

Tilgjengelige nettsteder

2:3 - Design og koding

```
<!DOCTYPE html PUBLIC  
"-//W3C//DTD XHTML 1.0 Strict //E  
"http://www.w3.org/TR/xhtml1/DTD  
<html lang="no">
```



Universell utforming

"Universell utforming er utforming av produkter og omgivelser på en slik måte at de kan benyttes av alle mennesker, i så stor utstrekning som mulig, uten behov for tilpassing eller spesiell utforming."

(The Center for Universal Design, North Carolina State University)

Tilgjengelig nettsted

For nettsteder betyr det at de må utformes med tanke på den store variasjonen som finnes i befolkningen. Barn, voksne, eldre, blinde, synshemmede, døve, hørselshemmede, bevegelseshemmede skal alle ha muligheten til å benytte mulighetene som Internettet tilbyr. Ikke bare fordi det er pålagt, eller fordi det er diskriminerende å stenge ute deler av befolkningen, men også fordi personer med ulike funksjonsnedsettelse utgjør en stor del av markedet for mange varer og tjenester.

Utgitt av Deltasenteret – Statens kompetansesenter for deltakelse og tilgjengelighet for mennesker med funksjonshemming.

Deltasenterets hovedmål er å bidra til at mennesker med nedsatt funksjonsevne kan delta i samfunnet på linje med andre. Visjonen er deltakelse og tilgjengelighet for alle.

Deltasenterets arbeid er basert på at universell utforming, tilrettelegging av miljø, teknologi og hjelpemidler sammen bidrar til økt deltakelse og tilgjengelighet for mennesker med redusert funksjonsevne.

Arbeidet består i å:

- identifisere funksjonshemmende barrierer og vise hvordan disse kan reduseres eller fjernes
- utvikle kunnskap om og være rådgiver for god tilgjengelighet
- være pådriver i samhandling med ansvarlige myndigheter
- synliggjøre behov for tverrfaglig innsats

Nettside: www.shdir.no/deltasenteret

Forord

Denne veilederen er den andre i en serie på tre veiledere om tilgjengelige nettsteder.

Deltasenterets motivasjon for å gi ut serien er:

- å inspirere til økt arbeid med tilgjengelighet på nettsteder
- å følge opp politiske målsettinger om flere tilgjengelige nettsteder
- å gi oversiktlig og praktisk innføring i hvordan tilgjengelige nettsteder lages

Målgruppen for serien er studenter, designere, utviklere, innkjøpere og eiere av nettsteder.

I 1993 sluttet Norge seg til FNs standardregler om like muligheter for funksjonshemmede. Siden den gang har skiftende regjeringer utarbeidet ulike handlingsplaner med sikte på forbedret tilgjengelighet. I 2001 kom den offentlige utredningen NOU 2001:22: "Fra bruker til borger". I 2003 kom St.meld. 40 (2002-2003) "Nedbygging av funksjonshemmende barrierer" og i 2004 kom "Regjeringens handlingsplan for økt tilgjengelighet for personer med nedsatt funksjonsevne". Ett av tiltakene i denne planen er å innføre god tilgjengelighet ved bruk av WAI-kriteriene på offentlige nettsteder.

Statens informasjonstjeneste tok i 2001 initiativ til utviklingen av kvalitetskrav for offentlige nettsteder, og Statskonsult testet disse kravene på offentlige nettsider i 2001 og 2003. Norge.no overtok ansvaret for å vurdere kvaliteten på offentlige nettsider i 2004 og samtidig fikk tilgjengelighet mer fokus i kvalitetskravene. I eNorge 2009 – planen er det et mål at i løpet av 2007 skal 80 % av offentlige nettsteder oppfylle Norge.no sine kriterier for tilgjengelighet.

I revidert lov om offentlige anskaffelser (januar 2007) kreves det at universell utforming skal være et utvalgsriterium ved alle offentlige anskaffelser. Når det gjelder privat sektor finnes det for øyeblikket ingen krav om tilgjengelige nettsteder. En rekke land, blant annet USA og England, har i flere år hatt lover som gjør diskriminering av personer med nedsatt funksjonsevne ulovlig og setter krav til at private virksomheters nettsteder skal være tilgjengelige for funksjonshemmede. I Norge er det også fremmet forslag til antidiskrimineringslov (NOU 2005:8 "Likeverd og tilgjengelighet").

Denne serien er basert på fem hefter kalt "Websites for flere", utgitt av Dansk Center for Tilgængelighed. Vi takker for velvillig tillatelse til å benytte dette materialet.

Oslo, november 2006



Toril Bergerud Buene
Leder av Deltasenteret

Innhold

Innledning	5
1. Tilgjengelighet – en positiv utfordring	6
2. Design	7
2.1 Layout	7
Kontroll over nettsteders utseende	7
Stilark /CSS	7
Andre muligheter	8
Menyer	9
Plug-ins og applets	9
2.2 Ikoner og bilder	10
2.3 Farger	11
2.4 Teksten	12
Tekststruktur.....	12
Overskrifter.....	12
Avsnitt og linjeskift.....	13
Skriftstørrelse	14
Farger og skrifttyper	14
Kontrast.....	15
2.5 Navigering.....	16
Lenker	16
Hjelp brukeren å navigere	17
Unngå lenker som åpner nye vinduer.....	18
2.6 Testing	19
3 Koding – råd til programmerere	20
3.1. Velg og deklarer HTML-standard	21
3.2. Bruk CSS og overhold standarden.....	23
3.3. Angi språk.....	24
3.4. Gi alle sider en tittel.....	25
3.5. Kod overskrifter korrekt	25
3.6. Unngå identiske lenkebetegnelser	26
3.7. Skriv meningsbærende alt-tekster.....	27
3.8. Bruk datatabeller korrekt.....	27
3.9. Bruk label-elementet på skjemaer.....	28
3.10. Bruk Javascript med omtanke	29
3.11. Bruk plug-ins med omtanke	31
3.12. Sørg for tilgjengelighet hele veien	32
4. Nyttige verktøy og referanser	33
4.1 Tilgjengelighetsverktøy	33
4.2 HTML - verktøy	34
4.3 Generell kunnskap	34

Innledning

Du som leser denne veilederen har trolig vært med på å lage en del nettsteder. Noen har du laget i en håndvending og andre har du jobbet lenge og mye med. Du kan ha testet nettstedet i Netscape, Opera, Internet Explorer eller i andre nettlesere, men har du også vurdert om designet er optimalt for alle brukere? Har du testet hvordan nettstedet framstår når standardinnstillingene i nettleseren er endret, eller når noen har behov for ekstra stor skrift eller bedre kontrast?

I og med at brukere kan påvirke visning av nettsteder ved å bruke innstillinger i nettleseren, kan det være god en ide å kjenne til disse mulighetene. De innstillingsmulighetene vi refererer i denne veilederen, er basert på Internet Explorer og Windows XP. De fleste andre nettlesere har imidlertid liknende muligheter.

Det er også viktig å vite noe om hvordan personer med ulike funksjonsnedsettelse bruker datautstyr og hvordan det kan tas hensyn til ved utformingen av et nettsted. Den første veilederen beskrev alternative måter å navigere på. Denne veilederen følger opp med kodeeksempler rettet mot brukere med ulike funksjonsnedsettelse.

Dette er den andre veilederen i en serie på tre og retter seg mot to av hovedprosessene i utviklingen av et nettsted:

- Kapittel 2 retter seg mot designfasen. Her finner du råd om hvordan nettsteder kan gjøres tilgjengelige. Med designere tenker vi her både på personer som lager et grafisk design i for eksempel PhotoShop og andre liknende programmer, og dessuten på prosessen med å gjøre om designet til HTML-baserte sidemaler og eksempelsider.
- Kapittel 3 retter seg mot koding av tilgjengelige nettsteder og tar for seg 12 praktiske råd om hvordan tilgjengelighet kan oppnås

Det finnes svært mye informasjon om HTML, CSS, tilgjengelighet og webutvikling generelt. Det vi ønsker å tilføre med våre veiledere, er å gi leserne korte, praktiske innføringer i temaer rundt tilgjengelighet på nett og referanser til annen praktisk informasjon. Vårt håp er at eiere og utviklere av nettsteder skal bruke veilederne som et verktøy til å komme videre med tilgjengelighet.

Redaksjonen har bestått av Frank Fardal fra Norge.no og Aina Olsen og Haakon Aspelund fra Deltasenteret.

1. Tilgjengelighet – en positiv utfordring

Hvordan kan man som designer lage en tilgjengelig løsning? Krav om tilgjengelighet kan virke vanskelig fordi designeren må ta hensyn til forhold som ikke bare har med det visuelle å gjøre.

En årsak til at tilgjengelige nettsteder ofte blir framstilt som kjedelige, er en utbredt misforståelse om at tilgjengelige nettsteder skal være uten bilder. Mange oppfatter tilgjengelighet som tilrettelegging for blinde, og at blinde ikke bryr seg om bilder. Tilgjengelighet handler imidlertid også om mange andre brukere enn blinde. Brukt riktig utgjør heller ikke bilder noe problem for synshemmede nettbrukere. Riktig bruk av farger og bilder er tvert i mot med på å gjøre nettsidene mer tilgjengelige for mange brukere. Problemet er vel at det kan være enklere å bruke velprøvde løsninger framfor å kaste seg ut i noe nytt og spennende, med det resultatet at tilgjengelige nettsider blir kjedelige nettsider.

Nettstedet www.csszengarden.com presenterer mange eksempler på at tilgjengelig design ikke behøver å være kjedelig forutsatt at designeren ser på oppgaven som en positiv utfordring. Her eksperimenteres det med ulike design til ett og samme HTML-dokument. Les mer om dette i kapittel 4, "Nyttige verktøy og referanser".

Se på god tilgjengelighet som en positiv utfordring. Du har som designer en oppgave som skal løses innenfor bestemte rammer, men samtidig har du muligheten til å være kreativ. Denne utfordringen kan sammenliknes med de kravene kunder stiller til utvikling av grafisk profil med bestemte farger, logo og så videre.

Undersøk alltid mulighetene for å lage tilgjengelige løsninger. Les og følg leverandørenes veiledninger om hvordan ulike elementer kan brukes på en tilgjengelig måte. Macromedia og Adobe har for eksempel veiledninger som viser hvordan tilgjengelig innhold kan lages med Flash og Acrobat. Som et minimum må det sørges for alternative løsninger for personer som ikke kan bruke bestemte elementer på nettsiden.

I Skandinavia har vi en lang og god designtradisjon der funksjon og utseende går hand i hand med estetikk og funksjonalitet. Dette kan videreføres ved å innarbeide tilgjengelighet som en naturlig del av webdesignet.

Denne utfordringen er kanskje noe annerledes enn de vanlige grafiske utfordringene du er vant til. Det skyldes blant annet at du for å oppnå et godt resultat i noen tilfeller må ta hensyn til hvordan HTML-koden er bygget opp.

2. Design

2.1 Layout

Kontroll over nettsteders utseende

Det har alltid vært en utfordring for designere å få kontroll over nettstedenes utseende. Den mest radikale løsningen vil være å lage hver enkelt side, både tekst og grafikk, som et bilde. Ved å legge inn lenkene som *imagemaps* tar designeren full kontroll over hvordan siden presenteres for brukeren. Denne typen løsning gir imidlertid ikke et tilgjengelig nettsted. Brukere som ønsker teksten lest opp eller forstørret, får store problemer i og med at det ikke er noen tekst å lese opp.

Grafikk som forstørres blir "hakkete" i kantene og dermed vanskeligere å lese. Det er også et problem at sidene ikke lar seg justere slik at de kan tilpasses til den skjermen de blir vist på. Alt for mange legger ut deler av teksten i form av grafikk med den konsekvens at denne teksten ikke er tilgjengelig for alle.

Hvordan er brukeropplevelsene?

Husk at brukere i stor grad selv kan styre hvordan nettsider skal vises. Du kan som designer aldri være helt sikker på hvordan din side vil vises for brukeren. Kampen for å sikre at nettsiden alltid skal se lik ut, er tapt på forhånd. Derfor er det bedre å la brukeren være en medspiller og tilby muligheter til selv å bestemme innenfor de rammene du legger opp til.

Tenk nytt og annerledes. Lag en utforming som ser riktig ut på de vanligste nettleserne, men som samtidig gir brukeren lov til å velge. Som designer får du uansett aldri full kontroll over hvordan en nettside vises.

Overvei konsekvensene i designfasen. Hva på siden kan endre størrelse? Hva vil brukere ønske å endre via innstillinger i nettleseren? Og hvilke behov vil de i tilfelle henge sammen med?

Stilark /CSS

- Bruk Cascading Style Sheets – CSS til mest mulig. Stort sett alt lar seg gjøre med CSS. Ved å bruke CSS i separate stilark er det enklere å presentere ulike layouter og å gjøre endringer.
- Hvis du fortsatt bruker tabeller for å holde designet på plass, må du forholde deg til hvordan de vil bli lest opp, – én og én celle om gangen.
- Bruk relative enheter i stilark. Bruk for eksempel "em" i stedet for "pixel" eller "punkt". Ved å benytte samme relative enhet blir det enklere å sikre ensartet skalering av nettsidens enkeltelementer. Les mer om "em" i kapittel 2.4.

- Tilpass designet til den plassen du har til rådighet. Brukeren vil normalt selv tilpasse størrelsen på vinduet etter egne preferanser.
- Lag designet slik at innholdet kan tilpasses den plassen du har til rådighet på en skjerm. Ved skrolling er det best å lage siden slik at brukeren kun skal skrolle i én retning.

CSSZengarden

Denne hjemmesiden utfordrer designere til å bruke CSS til å lage egne utgaver av det samme HTML-dokumentet. Alle har som mål å være tilgjengelig, men i praksis lever ikke alle opp til det. Siden illustrerer likevel godt hvilken stor variasjon CSS som presentasjonsverktøy tilbyr. Se: www.csszengarden.com



Skjermdump fra nettstedet csszengarden.com. HTML dokumentet er det samme i begge tilfeller, men presentasjonen er gjort forskjellig ved hjelp av CSS.

css.maxdesign.com.au

Her finnes det en rekke CSS veiledninger/tutorials og flere eksempler på hvordan menyer kan bygges opp med CSS: Se: css.maxdesign.com.au

HTML.dk

HTML.dk inneholder en rekke veiledninger/tutorials, artikler m.m., blant annet om bruk av CSS.

www.html.dk/tutorials/css/

Andre muligheter

Brukere som har behov for å tilpasse nettsidens visuelle layout kan ta i bruk egenproduserte stilark for å formatere et nettsted. I Internet Explorer finnes denne

valgmuligheten i menyen: **Verktøy > Alternativer for Internett > Tilgjengelighet > stilark.**

Det kan være situasjoner der brukeren ut fra eget behov optimaliserer nettleseren og dermed endrer nettstedets utseende. Noen av disse innstillingsmulighetene kan testes ved hjelp av "Web Accessibility Toolbar", som er omtalt i kapittel 4, "Nyttige verktøy og referanser". Den gjør det for eksempel mulig å se en nettside uten bilder, uten bruk av stilark/CSS, i gråtoner, på mindre skjermer med mer.

Husk også at det finnes brukere som benytter nettlesere med en helt annen visning av nettsidene enn det som er vanlig. Det kan være brukere med tekstbaserte nettlesere, leselister, eller brukere av håndholdte enheter med svært små skjermer, som PDAer og mobiltelefoner. Slik bruk forventes å øke sterkt i tida framover.

Menyer

Mange nettsteder bruker "aktive" menyer av typen "drop-down" eller menyer som folder seg ut. De er som oftest basert på javascript, plug-ins eller java-applets og de er derfor ikke tilgjengelige for alle.

Slike menyer er ofte basert på bruk av mus og reagerer ikke på tastatur. De vil derfor kunne utelukke brukere som ikke kan benytte mus. For eksempel synshammede.

Menyer av denne typen er også svært følsomme og lukkes med en gang markøren beveges utenfor menyen. Det gjør at slike menyer er veldig vanskelige å benytte for mange, særlig for personer som har problemer med å peke med stor presisjon. Dette kan gjelde nybegynnere, eldre og andre med dårlig hånd- og armfunksjon.

For blinde er det dessuten et problem at skjermleserne ikke finner menypunktene.

Hvordan gjøre menyer tilgjengelige?

Aktive menyer har sine fordeler. Spørsmålet er hvordan slike menyer kan gjøres tilgjengelige.

For det første er det viktig å sikre at funksjonaliteten kan benyttes uten bruk av javascript, plug-ins eller java-applets. Som et minimum må det sørges for en alternativ meny for dem som ikke har mulighet til å kunne bruke den teknologien som er brukt for å lage menyen. For det andre må menyer kunne betjenes ved hjelp av tastaturet. Alternativt kan menyoverskriften lages som en lenke som kan betjenes via tastaturet, og som sender brukeren til en side med tilgjengelige menyvalg.

Flere programleverandører leverer tilgjengelige menyvarianter basert på DHTML. Det er dermed ikke nødvendig å programmere slike menyer fra bunnen av. Søk etter "accessible DHTML menu" på nettet.

Plug-ins og applets

Med plug-ins eller applets er det mulig å få til funksjonalitet som ikke er mulig i HTML. Det kan finnes gode grunner for å ta dem i bruk, men slike programmer kan også skape problemer. Det er derfor viktig å være klar over vanlige problemer som kan oppstå og som det må tas hensyn til om du bruker disse teknikkene.

Forhold det må tas hensyn til:

- Muligheter for betjening via tastatur
- Synkronisert tekstlig beskrivelse av visuelt innhold. Dette gjelder for eksempel animasjoner og video.
- Synkronisert beskrivelse av lydinnhold.
- Mulighet for å kunne kontrollere hvordan funksjonaliteten virker, for eksempel å kunne stoppe igangsatte animasjoner.
- Struktur i innholdet - for eksempel når innholdet skal presenteres ved hjelp av skjermleserprogrammer.
- Kommunikasjon med kompensierende hjelpemidler, for eksempel med opplesningsprogrammer.

2.2 Ikoner og bilder

De fleste nettsider inneholder mye tekst som må leses. Mye tekst gjør det vanskelig for personer med lesevansker å navigere på nettet. Mange bruker programmer som leser teksten med syntetisk tale, men det tar svært lang tid å få lest seg gjennom et nettsted på denne måten. Mange med lesevansker orker derfor ikke ta i bruk et nettsted som består av en mur med tekst.

Lese- og skrivevansker er utbredt. Nesten 30 prosent av den voksne befolkningen i Norge har problemer med lesing og skriving i en eller annen form. Tallene finner vi i internasjonale undersøkelser initiert av OECD hvor Norge har deltatt (SIALS 2000 og ALL 2005). Se: Fakta om lesing og skriving, Vox Læring for arbeidslivet <http://www.vox.no/templates/CommonPage.aspx?id=1641> - 5. mai 2006.

Ikoner og bilder kan lette lesningen og gjøre det enklere å navigere på nettet for personer med lesevansker. Spesielt kan det være en fordel for disse brukerne at ikonene eller bildene blir knyttet til bestemte funksjoner og meny punkter.

Bruk meningsbærende ikoner og bilder konsekvent

Ikoner og bilder letter forståelsen av teksten, men fordi de er bilder, må de også ha en alternativ tekst, for å gi mening for synshemmede. Bilder må brukes riktig, ellers kan de virke mer forstyrrende enn forklarende for lesningen. Det er derfor viktig å bruke bilder som er knyttet til historien som fortelles. Bilder skal ikke benyttes fordi de ser pene ut.

Du kan utstyre funksjoner på nettstedet med et ikon i tillegg til ledetekst. E-postadresser kan for eksempel forsynes med en konvolutt. Bruk slike ikoner konsekvent på alle sider på nettstedet. Konsekvent bruk av ikoner er særlig til stor hjelp for personer med lese- og forståelsesvansker.

2.3 Farger

Farger kan skape liv på et nettsted, men også skape problemer for noen brukere. Dels kan farger minske kontrasten mellom tekst og bakgrunn, dels kan fargeblinde ha vansker med å skille enkelte farger fra hverandre, noe som gjelder ca 10 prosent av den mannlige befolkningen. Dessuten kan det finnes det brukere som ikke benytter fargeskjerm.

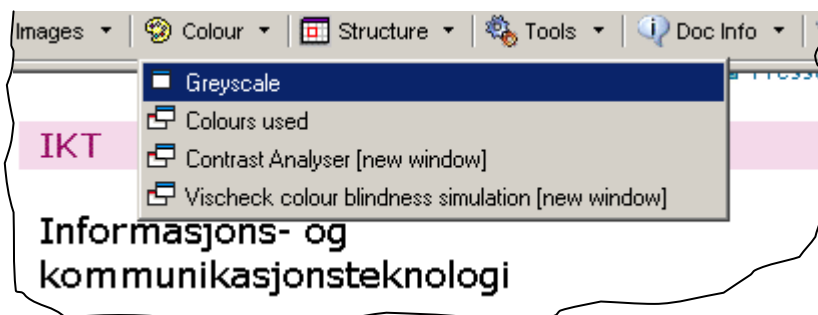
Farger kan på den annen side gjøre det enklere å orientere seg på et nettsted. Dette gjelder særlig når det samtidig er lagt til rette for at brukeren selv kan velge hvordan farger skal vises.

Bruk farger med omtanke

Kommunikasjon må ikke basere seg på fargebruk alene. Et regnskap der negative tall kun er markert som røde (uten minustegn) vil bli uleselig uten fargevisning. En grønn knapp for å starte, og en rød for å stoppe en funksjon må også ha ulik fasong, tekstes, eller på annen måte kunne skilles fra hverandre.

Sørg for at fargekombinasjonene som benyttes, har tilstrekkelig kontrast, slik at de syns for flest mulig.

Analyser fargeforskjeller og kontrast. Du kan teste fargene på flere måter. Web Accessibility Toolbar har for eksempel en funksjon som gjør det mulig å simulere gråtoner og fargeblindhet.



Colour-menyen i Web Accessibility Toolbar

I Web Accessibility Toolbar finner du også en lenke til et program som kan analysere fargeforskjeller og kontrast og som kan brukes til å sjekke om fargene du har valgt fungerer sammen.

2.4 Teksten

Tekststruktur

Ofte blir tekst laget som grafikk/bilder. Det kan enten være fordi utvikleren ønsker en bestemt skrifttype eller for å oppnå spesielle effekter med bakgrunn, roll-over eller likende.

Tekst laget som bilder er ikke tilgjengelig for mange brukere. Slik tekst gir ikke brukeren valgmuligheter. Det blir umulig å velge en skrifttype som passer ham eller henne best.

Grafikkbasert tekst kan ikke uten videre omgjøres til karakterbasert skrift og derfor ikke leses opp med syntetisk tale for synshemmede og personer med lese- og skrivevansker. Tekststørrelsen kan heller ikke endres i nettleseren, noe som er en ulempe for dem som har behov for det.

Grafikkbasert tekst blir ikke tydeligere når den forstørres. Tvert om blir den ofte mer utydelig. Det fører at svaksynte som bruker forstørrelsesprogrammer for å lese nettsider får problemer. Dersom teksten i tillegg er skrevet på en mønstret bakgrunn eller med svak kontrast til teksten, kan den bli nesten umulig å lese.

Den vanligste metoden for å sikre at grafiske elementer er tilgjengelige, er å legge inn alternativ, beskrivende tekst for bilder. Denne løsningen er først og fremst til nytte for blinde. Svaksynte vil fortsatt ha problemer dersom teksten ikke kan forstørres.

Overskrifter

Når normalt seende skal skaffe seg oversikt over en nettside, ser de etter overskriftene. Overskriftene benyttes til navigasjon. Dersom det ikke er tydelig hva som er overskrift og hva som er brødtekst, vil det være vanskelig å danne seg et klart bilde av nettsiden.

Noen brukere er avhengige av hjelpemidler for å kunne navigere etter overskriftene på siden. En metode er å liste opp overskriftene med informasjon om hvilket nivå hver enkelt befinner seg på. Skal dette være til nytte for brukeren, må overskriftene være kodet korrekt og konsekvent.

Sørg for at overskriftene er laget med korrekte heading-tagger: (eksempel: **<h1>Overskrift nivå 1 </h1>**) og at de brukes konsekvent og riktig på de forskjellige nivåene. Se mer om koding av overskrifter i kapittel 3.5.

Sørg for at det er tydelig forskjell på overskrifter og brødtekst. Dette gjelder både hovedoverskrifter og underoverskrifter. Gjør samtidig overskriftene så forskjellige at det er enkelt å skille mellom de ulike nivåene.

Husk også på å bruke korrekte tagger til å strukturere resten av teksten. Korrekte tagger er særlig viktig når du lager lister, sitater eller definisjoner.

Avsnitt og linjeskift

Det er alltid mye informasjon som skal formidles, og for designerens oppdragsgivere vil ofte det meste av informasjonen bli ansett som viktig. Dette fører til at oppdragsgiverne stiller "umulige" krav, som for eksempel at:

- brukeren skal komme fram til all informasjon på nettstedet med maks tre klikk
- startside skal ha snarveier til alle funksjoner
- alle avdelinger/produkter skal omtales på startside.

Slike krav resulterer ofte i at startside blir uoversiktlige med masser av lenker. Dersom det i tillegg legges ut reklame på siden, kan det bli virkelig vanskelig å få oversikt.

For personer som benytter opplesningsutstyr, fører uoversiktlige sider til store problemer. Å bruke skjermleser og talesyntese på et nettsted, kan til en viss grad sammenlignes med å betrakte det gjennom et sugerør. Jo mer informasjon, jo lengre tid tar det å få oversikt.

Det er problematisk for de fleste å navigere på en nettside uten å bruke mus. Tenk deg hvor lang tid det ville ta, dersom du skulle tabulere deg gjennom 100 lenker på en eneste side for å komme fram til den informasjonen du ønsker.

Innholdet på siden kan også være årsak til problemer. Dersom innholdsleverandøren ikke er vant til å skrive for nettsider, kan sidene bli fylt opp med for mye tekst og for få overskrifter. Lange sider fører til at sidene må skrolles, og det er ikke alltid like enkelt å håndtere – spesielt hvis det må skrolles både loddrett og vannrett.

Vær kritisk til bruk av annonser

Lag design uten overdreven bruk av "annonser". Det gjelder både ekte annonser, reklame for andre nettsteder, eller for andre sider på eget nettsted. Husk at nettstedets formål må være tydelig for brukeren. Samtidig er det viktig å begrense antall faste lenker på en side.

Velg en utforming med korte tekster, for eksempel ved å begrense plassen som brukes til inngresser.

Unngå at brukeren blir nødt til skrolle både loddrett og vannrett.

Skriftstørrelse

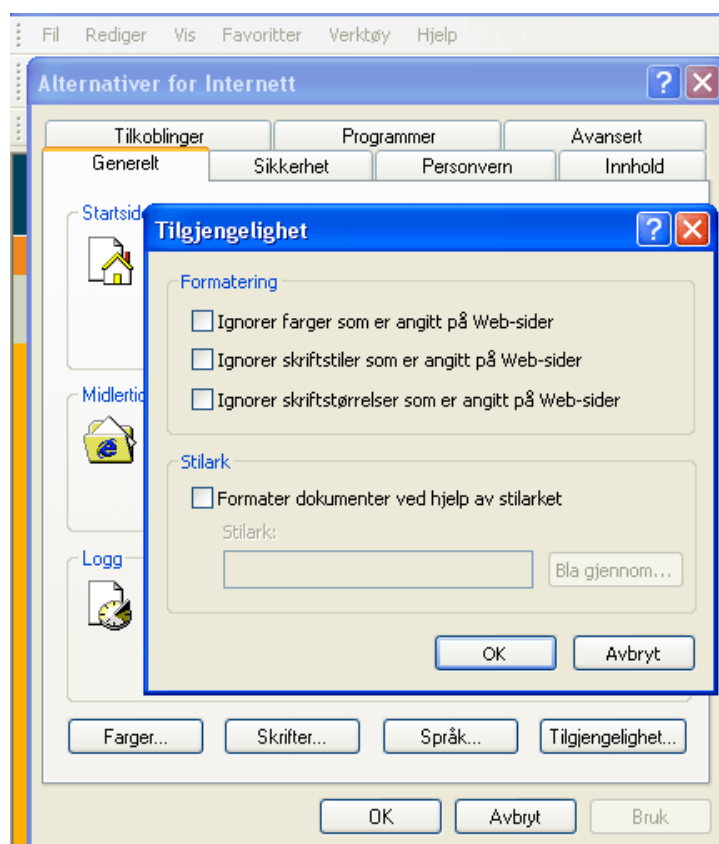
Mange har behov for stor skrift når de skal lese nettsider. De vil ofte ønske å benytte nettleserens mulighet til å vise teksten i ønsket størrelse. Internet Explorer har en slik valgmulighet i menyen **Vis > tekststørrelse**, eller ved å holde Control-tasten nede, samtidig som du bruker skrolle-hjulet på musa. Når tekststørrelsen er endret, kan det være lurt å trykke på Oppdater-knappen (F5) slik at hele siden vises med den nye innstillingen.

På denne måten skal all tekst på siden kunne skifte størrelse. Mange nettsider låser imidlertid teksten til en bestemt størrelse, og da vil i beste fall bare deler av teksten bli vist med den valgte størrelsen. Enkelte nettstedet er også designet slik at de forstørrede tekstene kuttes i margin og dermed gjør nettsiden uleselig.

Det finnes en innstilling som gjør det mulig å oppheve låste tekster. I Internet Explorer finner du den under menyen **Verktøy > Alternativer for Internett > Tilgjengelighet > Ignorerer skriftstørrelser som er angitt på Web-sider**.

Farger og skrifttyper

Noen brukere har problemer med å lese bestemte skrifttyper og/eller farger. Disse brukerne kan bruke innstillingsmuligheten som gjør det mulig å ignorere farger og skrifttyper. I Internet Explorer finnes slike valgmuligheter i menyen **Verktøy > Alternativer for Internett > Generelt > Tilgjengelighet**.



Tilgjengelighetsinnstillinger i Windows XP

Kontrast

Det er også mulig å velge en innstilling i Windows som gjør at alle elementer på skjermen vises med høy kontrast. Dette er særlig nyttig for svaksynte fordi skriften blir tydeligere. I Windows kan slik innstilling gjøres via: **Kontrollpanel >**

Tilgjengelighetsalternativer > Skjerm > Høykontrast. Ved å krysse av i feltet "Bruk høykontrast" vil Windows skifte til en visning med bedre kontrast. Gå deretter tilbake til Internet Explorer og oppdater (reload) siden for å se resultatet. I noen tilfeller vil det kunne virke som om lenkene ikke har fått optimal kontrast, men det kan forbedres ved å tilpasse fargene som brukes på lenkene.

Bruk HTML og velg skrifttyper som er tilgjengelig på mange plattformer og nettlesere. På den måten vil de fleste få det inntrykket av siden som du selv ønsker. Samtidig gir du brukeren muligheten til selv å velge den skrifttypen som passer best.

Vurder nøye bruken av bakgrunnsfarge og -mønster. Unngå alle former for bakgrunn som ikke øker lesbarheten av den teksten du skal publisere.

Angi den ønskede skrifttypen ved hjelp av CSS. Angi dessuten alternativer med en generisk mulighet som siste valg. Det vil sikre at nettleseren finner en skrifttype som ligner mest mulig på den du er ute etter, selv om de første valgene ikke finnes på den lokale PC-en.

```
<style>
.avsnitt {
    font-family: Arial, Helvetica, Verdana, sans-serif;
    color: green;
    font-size: 1.5em;
}
</style>
```

CSS -koden over angir at dersom skrifttypene Arial, Helvetica og Verdana ikke er tilgjengelige, skal nettleseren benytte en tilgjengelig skrift av typen Sans-serif. Sans-serif betyr "uten seriffer". Den generiske angivelsen kommer til slutt. Teksten skal i alle tilfeller være grønn og ha en størrelse på 1,5em.

Seriffer

- er de tversgående avslutningene av strekene noen bokstavtyper har.

Eksempel:

"med seriffer" og "uten seriffer".

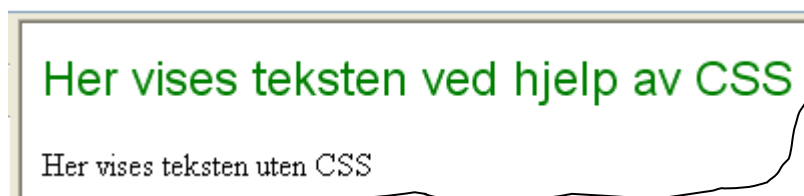
em

- brukes til å angi skalerbare fonter med størrelse relativt til hverandre. 1em angir en font i standard størrelse. 1.5em angir en 50% større font enn 1em. 0.8em betyr 80% av 1em – fonten. Fontene beholder sine innbyrdes størrelsesforhold om de skaleres.

I HTML-dokumentet kan CSS-koden i eksempelet over brukes på følgende måte:

```
<p class="avsnitt"> Her vises teksten ved hjelp av CSS</p>
<p>Her vises teksten uten CSS</p>
```

Den første linja bruker klassen avsnitt, som er spesifisert i CSSkoden. Den andre linja vises uten bruk av CSS



Koden slik den vises i en nettleser

2.5 Navigering

Lenker

Lenker er et basiselement på de fleste nettsider og skal kunne brukes av alle. Det innebærer at de må være lette å se og å treffe med markøren og at det er innlysende *hvor* lenken vil sende brukeren.

I "gamle dager" framsto lenkene med blåfarge og tydelig understreking. Dermed var en aldri i tvil hvorvidt et tekstelement var en lenke eller ikke. I dag vises lenker også på mange andre måter, og selv om de fleste klarer å finne ut av det, kan problemer oppstå når ei lenke ikke blir synlig før markøren beveges over den. Uerfarne brukere, personer med nedsatt syn og brukere som navigerer på nettstedet ved hjelp av tastatur, vil trenge godt synlige lenker. Dessuten er det ingen ulempe for andre heller!

Brukere som navigerer ved hjelp av tastatur, bruker tabulatortasten. Dersom det ikke er tydelig hvilken lenke som markeres, er det for disse brukerne nesten umulig å vite hvor på siden de befinner seg.

En ny trend er lenker i form av små trekanter eller plusstegn. Slike lenker kan være vanskelige å treffe for personer som skjelver på hendene og som har problemer med å styre en mus. Det samme gjelder når lenker er så tett plassert at det er lett å komme bort i nabolenken.

Lenketekster som "Les mer" eller "Klikk her" et problem for blinde og andre som bruker skjermlesere. Slike tekster uten eget innhold, gjør det umulig å skille lenker fra hverandre ved opplisting og det er også umulig å vite hvor lenken vil føre brukeren. Det finnes flere måter å takle dette på og du kan lese mer på Norge.no:

http://nettverk.norge.no/index.php?option=com_content&task=view&id=27

Noen råd om lenker:

- Lag designet slik at det er tydelig hva som er lenker og hva som er vanlig tekst. Uthevet tekst på websider bør derfor være i kursiv eller fet skrift framfor understreket. Det er også en god ide å sørge for å synliggjøre den lenka som er markert. Det gjelder enten man bruker mus eller tastatur.
- Gjør lenkeområdene store. Det holder ikke med et lite plusstegn eller en trekant. Lenken må omfatte et større tekstlig element.
- Sørg for at lenkene plasseres med god avstand mellom hver lenke.
- Lag meningsbærende lenketekster, slik at de kan forstås uten den omliggende teksten. Unngå å bruke samme lenketekst flere steder på samme side.
- Du kan bruke CSS til å merke lenker når de blir markert ved hjelp av tastaturet eller når markøren peker på lenkene. Bruk en av "pseudo-klassene" til dette: a:link, a:visited, a:hover, a:active, a:focus

Hjelp brukeren å navigere

De fleste har opplevd å miste oversikten over hvor de befinner seg på et nettsted. Særlig på større nettsteder med mange underliggende sider kan det være vanskelig å orientere seg.

Brukere har ofte problemer med å forstå hvordan nettstedet er strukturert, hva det inneholder og hvordan de har havnet på den aktuelle nettsiden. Problemet er spesielt stort for brukere som i utgangspunktet har problemer med å forstå hvordan nettsteder normalt er strukturert. Dette kan gjelde blinde (synshemmede), personer med orienterings- eller forståelsesvansker og uerfarne nettbrukere.

Sørg for å tydeliggjøre hvor på nettstedet brukeren befinner seg. Det må framgå hvilken side en er på, hvilket nivå som ligger over der en er, og hvilke andre sider som hører sammen med den siden en er på.

En mulighet er å lage en "nettsti" øverst på siden. En nettsti er en liste som viser hvordan du som bruker har kommet fram til den aktuelle nettsiden. Elementene i nettstien bør lages som lenker for å gjøre det mulig å gå til nivåene over.

Hjem > Bygg og uterom > Offentlige bygg > Artikkel
Kronikk:

Diskriminerende praksis fortsatt hverdag for funksjonshemmede.

På de fleste samfunnsområder er det fortsatt stor avstand mellom politiske mål om full deltakelse for alle og virkeligheten.

Eksempel på nettsti

Du kan også velge å vise siden som uthevet i menyen. Da bør du også sørge for at menypunktet ikke er klikkbart som lenke, fordi det kan være forvirrende at en side lenker til seg selv.



Du bør også gi hver enkelt side en meningsbærende tittel. Dette er en fordel for mange, særlig for dem som bruker hjelpemidler til opplesningen av teksten. De vil da få lest opp tittel på den siden de går inn på:

<title>Denne sidens tittel</title>

Søkemotorer bruker også ofte tittelfeltet når de søker på nettet og i tillegg bruker de fleste nettlesere innholdet i tittelfeltet som forslag til navn når noen vil lage et bokmerke til nettsiden.

Meny med valget "Internett" uthevet.
Her brukes også en "pil ned" og innrykk for å lette forståelsen

Unngå lenker som åpner nye vinduer

Åpning av sider i nye vinduer brukes ofte når det lenkes fra ett nettsted til et annet. Teknikken benyttes for å holde brukeren på nettstedet, men kan virke forvirrende. Når det åpnes ett nytt vindu, har ikke det nye vinduet noen "historie" og brukeren kan derfor ikke bla tilbake til tidligere viste sider.

Unngå så langt det er mulig å åpne sider i nye vinduer. Er det nødvendig, bør brukeren få informasjon på forhånd om at det åpnes et nytt vindu. Slik informasjon kan legges i lenketeksten eller liknende.

Eksempel: [Les mer om kvalitet på nett på norge.no \(nytt vindu\)](#)

En variant av nye vinduer er "Pop-up"-vinduer som brukes mye til reklame eller spørreskjema. Slike vinduer viser ofte ikke de verktøylinjene som det vanlige nettleservinduet har. De mangler ofte mulighet for tilpasning til skjermen og kan også være vanskelige å lukke, særlig for brukere som ikke bruker mus. Bruk derfor "pop-up"-vinduer så lite som mulig.

Et annet argument mot "pop-up" vinduer er at stadig flere bruker stiller inn nettleseren til å blokkere "pop-up" vinduer. Disse brukerne får dermed ikke den informasjonen som er plassert i "Pop-up"-vinduer.

"Pop-up"-vinduer kan brukes der det er viktig for brukeren å kunne lese i to vinduer samtidig, for eksempel i forbindelse med lesning av "Hjelp"-tekster. Bruker du "Pop-up"-vinduer må du imidlertid lage det slik at vinduets størrelse kan tilpasses av brukeren og at det kan lukkes.

2.6 Testing

Når du som designer leverer det ferdigstilte materialet til utvikleren/programmereren, kan vedkommende anta at all nødvendig tilgjengelighet er innebygd i designet. Dermed jobber han ut fra den forutsetningen og kontrollerer ikke tilgjengeligheten før løsningen er ferdigstilt. Slik kan dere ende opp med å levere en helt utilgjengelig løsning.

Sjekk tilgjengeligheten selv

Gå igjennom designet nøye med hensyn til tilgjengelighet. Sjekk at du har fått med alle punktene i dette heftet i forhold til valg med betydning for tilgjengeligheten. Informer alle andre prosjektdeltakere og kunden om valg som er gjort underveis.

Leverer du en HTML-mal og ett eller flere stilark, bør de kontrolleres med et valideringsverktøy. I "Web Accessibility Toolbar" vil du finne snarveier til flere nyttige verktøy. Les mer i kapittel 4: "Nyttige verktøy og referanser".

Dessverre finnes det ingen automatiserte verktøy som kan teste om et nettsted kan sies å være fullt ut tilgjengelig. Det er for eksempel ikke mulig å teste automatisk om en "alt-tekst" er meningsbærende. Slike former for testing må utføres av personer som har kompetanse på området. Vær også oppmerksom på at testingen bør dekke hver enkelt side på et nettsted, ikke bare startsiden. Fortsatt er det slik at ingen av de testverktøyene som er gratis, tester hele nettstedet.

3 Koding – råd til programmerere

Vi skal her gjennomgå 12 praktiske råd og tips beregnet på deg som programmerer nettsteder. Dette er råd som kan bidra til å forebygge noen av de vanligste problemene knyttet til tilgjengelighet.

Listen rommer eksempler på hvordan en forholdsvis liten innsats kan gi gode resultater og gjøre nettstedet brukbart for flere. Hvert råd er beskrevet med eksempler, løsninger og praktiske kodeeksempler. Vær oppmerksom på at selv om du følger disse rådene, gir det ingen absolutt garanti for at nettstedet blir fullt ut tilgjengelig.

Som programmerer kan du på flere måter gjøre en viktig innsats for å sikre tilgjengelighet:

- Du kan sikre at all kode blir skrevet korrekt og i overensstemmelse med tilgjengelighetskravene.
- Du kan bidra til at nettstedet overholder tilgjengelighetskravene.
- Du kan sørge for at det publiseringsverktøyet som nettstedet bruker, gjør arbeidet med tilgjengelighet enklere.

Den seinere tida har det vært fokusert mye på å gjøre nettsteder som sluttprodukt tilgjengelige. Nå kommer også krav til redaktørens publiseringsverktøy (CMSene). Det viser seg at mange publiseringsverktøy systematisk produserer nettsteder med dårlig tilgjengelighet, ved at de for eksempel låser tekststørrelsen, benytter standard lenketekster og så videre. Krav til slike systemer er allerede gjort gjeldende i USA. I tillegg fokuseres det mye på korrekt HTML, og i Sverige tester de jevnlig HTML på over 1000 nettsteder. Mer informasjon om dette finner du på:

www.verva.se/web/t/Page_2135.aspx .

EIAO

Det EU-støttede prosjektet EIAO (European Internet Accessibility Observatory) ledes av Høgskolen i Agder og har som målsetting å foreta månedlige automatiske evalueringer av tilgjengeligheten på 10.000 europeiske nettsteder.

Hensikten er blant annet å gi tilbakemelding om manglende tilgjengelighet og å følge utviklingen i nettstedenes tilgjengelighet over tid.

Les mer om EIAO på: www.eiao.net/

W3C konsortiet arbeider på samme måte som med WCAG (Web Content Accessibility Guidelines – (WAI)) også med ATAG (Authoring Tool Accessibility Guidelines). ATAG retter seg mot utviklere av publiseringsverktøy for å gjøre disse

verktøyene bedre med hensyn til å produsere mer tilgjengelige nettsteder. Se: <http://www.w3.org/WAI/intro/ataq.php>

Gjør så godt du kan

Det kan være vanskelig å implementere alle retningslinjene. Det er viktig å hele tiden vurdere systematisk hvilke krav som settes "på vent" og hvilke konsekvenser dette vil få for brukerne. Praktiske tester og ekspertvurderinger vil ofte være til god hjelp. Husk at det alltid er bedre å ta noen skritt i riktig retning enn å stoppe helt opp, selv om det innebærer at ikke alle kravene i retningslinjene innfris umiddelbart.

3.1. Velg og deklarerer HTML-standard

Det er mange nettsteder som ikke angir hvilken HTML-spesifikasjon som er lagt til grunn for nettstedet. I og med at de fleste nettleserne er ganske tolerante overfor feil, opplever ikke de fleste brukerne dette som noe problem. Brukere som trenger spesielle løsninger og alternative hjelpemidler vil derimot kunne møte problemer når HTML-koden ikke følger en angitt standard.

Velg standard

Velg en standard som passer til den aktuelle oppgaven og deklarer den i alle dokumenter. Test om koden overholder standarden. Dette kan gjøres enten med W3Cs HTML-validator eller med et annet verktøy. HTML-standardene er spesifisert på: www.w3.org/MarkUp/. W3C's online HTML validator finner du på: <http://validator.w3.org/>

HTML 4.01 Strict:

HTML 4.01 Strict omfatter bare elementer og attributer som er anbefalte og som ikke opptrer i forbindelse med framesets.

```
<!DOCTYPE HTML PUBLIC  
"-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">
```

HTML 4.01 Transitional:

HTML 4.01 Transitional omfatter alt som fines i strict DTD men inneholder i tillegg en del elementer og attributter for presentasjon, som ikke er ønsket (deprecated) i HTML 4.01 strict.

```
<!DOCTYPE HTML PUBLIC"-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

XHTML

XHTML er en nyere variant av HTML, som er utviklet for å bringe HTML nærmere den faste strukturen i XML og ytterligere skille innhold fra presentasjon.

Dersom du:

- skriver gyldig HTML 4.01 kode
 - bruker små bokstaver (minuskler) i tagger og attributter
 - benytter anførselstegn rundt alle attributter
 - konsekvent bruker avslutningstagger
</....>
 - bytter ut "name"-attributtet med "id"
- er du svært nær gyldig XHTML-kode.

XML

XML er et språk laget for å beskrive data. XML har ikke forhåndsdefinerte tagger og krever at alle tagger defineres i en "document type definition" (DTD). Alle tagger skal skrives med små bokstaver (minuskler) og skal alltid benyttes parvis, med en åpningstagg og en avslutningstagg.

XHTML 1.0 Strict

Inneholder ingen layout-elementer og krever en strikt kodeanvendelse med lukking av alle tagger. Layouten tas vare på med Cascading Stylesheets (CSS). Vær oppmerksom på at det kan finnes eldre nettlesere, (før ca. 2001) som ikke forstår CSS. Disse nettleserne vil se HTML-dokumentet uformattert. For eksempel kan ikke <frame>-elementet brukes i XHTML strict.

```
<!DOCTYPE html  
PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

XHTML 1.0 Transitional

Kan benyttes når du ønsker å benytte presentasjonselementene i HTML, eller ønsker å støtte nettlesere som ikke forstår CSS

```
<!DOCTYPE html  
PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

Om du avviker fra valgt HTML-standard, bør du likevel validere nettsidene slik at du kan legge deg så tett opp til standarden som mulig. Du bør samtidig undersøke hvilke konsekvenser dette får for brukerne. Korrekt HTML er en fordel for alle brukere.

Slik gjør du det

HTML_dokumenttypen deklarereres aller først i HTML-dokumentet og settes inn foran <html> start-taggen:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

Eksempelet over viser til dokumenttypen XHTML 1.0 strict. I tillegg til at det angis hvilken standard som er brukt lenkes det til en fil som viser hvilke elementer standarden omfatter.

Det beste alternativet for nye nettsteder, vil være XHTML strict med bruk av CSS til presentasjon. Dersom dette ikke er mulig, for eksempel fordi du er bundet opp til en eldre løsning, anbefales det å bruke en så strukturert standard som mulig, enten XHTML transitional, HTML 4.01 strict eller HTML 4.02 transitional.

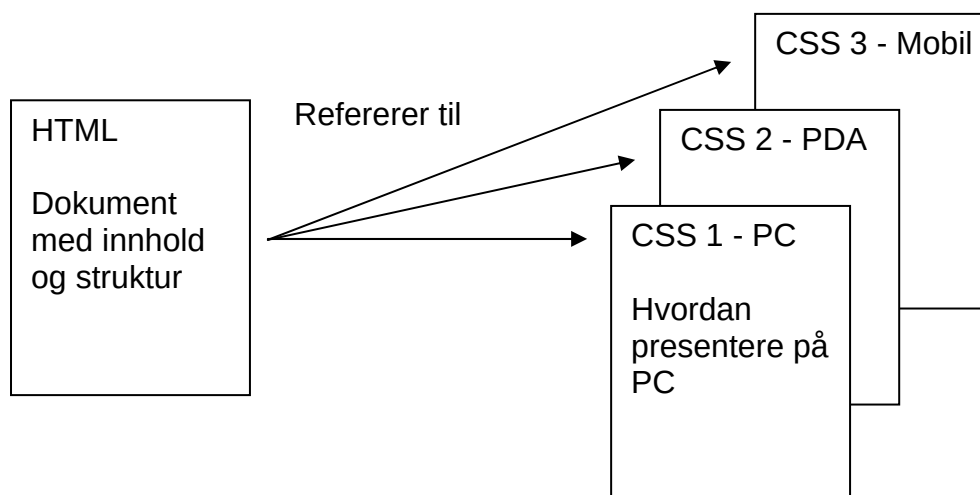
En oversikt over hvilke elementer som er lovlige i de ulike standardene finner du på <http://www.w3schools.com/tags/default.asp>

3.2. Bruk CSS og overhold standarden

På mange nettsteder er layouten laget ved hjelp av tabeller, font-tagger og andre formateringskoder som er spredt ut over hele HTML-dokumentet. Slik koding gjør det vanskeligere å tilpasse layouten til alternative visninger. Koden kan også gjøre innholdet uforståelig i situasjoner der skjermleseren fjerner de aktuelle tabellene.

Bruk CSS (Cascading Stylesheets)

CSS skiller form og innhold. CSS gjør det mulig å definere ulike visninger tilpasset forskjellige medier, med forskjellige skrift- og bakgrunnsfarger, med eller uten bilder og mye mer. Ved å gjøre CSS gjør du designet fleksibelt og funksjonaliteten bedre. Det gjør det mulig å presentere det samme HTML-dokumentet på mange ulike plattformer, som PDA-er og mobiltelefoner. Formatene kan lages med enten interne eller eksterne stilark. Tekststørrelser og plassering av elementer, skal angis ved hjelp av relative enheter som for eksempel "%" eller "em".



Mulighetene med CSS er mange. Det er også viktig å kontrollere at nettsiden lar seg lese også når den vises uten bruk av stilark. I eksempelet nedenfor viser vi hvordan stilark kan brukes til å fjerne uønsket informasjon ved utskrift av websider.

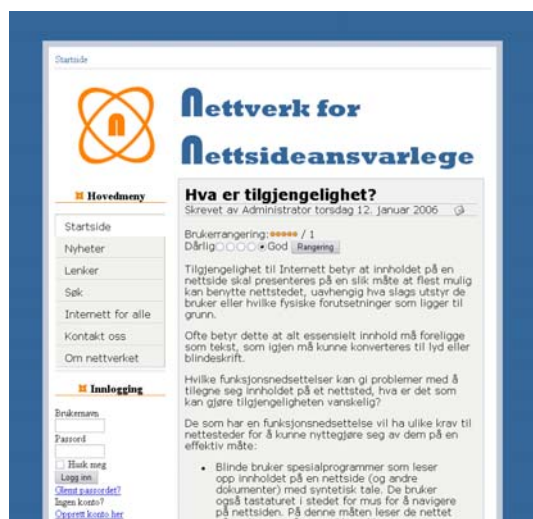
Slik gjør du det

Lag lenke til eksternt stilark, eventuelt til flere ulike stilark for å lette visning i flere medier:

```
<link href="printstylesheet.css" type="text/css" rel="stylesheet"
media="print"/>
```

```
<link href="screenstylesheet.css" type="text/css" rel="stylesheet"
media="screen"/>
```

Koden over viser hvordan CSS-filene "printstylesheet.css" "screenstylesheet.css" benyttes for presentasjon på skjerm og i utskrift.



Skjermdump av samme nettside med ulike stylesheets for skjerm og utskrift.

De mest vanlige medievariantene er:

- print – for bruk på skivere
- screen – for bruk på dataskjermer
- projection – for bruk på videokonon
- handheld – for bruk på mobiltelefon og PDA
- all – for alle medier

Du finner mer informasjon om mediatyper på <http://www.w3.org/TR/REC-CSS2/media.html>

3.3. Angi språk

Dersom et HTML-dokument publiseres uten å angi hvilket språk innholdet er skrevet i, vil ikke skjermleserprogrammer vite hvilken syntese som skal bruke til opplesing. Et dokument skrevet på norsk, men lest opp på engelsk vil da bli helt uforståelig.

Språkkoder

Språk angis i overensstemmelse med betegnelsene angitt i ISO639-2.

Oversikt over språkkoder finnes på: www.loc.gov/standards/iso639-2/langcodes.html

En ytterligere fordel ved å angi språk er at det gjør det enklere for søkemotorer å finne fram til relevante sider.

Slik gjør du det

Et dokumentets hovedspråk angis i start-taggen: **`<html lang="no">`**

Dersom det brukes flere språk i dokumentet, kan avsnitt med andre språk kodes ved hjelp av "span"-taggen i teksten, slik:

Dette er norsk tekst `<span lang="en">`This is english.`</span>` her er det norsk igjen.

Dokumentets hovedspråk kan også angis ved hjelp av Dublin Core standarden for metadata ved å legg inn en "meta"-tag i "head"-elementet, slik:

`<meta name="DC.language" schme="ISO 639-2/B" content="no">`

3.4. Gi alle sider en tittel

En side uten tittel eller et nettsted der alle sider har samme tittel, gjør det vanskeligere å navigere mellom flere vinduer.

Meningsbærende titler

Sørg for at alle sider får sin unike og meningsbærende tittel. Slike titler gjør navigasjonen enklere for alle brukere. Tittelen foreslås også som navn når du legger bokmerke til sida i "favoritter"

Slik gjør du det

Tittel legges inn i "title"-taggen i "head": **`<title>Min sidetittel</title>`**

3.5. Kod overskrifter korrekt

Blinde brukere som er avhengige av skjermleserprogrammer, navigerer ofte via overskrifter og lenker fordi det tar lang tid å få lest opp en hel side. Dersom nettsiden er formatert ved hjelp av "font"-tagger eller CSS, uten å identifisere hva som er overskrifter, blir brukeren tvunget til å lese hele siden for å få oversikt. Velstrukturerte, korrekt kodede dokumenter er enklere å navigere i både for blinde, og alle andre og det kan forbedre søkemotorers treff på sidene.

Hierarkisk sammenhengende overskrifter

Bruk HTML-strukturelementene til det de er ment til og ikke ut i fra hvordan de vises. Ønsker om forskjellige visninger tilpasser du ved hjelp av CSS. Overskrifter skal alltid være hierarkisk sammenhengende.

Kontroller også at nettreddaktøren koder overskrifter på en korrekt måte.

Slik gjør du det

Regelen er at nedover i strukturen går man ett nivå om gangen. Ellers kan en holde nivået, eller gå oppover ett eller flere nivåer.

Et kapittel i en rapport kan for eksempel bygges opp slik:

```
<h1>Overskrift nivå 1</h1>
  <h2>Overskrift nivå 2</h2>
    <h3> Overskrift nivå 3</h3>
    <h3> Overskrift nivå 3</h3>
  <h2> Overskrift nivå 2</h2>
    <h3> Overskrift nivå 3</h3>
<h1>Overskrift nivå 1</h1>
  <h2>Overskrift nivå 2</h2>
    <h3> Overskrift nivå 3</h3>
    <h3> Overskrift nivå 3</h3>
```

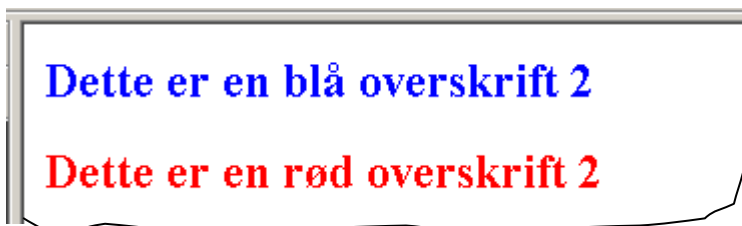
Dersom det er ønskelig å vise et element på ulike måter, kan forskjellige "class"-typer brukes på elementet.

```
h2 { color:"blue"}
h2.red {color:"red"}
```

I CSS-koden over, settes standard h2 overskrift til blå, mens h2 overskrifter av klassen "red" settes til å være rød.

I HTML-dokumentet kan bruk av overskriftene se slik ut:

```
<h2>Dette er en blå overskrift 2</h2>
<h2 class="red">Dette er en rød overskrift 2</h2>
```



Overskriftene slik de vises i nettleseren.

3.6. Unngå identiske lenkebetegnelser

Brukere som trenger skjermleser for å navigere på sidene, velger ofte å starte med å få lest opp lenkene på en side for å få oversikt over innholdet. Dersom flere lenker har lik lenketekst, som for eksempel de mye benyttede "Les mer" eller "Klikk her", blir ikke brukeren så mye klokere. Slik er det også dersom flere lenker peker til det samme dokumentet.

Overskriften som lenke

Framfor å bruke lenketekster av typen "Les mer" eller "Klikk her", er det en mulighet å bruke overskrifter og / eller ingresser som lenker.

3.7. Skriv meningsbærende alt-tekster

Bilder, lyd video og andre innbakte elementer kan være utilgjengelige for noen brukere. For eksempel kan ikke blinde nyttiggjøre seg bilder og helt døve kan ikke høre lydopptak.

Alternativ tekst

Beskriv alle ikke-tekstlige elementer ved hjelp av "alt"-attributtet. Dersom et element ikke har innholdsmessig betydning, kan "alt"-attributtet angis uten innhold(`alt=""`). Som programmerer er det din jobb å sørge for at nettrektøren blir minnet på at det skal lages en alternativ tekst hver gang et ikke-tekstlig element legges inn på nettstedet.

Ved bruk av komplekse og innholdsrike elementer kan det være praktisk å legge inn en lengre beskrivelse av hvert element i et eget HTML-dokument ved hjelp av "longdesc"-attributtet.

Slik gjør du det

Eksempel på alternativ tekst til bilde:

```

```

Eksempel på en lengre beskrivelse med "longdesc":

```

```

Longdesc gjør i prinsippet det samme som om du legger en lenke til et annet dokument. Det kan være en saksak hvilken metode som brukes.

3.8. Bruk datatabeller korrekt

Tabeller er beregnet til presentasjon av data og må skrives på en måte som gjør det mulig å skille mellom tabelloverskrifter og data for ikke å skape problemer for blinde brukere.

Noen nettsteder benytter fortsatt en løsning der tabeller brukes til kontroll av layout. Slike løsninger bør avlases av CSS-basert layout. Tabeller brukt til kontroll av layout kan være svært forvirrende for brukere av skjermlesere fordi innholdet kan bli lest opp i en ulogisk rekkefølge.

Gjør tabeller forståelige

Med helt enkle tabeller der øverste rad og/eller første kolonne er overskriftsfelt, er det ganske enkelt å få overskriftene markert som overskrifter. Snakker vi om mer avanserte tabeller, med overskrifter som strekker seg over flere kolonner, kan det være mer komplisert å få gjort tabellen forståelig for skjermleserprogrammene. Det finnes imidlertid spesialverktøy som kan lage helt korrekte tabeller.

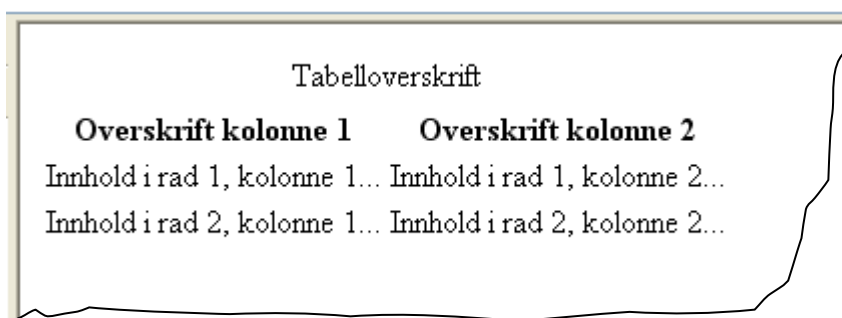
Bruker du fortsatt tabeller til layout, må du kontrollere at innholdet blir meningsbærende når det vises lineært. Det finnes verktøy som kan vise slike lineære versjoner av en tabell. Sjekk med "Web Accessibility toolbar".

Slik gjør du det

Tabeller har en rekke attributter som blir oversett: "Summary" er et attributt som leses av skjermlesere, "scope"-attributtet angir hvilket innhold en overskrift viser til. For å unngå opplesning av hele overskriften hver gang den enkelte verdien i tabellen leses, er det mulig å angi en forkortelse for en overskrift ved hjelp av attributtet "abbr".

En tilgjengelig tabell kan for eksempel kodes på følgende måte:

```
<table summary="Sammendrag for skjermlesere: tabellen viser .....">
  <caption>Tabelloverskrift </caption>
  <thead>
    <tr>
      <th scope="col" abbr="Kol.1"> Overskrift kolonne 1</th>
      <th scope="col" abbr="Kol.2"> Overskrift kolonne 2</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Innhold i rad1, kolonne 1... </td>
      <td>Innhold i rad1, kolonne 2... </td>
    </tr>
    <tr>
      <td>Innhold i rad2, kolonne 1... </td>
      <td>Innhold i rad2, kolonne 2... </td>
    </tr>
  </tbody>
</table>
```



Overskrift kolonne 1	Overskrift kolonne 2
Innhold i rad 1, kolonne 1...	Innhold i rad 1, kolonne 2...
Innhold i rad 2, kolonne 1...	Innhold i rad 2, kolonne 2...

Koden ovenfor slik den vises i nettleseren.

3.9. Bruk label-elementet på skjemaer

Utfylling av skjema er en vanlig oppgave på mange nettsteder. Slike skjema kan være vanskelige å bruke for personer med nedsatt syn eller blinde. På mange nettsteder er det plassert tekst ved siden av skjemafeltene, ofte i en tabell som skal

holde dem på plass. Når et skjermleserprogram skal lese skjemaet, oppstår det fort forvirring om sammenhengen mellom feltene i skjemaet og ledeteksten, dersom koderen ikke har tatt tilstrekkelig hensyn til dette.

Label-element

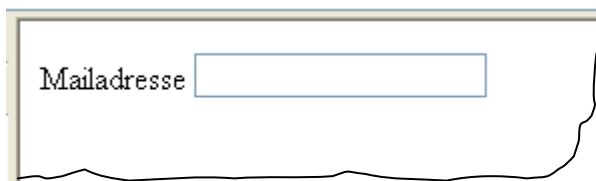
Løsningen er å knytte et label-element til hvert felt i skjemaet. Merk at det ikke er nødvendig å plassere den aktuelle teksten direkte i feltet.

Slik gjør du det

For å knytte et label-element til et felt i et skjema, må feltet ha et "id"-attributt som labelen kan referere til.

```
<Label for="adresse">Mailadresse</label>  
<input type="text" name="adresse" id="adresse">
```

I eksempelet over knyttes labelen "Mailadresse" til inputfeltet "adresse": "name" er verdien som blir sendt til mottakeren av skjemaet, og generelt anbefales det at "id" og "name" er like. I vårt tilfelle kalles begge "adresse".



Koden over slik den vises i nettleseren:

3.10. Bruk Javascript med omtanke

Javascript brukes på mange nettsteder enten til menyer eller for å validere utfylling av skjema. Javascript kan imidlertid føre til problemer:

- Det kan være at javascriptet ikke virker fordi brukeren enten har slått av javascript-muligheten i nettleserens innstillinger, eller fordi javascript ikke kan leses av den aktuelle nettleseren.
- Javascriptet kan være skrevet på en måte som gjør at funksjonen kun reagerer på mus, og altså ikke er tilgjengelig fra tastaturet.

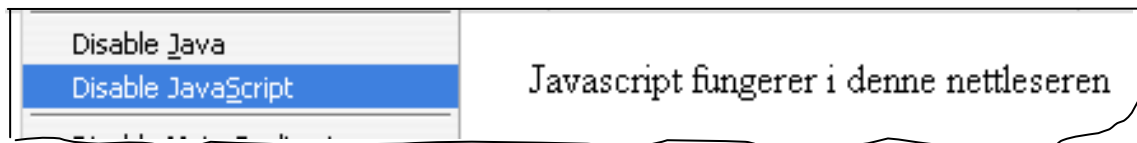
Hvordan bruke javascript

Sørg for at nettstedet kan fungere uten javascript. Bruk javascript til å gjøre ting enklere, men ikke gjør nettsiden ubrukelig ved å forutsette at javascript virker. Dersom javascript faktisk er påkrevd for å kunne ta i bruk nettstedet, må dette varsles på en tydelig måte. Sørg for at alle de rutinene som reagerer på en mus også kan fungere via tastaturet.

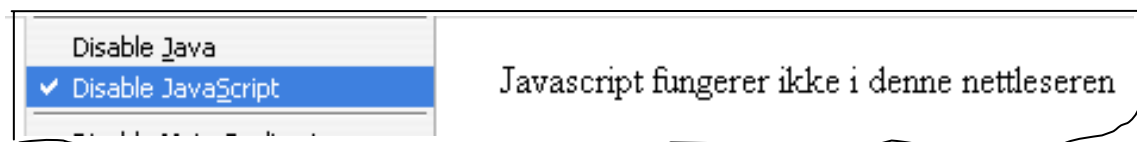
Slik gjør du det

Bruk "noscript"-taggen til å beskrive alternativ funksjonalitet. Koden nedenfor skriver: " Javascript fungerer i denne nettleseren" dersom den kan kjøre javascript. Alternativt skriver den: "Javascript fungerer ikke i denne nettleseren".

```
<script type="text/javascript">
document.write("<p>Javascript fungerer i denne nettleseren</p>");
</script>
<noscript>
<p>Javascript fungerer ikke i denne nettleseren</p>
</noscript>
```



Med javascript



Uten javascript

Sørg for at de forskjellige scriptene reagerer både på mus og tastatur. Dette gjøres ved at events/hendelser knyttes til både tastatur og mus. Følgende hendelser bør benyttes sammen for å sikre god funksjonalitet:

"onmouseover" og "onfocus"
"onclick" og "onkeypress"

Eksempel på bruk av mus- og tastatur-events:

```
<div id="tekst" tabindex="1"
onmouseover="alert('Markøren pekte på overskriften!')"
onkeypress="alert('Du trykket en tast mens overskriften var
markert!')">
<h1>En overskrift</h1>
</div>
```



Slik vil nettleseren reagere når markøren peker på overskriften

3.11. Bruk plug-ins med omtanke

Plug-ins, java-applets og likende kan føre til problemer for personer som benytter hjelpemidler. Det kan være slik at plug-ins programmene ikke fungerer sammen med den aktuelle nettleseren, at programmene ikke reagerer på tastaturet, eller at innholdet ikke lar seg lese ved hjelp av hjelpeprogrammer som for eksempel skjermlesere.

Plug-ins og alternative løsninger

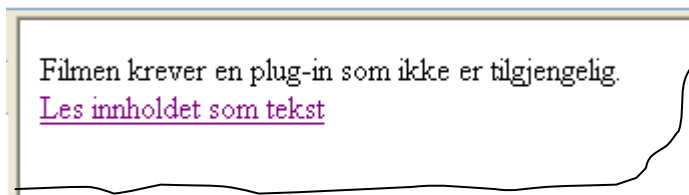
Den beste vil være å sørge for at det aktuelle plug-in programmet fungerer sammen med både nettleser, inputenheter og hjelpeprogrammer. Dette er det imidlertid umulig å gi noen garanti for. Sørg derfor alltid for å tilrettelegge alternative løsninger dersom du velger å bruke plug-ins. Brukere av kompenserende hjelpemidler har tidligere hatt store problemer med Flash. Fra og med versjon 6 har imidlertid mulighetene blitt bedre, men tilgjengelige løsninger med Flash krever stor omtanke og betydelig innsats fra programmereren.

Slik gjør du det

Plasser beskrivelsen av innholdet i en "object"-tagg eller sett inn en lenke som peker på en alternativ presentasjon av objektets innhold. På den måten kan brukere som ikke har, eller kan benytte den nødvendige plug-in, få opp en alternativ tekst om hva plug-in programmet gjør.

```
<object  
  type="application/x-shockwave-flash"  
  data="movie.swf"  
  width="550" height="400">  
  Filmen krever en plug-in som ikke er tilgjengelig.<br />  
  <a href="tekstbeskrivelse.htm">Les innholdet som tekst</a>  
</object>
```

I eksempelet over spilles animasjonen "movie.swf" dersom plug-in er til stede. Andre får beskjed om at plug-in mangler og lenke til en alternativ tekstlig beskrivelse.



Resultatet i nettleser som ikke støtter plug-ins

3.12. Sørg for tilgjengelighet hele veien

Designeren har sørget for tilgjengelig design, programmereren har skrevet tilgjengelig kode, men har webredaktørene husket på tilgjengelighetskravene? Dessverre er det ikke alltid slik. Det er ikke nødvendigvis mangel på kunnskap om en skal beskrive bilder eller gi lenker navn som er problemet. Utfordringen er like mye at en webredaktør glemmer å tenke på tilgjengelighet i en stresset arbeidssituasjon. Selv om den tekniske tilgjengeligheten er i orden, kan innholdet være vanskelig å forstå for brukerne. Innholdet kan være for komplisert, teksten kan være for lang, eller det kan være for mange lenker på nettsiden. Slike forhold skaper problemer for uerfarne webbrukere, for dem som bruker skjermlesere, personer med lese- og skrivevansker eller med nedsatt kognisjon.

Kontroller den tekniske tilgjengeligheten

Sørg for å kontrollere den tekniske tilgjengeligheten. Kontroller at det blir lagt inn relevant "alt-tekst" hver gang det legges inn et nytt ikke-tekstlig element. Gjør det enkelt for webredaktøren å lagre bilde-beskrivelser sammen med selve bildet, slik at teksten følger med dersom bildet brukes flere ganger

Slik gjør du det

Følg de retningslinjene som er fastlagt for nettstedet med hensyn til sidestørrelse, antall meny punkter og lenker.

Følg gjeldende retningslinjer for tilgjengelige publiseringsverktøy, ATAG, Se: <http://www.w3.org/WAI/intro/ataq.php>

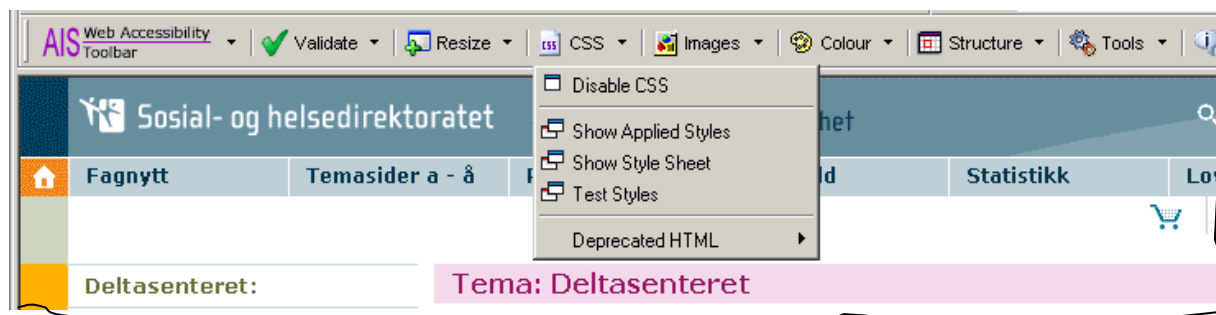
4. Nyttige verktøy og referanser

4.1 Tilgjengelighetsverktøy

Web Accessibility Toolbar

Accessibility Toolbar(AIS) er et meget nyttig verktøy som du kan laste ned og legge som en ekstra verktøylinje på Internet Explorer. AIS har en rekke nyttige funksjoner som kan brukes for å utvikle, kontrollere og teste nettsteders tilgjengelighet.

Accessibility Toolbar er utviklet av National Information Library i Australia og kan lastes ned fra: www.visionaustralia.org.au/ais/toolbar/



Bilde av Web Accessibility Toolbar

Web Developer Extension

Web Developer Extension er et verktøy som tilsvarer AIS, men som fungerer i nettleserene Firefox og Mozilla. Verktøyet er utviklet av Chris Pederick, og kan lastes ned fra: <http://chrispederick.com/work/webdeveloper/>

Cynthia Says

Det finnes flere verktøy som kan brukes for å kontrollere tilgjengelighet. Cynthia Says er et eksempel. Cynthia Says kan teste en side ut i fra WCAG-retningslinjene og leverer en rapport som tilsvarer WCAG-sjekklisten fra WAI. Cynthia Says er integrert i Accessibility toolbar og finnes dessuten på adressen: www.cynthiasays.com

Dessverre finnes det ingen automatiserte verktøy som kan teste hvorvidt et nettsted kan sies å være fullt ut tilgjengelig. Det er for eksempel ikke mulig med automatisert testing av hvorvidt en "alt-tekst" er meningsbærende. Slike former for testing må utføres av personer som har kompetanse på området.

Vær også oppmerksom på at testen bør dekke alle enkeltsider på et nettsted, ikke bare startsiden. Fortsatt er det slik at ingen av de testverktøyene som er gratis, tester hele nettstedet.

Accessible Table Builder

Accessify har utviklet et verktøy som gjør det litt enklere å sette opp tabeller som tilfredsstiller tilgjengelighetskravene. Verktøyet heter Accessible Table Builder og gir deg den HTML-koden du trenger for å lage tabeller. For å bruke verktøyet må du ha et publiseringsverktøy som lar deg jobbe direkte i HTML-koden.

Se: www.accessify.com/tools-and-wizards/accessible-table-builder_step1.asp

4.2 HTML - verktøy

W3Schools

W3Schools er en referansemanual for HTML. Her kan du se hvilke HTML-tagger som er i bruk og hvilke attributter som finnes. Her kan du også finne ut hvilke tagger som hører til i hvilke HTML-versjoner, hva taggene kan brukes til og hvilke nettlesere de fungerer i. Se: www.w3schools.com/

W3Schools er integrert i Accessibility Toolbar. Se: enypunktet References > Language references, eller på adressen: www.w3schools.com/tags/

W3C HTML validator

W3C's HTML-validator kan brukes for å teste hvorvidt nettsiden overholder en valgt HTML-standard. Dersom HTML-koden ikke er riktig, er det stor sannsynlighet for at nettsiden heller ikke er tilgjengelig. Testing av nettsider bør derfor alltid starte med en test av HTML-koden.

HTML-validatoren er integrert i Accessibility Toolbar, men finnes også på adressen: <http://validator.w3.org>

Med HTML-validatoren kan man både teste sider som er lagt ut på nettet og laste opp og teste sider som ennå ikke er lagt ut.

W3C CSS validator

W3C's CSS-validator kan brukes for å teste hvorvidt nettsiden overholder en valgt CSS-standard. Dersom CSS-koden ikke er riktig, er det stor sannsynlighet for at nettsiden ikke vil bli vist korrekt i brukerens nettleser. Validatoren finner du på: <http://jigsaw.w3.org/css-validator/>

HTML Tidy

HTML Tidy er et program som rydder opp i HTML-dokumenter. Programmet kan fjerne spesielle koder som blir satt inn når Microsoft Word brukes til å generere HTML-dokumenter. HTML Tidy er integrert i Accessibility Toolbar, men du finner den også på adressen: www.infohound.net/tidy/

4.3 Generell kunnskap

Hvordan funksjonshemmede opplever nettet

Film med funksjonshemmede nettbrukere kan lastes ned fra: www.doit.wisc.edu/accessibility/video

Offentlige retningslinjer for nettsteder

Du finner nærmere informasjon om hvilke krav til som stilles til offentlige nettsteder på Norge.no. Retningslinjer for hvordan Internett kan bli tilgjengelig kan du finne på sidene til Norge.no og Sosial- og helsedirektoratet / Deltasenteret. Se: www.norge.no/kvalitet/default.asp
http://www.shdir.no/samspill/involvere_brukerne/internett_for_alle/
<http://www.shdir.no/deltasenteret/ikt/internett/>

Heftets tittel: Tilgjengelige nettsteder
2:3 – Design og koding

Bestillingsnummer: IS-1417

Utgitt av: Sosial- og helsedirektoratet
Deltasenteret
Postboks 7000
0130 Oslo
Tlf.: 810 20 050
www.shdir.no/deltasenteret

Forfattere: Haakon Aspelund
Frank Fardal
Aina Olsen

Trykk: Flisa trykkeri

Papirkvalitet: Omslag: Multi art silk 250 g
Innsider: Multi art silk 130 g

Opplag: 2000

Utgitt: November 2006

Elektronisk utgave: www.shdir.no/deltasenteret

**Veilederen kan
Bestilles hos:** Sosial- og helsedirektoratet
v/Trykksakekspedisjonen
e-post: trykksak@shdir.no
Telefon: 24 16 33 68
Telefaks: 24 16 33 69