

Projekt i Programmering C

Menu til hjemmeside.

<u>Punkt 1</u>	<u>Punkt 2</u>	<u>Punkt 3</u>	<u>Punkt 4</u>
CSS under-menu		<u>Sub Menu 1</u>	.
		<u>Sub Menu 2</u>	
Det er til hoved-menu-		<u>Sub Menu 3</u>	
		<u>Sub Menu 4</u>	
Dette er bare en tekst på siden.			

Synopsis.

Projektet her er en menu til hjemmesider, som er lavet så den er let at tilrette, når man tilføjer eller fjerner sider. Ideen i at tilføje og fjerne sider fra menuen er, at man kun skal rette i de data der knytter sig til menuen, og ikke rette i koden der laver menuen.

Menuen er udformet, så der er en hovedmenu som altid er synlig, og til hvert punkt i hovedmenuen kan der være en række underpunkter.

Menuen er lavet i JavaScript, og er udviklet, så den fungerer i Internet Explorer 6.0.

Det at kunne tilrette menuen er lavet ved at alle menupunkter (teksten) og de tilhørende referencer til undersider er placeret i nogen arrays, så det er nemt at tilrette menuen. Menuen og det stylesheet der hører til, skal der bare henvises til i alle de sider der er i menuen.

Man kan også rette noget i udseendet på menuen, dels ved at rette i style-sheetet, og dels ved at rette i de konstanter der bestemmer placering og størrelse af menuen.

Disse ting kan rettes uden man skal have den dybere forståelse af hvordan koden fungerer.

Projektet er lavet som et eksempel, til at dokumentere, for at vise hvordan en rapport til et eksamensprojekt i Programmering C kunne se ud

Indholdsfortegnelse

Synopsis	2
Indholdsfortegnelse	3
Indledning	3
Problemformulering	4
Problemanalyse	4
Programmets overordnede retningslinier	5
Valg af miljø og programmeringssprog	5
Menuens udseende	5
Menuens funktion	6
Design af koden	6
Datastrukturer	6
Koden der skriver menuen	7
Events i menuen	7
Funktioner i programmet	8
Konstanter i programmet	8
Anvendelsen af menuen	9
Begrænsninger i menu-koden	9
Bilag	11
Bilag 1 – Koden i menu.js	11
Bilag 2 – Stylesheet til menuen, menu.css	13

Indledning

Projektet her er lavet som et eksempel på et programmeringsprojekt til faget Programmering C, mest for at illustrere hvordan man kan dokumentere et programmeringsprojekt

Selve koden er et af de eksempler der bruges i undervisningen. Eksemplet er lavet for at illustrere hvordan man kan lette programmeringen i JavaScript, ved at anvende CSS (Cascading Style Sheets).

Problemformulering.

Hvordan kan man opbygge en menu, der kan holde styr på mange undersider, så den på en nem måde kan tilpasses forskellige hjemmesider.

Hvordan kan man lave koden, så man kun skal angive de ting man ønsker at tilføje til menuen, og koden så viser menuen ud fra det man har angivet.

Problemanalyse.

Menuen skal egentlig ligne mange andre, med en hovedmenu og med mulighed for undermenuer til hvert punkt i hovedmenuen.

Menuen skal gå igen på alle undersider og have samme udseende, så det skal være den samme kode der laver menuen på alle sider.

Det er vigtigt at man kun skal rette et sted, når man vil rette i menuen.

Udseendet af menuen skal være som følger:

Hovedpkt. 1	Hovedpkt. 2	Hovedpkt. 3	Hovedpkt. 4	Hovedpkt. 5
-------------	-------------	-------------	-------------	-------------

Når man fører musen hen over et af punkterne i hovedmenuen, så skal den tilhørende undermenu komme frem, så der kommer følgende udseende frem:

Hovedpkt. 1	Hovedpkt. 2	Hovedpkt. 3	Hovedpkt. 4	Hovedpkt. 5
	Underpkt. 1			
	Underpkt. 2			
	Underpkt. 3			
	Underpkt. 4			

Når musen føres hen over et nyt punkt i hovedmenuen lukkes det sidste underpunkt og en ny undermenu kommer frem.

Samtidigt med at musen føres over et punkt, skal menupunktet vise at det er aktivt, f.x. ved at baggrunden i feltet skifter baggrundsfarve.

Når man klikker på et menupunkt, så er det selvfølgelig meningen at den side der svarer til det menupunkt man har klikket på, skal komme frem.

Programmets overordnede retningslinier

Her beskrives det der er forudsætningerne for programmet.

Valg af miljø og programmeringssprog

Miljøet som menuen skal arbejde i er givet ved at det er en hjemmeside, men i første omgang sættes den begrænsning, at menuen skal arbejde i Internet Explorer, og kun testes i version 6.0.

Som programmeringssprog vælges javaScript, dels fordi det er det undervisningssprog der har været brugt i faget, men også fordi det afvikles client-side, og derfor vil arbejde hurtig. Man kunne lige så godt opbygge menuen i et server-side sprog som ASP eller PHP, men da vi skal bruge javaScript for at lave de grafiske funktioner, så vil det blot komplicere projektet, hvis man skulle skrive det i to forskellige sprog.

Menuens udseende

For at få menuen til at se ud som ønsket, lægges det ind i en tabel, der er den nemmeste måde at styre det på.

De sorte rammer omkring menuen defineres ved at lave et style-sheet, der angiver rammerne, understregningen og at der kommer en hånd frem, så man kan se det er et link og baggrundsfarven på feltet.

For at få rammen til at fylde det samme rundt om menuen og inde imellem menupunkter er det nødvendigt at man kun sætter kant på de 3 af siderne. Der kommer så til at mangle en kant på den sidste side i hovedmenuen, den sættes på, når man opbygger menuen. Mere om det i beskrivelsen af koden.

Det er kun hovedmenuen der skal vises, når brugeren ikke anvender menuen. Derfor anvendes egenskaben visibility for at skjule punkterne i submenuen og få dem frem enkeltvis. Dette sker ved hjælp af programmet.

Menuen får følgende udseende, når man har bevæget musen hen over punkt 3 i hovedmenuen, og videre ned over Sub Menu 3 – her får Sub Menu 3 en anden baggrund. Når cursoren bevæger sig i menuen er den normale cursor-pil ændret til en hånd.

<u>Punkt 1</u>	<u>Punkt 2</u>	<u>Punkt 3</u>	<u>Punkt 4</u>
CSS under-menu		<u>Sub Menu 1</u>	
		<u>Sub Menu 2</u>	
Det er til hoved-menu-		<u>Sub Menu 3</u>	
		<u>Sub Menu 4</u>	

Dette er bare en tekst på siden.

For at vise at det er et menupunkt, som kan udføre en aktivitet, sættes en baggrundsfarve på det menupunkt som musen står over. Det er igen programmet der håndterer dette ved hjælp af egenskaben `backgroundColor`.

Menuens funktion

For at menuen skal virke overskuelig, så er det kun et sæt undermenuer der bliver vist af gangen. Det gøres ved at skjule alle undermenuer inden en ny vises.

For at dokumentet skal kunne læses, så skal undermenuen også forsvinde igen. Den er sat til at forsvinde efter 2 sekunder, eller hvis brugeren klikker nede i dokumentet.

Hvis brugeren klikker i menuen, så skal brugeren selvfølgelig hen på den side der svarer til det ønskede menupunkt.

Design af koden

Datastrukturer.

Hvert punkt i hovedmenuen kræver to oplysninger, nemlig hvad der skal stå i menupunktet, og hvilken side der skal åbne, når man klikker på menupunktet.

Disse oplysninger lægges i to arrays:

```
var menuText = new Array("Punkt 1", "Punkt 2", "Punkt 3", "Punkt 4");
var menuURL = new Array("css2.html", "side2.html", "side3.html", "side4.html");
```

Dette er en-dimensionelle arrays, da menuen kun kan have en række menu-punkter. Undermenuerne har samme struktur, men her er det hvert punkt i hovedmenuen, der har en undermenu. Derfor oprettes to arrays, der i hvert punkt indeholder et array, så det bliver to to-dimensionelle arrays.

Arrayene og oprettelsen af den første undermenu ser ud som følger:

```
var subMenuText = new Array();
var subMenuURL = new Array();
subMenuText[0] = new Array("Underpunkt 1", "Underpunkt 2", "Underpunkt 3");
subMenuURL[0] = new Array("side1-1.html", "side1-2.html", "side1-3.html");
```

Læg mærke til at første punkt hedder nummer 0

Ud over disse arrays, der definerer hvad der skal være i menuen, er der et array, som angiver hvor brede de enkelte menupunkter skal være:

```
var menuBredde = new Array(100, 120, 80, 100);
```

Koden der skriver menuen

Hovedmenuen skrives ud i en HTML-tabel ved hjælp af noget kode ud fra den datastruktur der er beskrevet i foregående afsnit. Koden beskrives her ved hjælp af pseudokode.:

Skriv starten på tabellen og starten på en række

For alle punkter i menuText:

 Skriv starten af en tabelcelle med et unikt ID.

 Hvis det er den første celle så:

 Sæt en venstre ramme på tabelcellen

 Skriv cellen med angivelse af bredde, indhold, og hvad cellen skal reagere på

 Afslut tabelcellen

Skriv afslutningen på tabelrækken og af tabellen

Undermenuen skrives igen ud i tabeller og igen ud fra de datastrukturer der er beskrevet. Hvert enkelt punkt i menuen skriver sin egen tabel, og positionen af tabellerne styres absolut.

Koden her beskrives ved hjælp af et NS-diagram, blot for at vise, at der er forskellige måder at dokumentere kode på.

Sæt vandret offset til venstre menukant
For alle punkter i menuTekst
Skriv starten på en tabel med sit eget unikke ID, sæt den til skjult
Placer den ud fra vandret offset.
Beregn offset til den næste submenu
For alle punkter i den aktuelle submenu
Skriv en tabelrække med en celle, der har den ønskede bredde
Sæt et unikt ID på cellen
Skriv hvad cellen skal reagere på og indholdet af cellen
Afslut tabellen

Events i menuen

Menuen skal reagere på musen ved forskellige ting. De events der bruges er følgende:

`onMouseOver()`

`onMouseOut()`

`onClick()`

Disse events kalder forskellige funktioner, der beskrives i næste afsnit.

En sidste event der anvendes er en timeout, der sættes op, når en submenu bliver vist. Når timeouten indtræffer, så gøres alle submenuen usynlige.

Objekter i menuen

Alle tabelceller både i hovedmenuen og i submenuerne er lavet til objekter. Det kan virke lidt omstændeligt, men er gjort for at kunne henvise til hver enkelt tabelcelle, når man skal skifte farve på dem. Hver submenu er også et objekt, så man kan sætte dem synlige og usynlige.

Navnet på hvert objekt (sat med ID i deres respektive tags) er lavet struktureret, så man kan henvise til dem fra funktionerne.

Alle hovedmenuerne hedder menu0, menu1 osv.

Alle submenuerne til hovedmenu 0 hedder menu00 menu01 menu02 osv. Og til hovedmenu 1 hedder de menu10 menu11 menu12 osv.

Tabellerne med submenuerne hedder sub0, sub1 osv.

Funktioner i programmet

Funktionen `hideall()` sætter alle undermenuernes `visibility` til `hidden`, så alle undermenuer skjules. Denne funktion kaldes dels fra de andre funktioner, dels fra `timeout` og dels fra en `onClick()` på teksten (mere herom senere).

Som omtalt så skal der ske forskellige ting, når musen bevæges rundt og der klikkes.

Fra punkterne i hovedmenuen kaldes to forskellige funktioner:

Funktionen `overMain(nr)` kaldes når musen bevæger sig ind over et menupunkt i hovedmenuen.

Funktionen `outMain(nr)` kaldes når musen bevæger sig ud af punktet igen.

`nr` er en parameter der sættes på i menuen, som angiver hvilket menupunkt det er.

Det der sker i `overMain(nr)` er at menupunktet farves med en anden baggrundsfarve, og for at sikre at der kun vises en menu ad gangen, så kaldes `hideall()` inden den ønskede submenu laves synlig.

I `outMain(nr)` farves menupunktet med den passive farve igen, og der sættes et `timeout` op, så `hideall()` kaldes, hvis der ikke sker mere.

Til submenuerne er der ligeledes to funktioner:

Funktionen `overSub(nr, subm)` kaldes når musen bevæges ind over et punkt i en submenu.

Funktionen `outSub(nr, subm)` kaldes når musen bevæges ud af punktet igen.

`nr` er igen en parameter, der angiver hvilken hovedmenu punktet er tilknyttet.

`subm` er så en parameter, der angiver hvilket punkt i submenuen der arbejdes med.

I `overSub(nr, subm)` sættes baggrundsfarven på det aktuelle punkt til at være aktiv, og `timeout` sættes igen op til at lave `hideall()` efter 2 sekunder. Inden `timeout` sættes op, kontrolleres lige om der er et andet `timeout` der kører, hvis der er det, så fjernes det, så det ikke kommer til at lave konflikt.

I `outSub(nr, subm)` sættes baggrundsfarven på punktet bare tilbage til passiv.

Konstanter i programmet

For ikke at have tal stående rundt i programmet, der angiver forskellige ting bl.a. omkring placeringen af menuen, så er der i starten erklæret en række konstanter. `JavaScript` har ikke nogen konstant-definition, derfor er de erklæret i en `var`-sætning, men de anvendes som konstanter:

`menuTop` = 10 – det antal pixels menuen er fra toppen.

`subMenuTop` = 37 - det antal pixels submenuen er fra toppen.

`menuLeft` = 10 – det antal pixels menuen er fra venstre kant.

subMenuEkstra = 7 - det antal pixels ekstra, der er ud over menubredden til hver submenu.

Det er to gange textSpace og en pixel til rammen

textSpace=3 – det luft der skal være rundt om teksten i menuen, i pixels.

Anvendelsen af menuen

For at anvende menuen skal den naturligvis ind i alle de undersider der henvises til, så det er den samme menu der kommer på alle sider.

Der gøres ved at placere koden i filen ”menu.js” (se bilag 1).

Ligeledes skal menuen have de rigtige stylesheets, det gøres ved at gemme dem i filen ”menu.css” (se bilag 2).

Det man så gør er, at man i toppen af siden skriver følgende:

```
<head>
<link rel="stylesheet" type="text/css" href="menu.css">
</head>
<body onClick="hideAll();">
<script type="text/javascript" language="JavaScript" src="menu.js"></script>
<br />
```

Den onClick der er i body-tag’et er lavet så brugeren kan slukke for menuen inden de to sekunders timeout, blot ved at klikke et eller andet sted på siden.

Efter linieskiftet
 er det bare at sætte det indhold der skal være på siden ind i filen.

Et eksempel på menuen kan ses på <http://www.holstebrohtx.dk/bar/javascr/opg/css/css2.html> eller man kan hente hele menuen med undersider og stylesheet på <http://www.holstebrohtx.dk/bar/javascr/opg/css/css2.zip>

Begrænsninger i menu-koden

Den måde de ID’er der er sat på de enkelte punkter i submenuerne kan give en begrænsning, hvis man arbejder med mange menupunkter.

Har man 11 hovedmenu-punkter vil ID’et komme til at hedde menu10

Hvis man så samtidigt har en submenu i den anden hovedmenuen, så ville det første punkt i dens submenu komme til at hedde menu10 – altså det samme, hvilket ville give anledning til konflikt.

For at undgå det ville det være nødvendigt at lave lidt mere kreative navne på ID’et, f.x. ved at sætte et s ind imellem hovedmenu-nummeret og submenu-nummeret, så de to omtalte ID’er ville komme til at hedde menu10 og menu1s0.

En anden nok lige så realistisk begrænsning, som man nok ville støde på før, ville være den rent grafiske bredde af menuen.

Test

Siden er kun gennemtestet i Internet Explorer 6.0. Jeg har ikke haft den åbnet i Netscape, men i Mozilla har den en enkelt fejl, hvor rammen mellem hovedmenuen og submenuen forsvinder. Det kunne let klares ved en browser-test og skelnen mellem hvilken browser koden arbejder i, ved at lave forskellig placering, alt efter hvilken browser det er. Ud over det fungerede koden som den skulle.

Perspektivering

En ting der kunne struktureres anderledes i koden er hvordan dataene der angiver menuens indhold skal defineres.

I denne version er der valgt at lægge dem i simple arrays.

Det kunne måske give en lidt mere logisk struktur, specielt på hovedmenuen, at angive dem som et array af objekter, hvor man kunne indkapsle både hovedmenu-punktet, men også hele submenuen i objektet.

Det kunne måske også give en lidt pænere erklæring af menupunkterne.

En anden ting der kunne tilføjes var endnu et eller flere niveauer mere i menuen. Hvordan det skulle se ud rent grafisk kan måske være lidt svært at se, men der kunne måske laves en eller anden forskel i typografien, hvis der var underpunkter til et submenupunkt, og den sub-sub-menu der så skulle komme kunne tilføjes ved at placere menuen ved siden af submenuen.

Der kunne også gøres en del ved layoutet, da hovedparten er holdt til at være standard web-typografi.

Konklusion

Formålet, at lave en menustruktur, der er let at redigere, må siges at være opfyldt til to niveauer. Tingene fungerer og menuen er fornuftig brugervenlig, da den opfører sig rimeligt logisk i forhold til hvad man ellers ser af menuer på nettet.

Der mangler test af produktet i forskellige browsere, men det vurderes at der ikke ville være det store arbejde i at få tingene til at fungere.

Det vil også være en begrænsning at der kun er de to niveauer, hvis menuen skal anvendes til et større websted med mange undersider, men det vil kræve en gennemgribende omskrivning at tilføje flere niveauer, mens det vil være relativt enkelt at løse begrænsningen med 11 menuer i bredden.

Bilag

Bilag 1 – Koden i menu.js

```
// Definitioner på menuens størrelse og placering
var menuTop = 10, subMenuTop = 37, menuLeft = 10, subMenuEkstra = 7, textSpace=3;
var menuBredde = new Array(100, 120, 80, 100);

// Tekst og henvisninger til hovedmenuen
var menuText = new Array("Punkt 1", "Punkt 2", "Punkt 3", "Punkt 4");
var menuURL = new Array("css2.html", "side2.html", "side3.html", "side4.html");

// Tekster og henvisninger til submenuerne
var subMenuText = new Array();
var subMenuURL = new Array();
subMenuText[0] = new Array("Underpunkt 1", "Underpunkt 2", "Underpunkt 3");
subMenuURL[0] = new Array("side1-1.html", "side1-2.html", "side1-3.html");
subMenuText[1] = new Array("Sub 1", "Sub 2", "Sub 3", "Sub 4", "Sub 5", "Sub 6");
subMenuURL[1] = new Array("side2-1.html", "side2-2.html", "side2-3.html", "side2-4.html", "side2-5.html", "side2-6.html");
subMenuText[2] = new Array("Sub Menu 1", "Sub Menu 2", "Sub Menu 3", "Sub Menu 4");
subMenuURL[2] = new Array("side3-1.html", "side3-2.html", "side3-3.html", "side3-4.html");
subMenuText[3] = new Array();
subMenuURL[3] = new Array();

// Farvedefinitioner
var passive = "#FFFFFFE", active = "#FFFF99";

// Variabel til timeren, der skal indeholde slette-tiden
var timer = null

// Funktion der laver alle submenuer usynlige
function hideAll() {
    if (timer) { window.clearTimeout(timer) }
    for (var n in menuText) {
        var navn = "sub" + n;
        pkt = document.getElementById(navn);
        pkt.style.visibility = 'hidden';
    }
}

// Funktion viser et hovedmenupunkt som aktivt, og sætter den tilhørende submenu synlig
function overMain(nr) {
    var navn = "menu" + nr;
    var pkt = document.getElementById(navn);
    pkt.style.backgroundColor = active;
    hideAll();
    navn = "sub" + nr;
    pkt = document.getElementById(navn);
    pkt.style.visibility = 'visible';
}

// Funktion viser et hovedmenupunkt som passiv, og sætter submenuen til at forsvinde efter 2 sek.
function outMain(nr) {
    var navn = "menu" + nr;
    var pkt = document.getElementById(navn);
    pkt.style.backgroundColor = passive;
    timer = setTimeout("hideAll()", 2000);
}

// Funktion viser et submenupunkt som aktivt og sætter igen tiden til 2 sek.
function overSub(nr, subm) {
    var navn = "menu" + nr.toString() + subm.toString();
    var pkt = document.getElementById(navn);
    pkt.style.backgroundColor = active;
    if (timer) { window.clearTimeout(timer) }
    timer = setTimeout("hideAll()", 2000);
}

// Funktion viser et submenupunkt som passivet
function outSub(nr, subm) {
    var navn = "menu" + nr.toString() + subm.toString();
    var pkt = document.getElementById(navn);
    pkt.style.backgroundColor = passive;
}
```

```
// Herunder ligger koden, der skriver tabellerne ud, som defineret i variablerne

// Lav tabellen med hovedmenuen ud fra menu-parametrene og arrayene med menu-indholdet
document.write("<table cellpadding='" + textSpace + "' cellspacing='0' style='position: absolute;
top:" + menuTop + "; left:" + menuLeft + ";><tr>");
for (var n in menuText) {
    document.write("<td class='hMenu' id='menu" + n + "' ");
    if (n == 0) document.write("style='border-left: black 1px solid;' ");
    document.write("width='" + menuBredde[n] + "' onMouseOver='overMain(" + n + ")' ");
    document.write("onMouseOut='outMain(" + n + ")' onClick='window.location.href=\"" +
menuURL[n] + "\">" + menuText[n] + "</td>");
}
document.write("</tr></table>");

// Lav et antal tabeller med undermenuerne ud fra parametrene og arrayene med indholdet til
submenuerne
var offset = menuLeft;
for (n in menuText) {
    document.write("<table cellpadding='" + textSpace + "' cellspacing='0' id='sub" + n +
"' style='position: absolute; top:" + subMenuTop + "; left:" + offset + "; visibility: hidden;>");
    offset += menuBredde[n] + subMenuEkstra;
    // Lav det antal underpunkter der er i den aktuelle submenu
    for (i in subMenuText[n]) {
        document.write("<tr><td class='subMenu' width='" + menuBredde[n] + "'
onMouseOver='overSub(" + n + ", " + i + ")' ");
        document.write("onMouseOut='outSub(" + n + ", " + i + ")' id='menu" + n +
i + "' ");
        document.write("onClick='window.location.href=\"" + subMenuURL[n][i] +
"\">" + subMenuText[n][i] + "</td></tr>");
    }
    document.write("</table>");
}
}
```

Bilag 2 – Stylesheet til menuen, menu.css

```
.hMenu {
    background-color: #FFFFFFEE;
    border-right: black 1px solid;
    border-top: black 1px solid;
    border-bottom: black 1px solid;
    text-align: center;
    text-decoration: underline;
    cursor: hand;
}
.subMenu {
    background-color: #FFFFFFEE;
    border-left: black 1px solid;
    border-right: black 1px solid;
    border-bottom: black 1px solid;
    text-align: center;
    text-decoration: underline;
    cursor: hand;
}
```