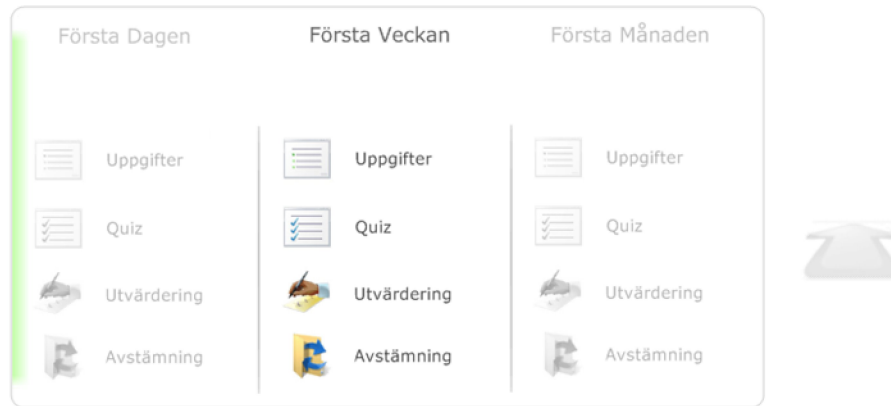
Allteftersom användaren klarar av de olika delarna så visualiseras detta i huvudmenyn med en grön markering.



Figur 17 Menyn markeras som avklarad.

5.2.6 Avklarade moduler

Efter att en modul har blivit avklarad så visualiseras detta genom den gröna markeringen i exemplet ovan (Fig. 17) men också genom att fokus flyttas från en modul till nästa, som i exemplet i Figur 18. Genom att tona ner färgerna i de andra modulerna läggs fokus på den aktuella modulen.



Figur 18 Menyn flyttar fokus mellan moduler.

5.2.6 Avklarad process

När användaren har gått igenom hela processen och behandlat och fullföljt alla delar så är introduktionen klar, vilket visualiseras genom att pilen längst till höger lyser upp (Fig. 19).



Figur 19 Introduktionen är avklarad.

6 Diskussion

6.1 Design

Designen och upplägget av applikationen har till stor del följt den webb- och applikationsdesignstudie som gjordes i kapitel 2.

När beslutet togs om att inte utveckla applikationen i SharePoint så behölls ändå tanken att den en gång i framtiden kan komma att implementeras där. Därför har designen hållits relativt avskalad men ändå stilren samtidigt som Microsofts design har varit den största inspirationskällan. Dessutom har ikoner och designidéer från andra Microsoft-program använts som inspiration till utseendet hos applikationen. Färgerna vitt och blått har legat som grund genom hela applikationsdesignen vilket också stämmer överrens med Microsofts teman och utseende.

6.1.1 SharePoint

Valet av att inte utveckla prototypen i SharePoint har väckt en del intressanta frågor angående applikationens framtid. En av fördelarna med att placera applikationen utanför ett större system är ju såklart att den blir helt oberoende. SharePoint kostar en viss summa pengar att införskaffa och företag och organisationer som inte har det här systemet kommer ju därför heller inte att kunna använda applikationen. Om applikationen utvecklas som ett eget system, utanför SharePoints regler och ramverk, så kommer den att kunna rikta sig till fler företag och organisationer. Något som innebär större målgrupp och därmed en potentiellt större marknad.

Vad som talar för SharePoint är däremot att det redan finns en struktur och en bred grund att utveckla applikationen på. Användarhantering, dokumenthantering och processer finns redan som färdiga moduler vilket underlättar vid utvecklingen av applikationen. Funktioner som redan finns färdiga behöver man därför inte lägga tid och pengar på att utveckla separat till just den här applikationen.

Vilket alternativ som är att föredra beror på fler faktorer än vad den här rapporten och det här arbetet har gått igenom. Därför lämnas ett definitivt svar på den här frågan till framtida studier.

6.2 Framtida arbete

För att få en så komplett applikation som möjligt så bör applikationen innehålla fler funktioner, som innefattar fler användare och större bredd. Ett par av de här rekommenderade funktionerna ges nedan .

6.2.1 Administrationsgränssnitt

En annan funktion som praktiskt taget måste finnas är ett gränssnitt för chefen eller den som ansvarar för innehållet i applikationen. Ett enkelt sätt att byta ut, uppdatera och ta bort frågor eller uppgifter som inte behövs bör vara en självklarhet i en sådan här applikation.

Dessutom behövs ett sådant här gränssnitt när t.ex. en ny introduktion för en nyanställd ska skapas, vilket visas som exempel i bilden nedan.

Ny introduktion

1. Välj användare till introduktionen:
2. Välj uppgifter till introduktionen:

Första dagen	Första veckan	Första månaden
☒ ID-kort	☒ Annan uppgift	☒ Annan uppgift
☒ Arbetsplats	☒ Annan uppgift	☒ Annan uppgift
☒ Om företaget	☒ Annan uppgift	☒ Annan uppgift
☐ Historia	☒ Annan uppgift	☒ Annan uppgift
☐ Annan uppgift	☒ Annan uppgift	☐ Annan uppgift
☐ Annan uppgift	☐ Annan uppgift	☐ Annan uppgift
☐ Annan uppgift	☐ Annan uppgift	☐ Annan uppgift

markera alla | avmarkera alla

Figur 20 Exempel på hur en ny introduktion kan skapas.

6.2.2 Sammanställning

Applikationen som den är utvecklad så här långt tar hand om den nyanställda genom dennes första månad på den nya arbetsplatsen. Men för att verkligen få ut så mycket som möjligt ur applikationen så bör även en sammanställning finnas över den här månaden. Den här sammanställningen kan innehålla statistik över användarens med- och motgångar med applikationen.

Genom att skapa en sida med statistik från en och flera användare så kan man få en översikt över applikationen och alla dess delar, vilket i sin tur kan

hjälpa att utvärdera applikationen. Vilka delar har varit svårast? Vilka frågor och uppgifter kan man stryka eller förbättra eftersom många användare har missuppfattat eller svarat fel på dem? Vilken del har tagit längst tid att genomföra? Varför har vissa användare slutat att använda applikationen?

Dessa och många andra tänkbara frågor kan man få svar på genom att utveckla en sida med data och statistik från alla användare som sedan kan analyseras av en ansvarig.

6.3 Egna tankar

Den absolut största fördelen med en datorbaserad introduktion, som jag kan se, är att man får all information samlad på ett och samma ställe. Dessutom ökar man möjligheterna att ge sina nyanställda en mer enhetlig och jämn introduktion där alla anställda kan ta del av exakt samma information.

Ett interaktivt stöd kommer nog mer till sin rätt om den används till att introducera en ny programmerare än t.ex. en ny hantverkare. Allt handlar om att det interaktiva stödet smälter in arbetarens vardag och inte blir till ett extra moment som besvärar mer än det hjälper. Jag har även sett att ett program som detta kommer bäst till användning ju större företaget är, som använder sig av det.

Uppgiften i sig utvecklades till att bli riktigt intressant i slutändan. Till en början var jag rädd att arbetet skulle bli mer av en programmeringsuppgift snarare än att handla om design och utveckling. Men jag känner själv att jag lyckades styra allt i rätt riktning och hålla det inom ramarna för interaktionsdesign.

Referenser

Baxley, Bob, *Making the Web Work*, New Riders Publishing, 2002.

Cooper, Alan, Reimann, Robert, Cronin, David, *About Face 3: The Essentials of Interaction Design*, John Wiley & Sons, 2007.

Jia, Xiaoping, *Object-oriented software development using Java: principles, patterns, and frameworks*, Addison – Wesley, 2002.

Jordan, Patrick W., *Designing pleasurable products*, Taylor & Francis Group, 2000.

Karlsson, Mari Anne, *Medierande verktyg och deras roll i användarstudier*, <http://www.cs.chalmers.se/idc/ituniv/kurser/06/analys/medierande%20verktyg.pdf>, [Hämtad den 22 juli 2008].

Klöfver, Stefan, *Medarbetarintroduktion*, opublicerad, 2003.

Klöfver, Stefan, *Metodpärm - medarbetarintroduktion*, opublicerad, 2003.

Krug, Steve, *Don't Make Me Think!: A Common Sense Approach To Web Usability*, New Riders, 2000.

Microsoft Corporation, *Microsoft Office SharePoint Server*, <http://www.microsoft.com/Sharepoint/default.aspx>, [Hämtad den 3 mars 2008].

Microsoft Corporation, *IIS – Internet Information Services*, <http://www.iis.net>, [Hämtad den 12 oktober 2009].

Microsoft Corporation, *ASP.NET*, <http://www.asp.net>, [Hämtad den 14 oktober 2009].

Nielsen, Jakob, *Designing Web Usability*, Peachpit Press, 2000.

Norrby, Ann, *Välkommen till nya jobbet*, Ny teknik, Nr 6, pp.24, 6 februari 2008.

Pyles, James, Buechler, Christopher M., Fox, Bob, Gordon, Murray, Lotter, Michael, Medero, Jason, Mehta, Nilesh, Poelmans, Joris, Pragash, Christopher, Prussak, Piotr & Regan, Christopher J., *SharePoint 2007: The Definitive Guide*, O'Reilly Media, Inc., CA, First Edition, 2007.

Webb, Jeff, *SharePoint 2007: Essential*, O'Reilly Media, Inc., CA, Second Edition, 2007.

Shneiderman, Ben, *Designing the user interface*, Addison – Wesley, 1987.

Sun Microsystems, Inc., *JavaScript Technology*, <http://developers.sun.com/scripting/javascript>, [Hämtad den 14 oktober, 2009].

Svenska Akademiens ordlista över svenska språket, *Mediera*, Svenska Akademien och Norstedts, 2006.

Wikipedia, *Gestalt psychology*, http://en.wikipedia.org/wiki/Gestalt_psychology, [Hämtad den 20 juli 2008].

Wikipedia, *C Sharp*, [http://en.wikipedia.org/wiki/C_Sharp_\(programming_language\)](http://en.wikipedia.org/wiki/C_Sharp_(programming_language)), [Hämtad den 30 juli 2008].

Wikipedia, *Interview*, <http://en.wikipedia.org/wiki/Interview>, [Hämtad den 12 oktober 2009].

W3C (World Wide Web Consortium), *Cascading Style Sheets (CSS)*, <http://www.w3.org/Style/CSS>, [Hämtad den 14 oktober 2009].

Övrig litteraturförteckning

Berkeley, University of California, *Guide to Managing Human Resources, Chapter 5: New Employee Orientation*, <http://hrweb.berkeley.edu/guide/orient.htm>, [Hämtad den 14 juni 2008].

Baxley, Bob, *What is a Web Application?* http://www.boxesandarrows.com/view/what_is_a_web_application_, [Hämtad den 28 mars 2008].

Clancy, Annette, *qualitative versus quantitative data*, http://www.interactions.biz/blog/2007/08/qualitative_versus_quantitativ.html, [Hämtad den 13 juni 2008].

McClurg-Genevese, Joshua David, *The Principles of Design*, http://www.digital-web.com/articles/principles_of_design, [Hämtad den 16 juni 2008].

Hirschi, Hans M., *Platonmetoden – Internetpedagogik*, Industrilitteratur, Stockholm, andra utgåvan, 2002.

Kolko, Jon, Beldner, Ellen, Gajendar, Uday, Connors, Chris, Petro, Justin, *Thoughts on Interaction Design*, <http://www.thoughtsoninteraction.com/index.html>, [Hämtad den 13 juni 2008].

Wikipedia, *Software Prototyping*, http://en.wikipedia.org/wiki/Software_prototyping, [Hämtad den 1 augusti 2008].

Wikipedia, *Web application*, http://en.wikipedia.org/wiki/Web_application, [Hämtad den 20 juli 2008].

Bilaga A - Intervjufrågor

Den intervjuades yrke: Human Resources Datum: 2008-05-12

Inledande om nyanställning

Vilka typer av moment/uppgifter ska en nyanställd gå igenom när han/hon kommer till företaget? Exempel?

Har ni en plan att gå efter eller introducerar ni information allt eftersom det blir relevant?

Vad har tidigare fungerat bra/dåligt när ni introducerat nyanställda?

Vad är din första spontana reaktion till ett interaktivt introduktionsprogram? Fördelar/Nackdelar?

Om programmet

Finns det uppgifter en nyanställd inte kan ta del av i ett sådant här program?

Hur mycket tid är rimligt att nyanställda lägger på introduktionen? Chefer och andra anställda?

Hur mycket information kan en nyanställd ta in när nervositet, prestationskrav och stress påverkar?

Vilken typ av information vill man läsa om och vilken typ av information vill man få direkt från en person?

Kan det finnas känslig information som man inte bör ha med? För företaget eller individen.

Hur noga vill chefen följa framstegen? Daglig uppdatering?

Vågar man som nyanställd ifrågasätta eller kritisera företaget i en utvärdering?

Utvärderingar som är anonyma eller inte? Både och?

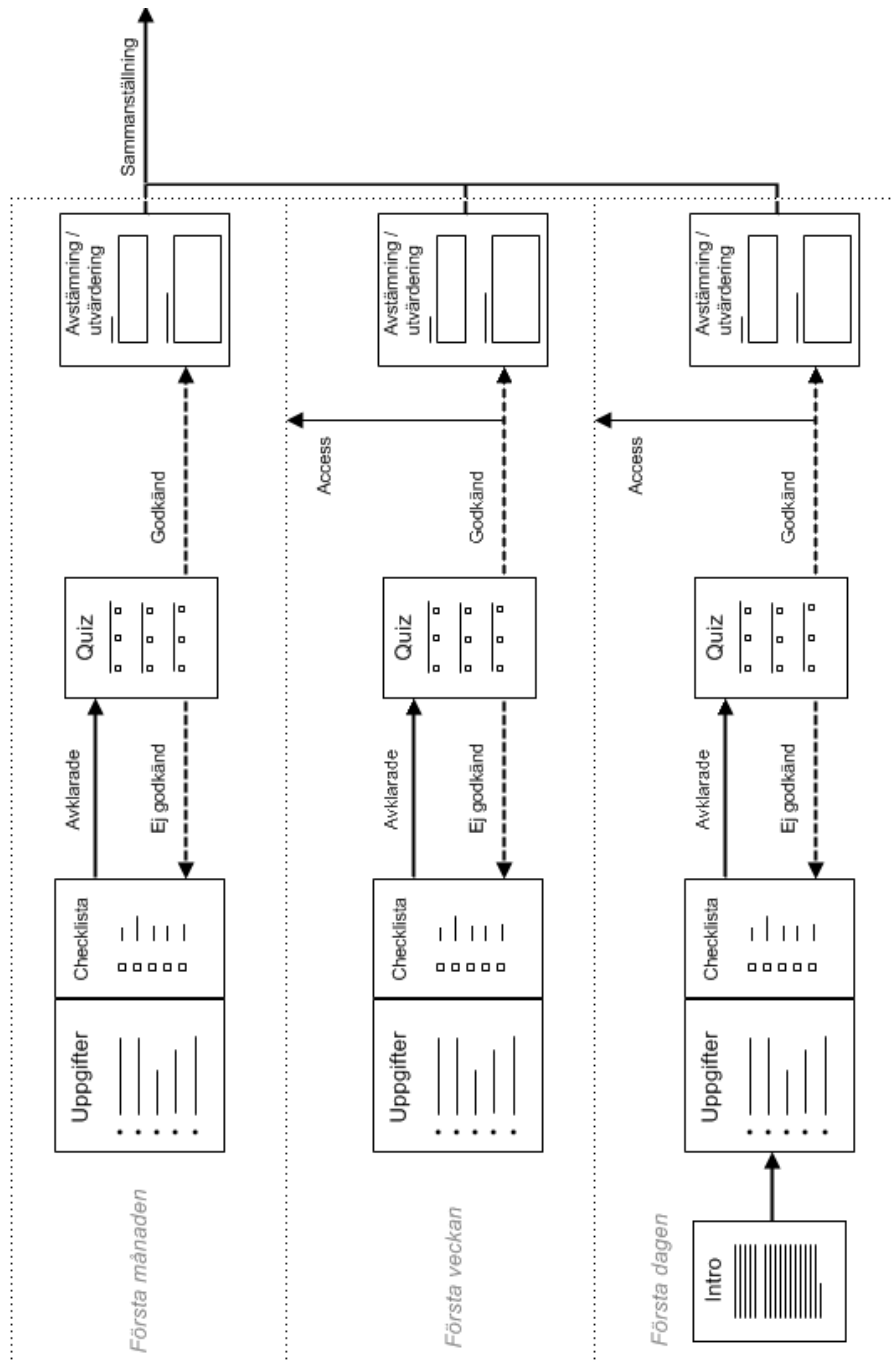
Avslutande

Idéer om vad ett introduktionsprogram skulle kunna innehålla?

Finns det något annat man bör tänka på när man skapar ett sådant här program?

Skulle det här introduktionsprogrammet vara intressant för ert företag? Varför/varför inte?

Bilaga B – Processen



Bilaga C – Användningsfall

