

Tilgængelighed

Formålet med at teste for tilgængelighed er at sikre at et website er brugbart for brugere med særlige behov. Særlige behov kan have baggrund i problemer relateret til syn, motorik, kognition, mm. Derfor er det væsentligt at fokusere på andre end blinde brugere.

Krav

Kravene til websites er beskrevet i en eller flere standarder, anbefalinger og lignende. De mest kendte er: *Web Content Accessibility Guidelines (WCAG) 2.0* fra W3C og *Electronic and Information Technology Accessibility Standards (Section 508)* fra USA.

For danske websites kommer der til at gælde:

3/12 2012 EU-Kommissionen fremsætter direktivforslag om offentlige hjemmesiders tilgængelighed

I direktivforslaget anføres den internationale standard ISO/IEC 40500:2012 AA, som det niveau, de omfattede hjemmesider skal overholde. Standarden er også kendt som WCAG 2 level AA, og er i Danmark allerede en af flere obligatoriske, åbne standarder for offentlige myndigheder.

Kilde:

<http://www.digst.dk/ServiceMenu/Nyheder/Nyhedsarkiv/Digitaliseringsstyrelsen/EUKommissionen-fremsaetter-direktivforslag-om-offentlige-hjemmesiders-tilgaengelighed>

Det betyder at offentlige institutioner senest når direktivet er vedtaget skal efterleve WCAG 2.0 level AA.

WCAG 2.0 grupperer kravene på tre niveauer (levels) med A som de mindst omfattende og AAA som mest omfattende.

Det er væsentlig at forstå at opfyldelse af kravene også kan være en fordel for brugergruppen uden særlige behov. Et par eksempler:

- En fyldestgørende title på et link kan hjælpe alle, ikke kun blinde.
- Muligheden for at kunne nå enhver funktion på siden vha. tabulator/enter er en fordel for brugere med motoriske nedsættelser eller brugere af smartphones.

I nogle situationer kan opfyldelse af et krav for en brugergruppe være til gene for en anden. 'Læs mere...' er et eksempel hvor linkteksten til forskellige URL'er skal være forskellige af hensyn til blinde. Normalt seende opfatter derimod 'Læs mere...' som et ikon der får mening ud fra de nære omgivelser. En mere fyldestgørende tekst vil derfor være overflødig for dem.

Værktøjer og strategier

Det er ikke muligt at foretage test med brugere med alle former for behov. Derfor er der udviklet en række værktøjer til at foretage inspektion af html siderne. Værktøjerne tager udgangspunkt i ovennævnte guidelines og standarder suppleret med nogle mere operationelle krav fx *Illinois Information Technology Accessibility Act* (<http://www.dhs.state.il.us/IITAA/IITAAStandards.html>).

Der findes to strategier i forbindelse med analyse af tilgængelighed: en komplet automatiseret analyse af alle sider på et website eller selektiv analyse af udvalgte sider ved hjælp af en kombination af automatiske og manuelle værktøjer. Den sidste er bedst, da man let kommer til at overse, at selv den bedste automatiske analyse kræver menneskelig vurdering på en række punkter. Eksempelvis er kravet om, at et billede skal have en fornuftig beskrivende alt-tekst en

vurdering, der ikke kan automatiseres.

Den selektive analyse er også den billigste, hvis siderne på websitet er lavet ud fra et lille antal skabeloner.

Det primære værktøj til den selektive analyse kan være et antal plugins som findes til Firefox, nogle procedurer og nogle online services.

- <http://validator.w3.org/>
En forudsætning for at opfylde WCAG 2.0 er at siden validerer. W3C har lavet en række online services, som bl.a. kan validere HTML og CSS. Når der valideres gøres det op mod en eller flere standarder. Det er svært at forholde sig til hvad der er 'standard', fx er HTML5 ikke en standard; man kan godt validere i henhold til denne doctype. Det kan give fejl, som ikke er fejl.
Et eksempel er, at brug af protokollen 'tel' i `` giver fejl.
- <http://www.atutor.ca/achecker/>
Online service, AChecker, hvor man på sidebasis kan få en kombineret validering af HTML og CSS med WCAG 2.0 og 508. Input kan enten være en URL, en fil der uploades eller html, der pastes i et felt. Resultatet er fx en .pdf, der meget detaljeret beskriver fejl, problemer og ting, der skal kontrolleres manuelt. En typisk rapport for en side 'uden fejl' er på ca. 10 sider og beskriver omkring 100 ting der skal kontrolleres manuelt.
- Total Validator
Plugin til Firefox kombineret med program (<http://www.totalvalidator.com/>) er et fornuftigt værktøj, der kombinerer valideringen af HTML og CSS med WCAG 2.0 og 508. Den giver overordnet set et retvisende billede; men resultatet skal vurderes. Eksempelvis er '&' i en URL ikke valid, der skal bruges '&'; tilsvarende giver æøå/ÆØÅ i en URL fejl; men det er uvæsentlig i relation til tilgængelighed.
- Accessibility Inspector Extension for Firebug - a11y
Extension til Firebug plugin (<http://code.google.com/p/ainspector/>) som i stil med Total Validator giver komplet validering. Der mangler stadig noget udvikling; men det er en fordel, at den integrerer med Firebug, som bruges i udviklingssammenhæng.
- Semantic Checker
Plugin til Firefox (<http://www.semantic-checker.com/>) der finder og markerer semantiske HTML4 og HTML5 elementer, HTML5 input attributes og elementer opmærket som microformat. Det giver en visuel markering på siden af den semantiske struktur. Der udregnes nøgletal for brugen af semantiske elementer. Et højt indhold af semantiske elementer er positivt set fra et tilgængeligheds synspunkt.
Bemærk, at 'strong' regnes med som semantisk element og hvis det bruges i stedet for headings (h1, H2...) så er det problematisk.
En tommelfingerregel er, at et forhold mellem semantiske elementer og ikke-semantiske elementer (gange 100) på 10 eller derover er fint.
- Fangs Screen Reader Emulator
Plugin til Firefox (<https://addons.mozilla.org/en-us/firefox/addon/fangs-screen-reader-emulator/>) der emulerer en skærmlæser. Der er mulighed for at se strukturen i form af overskrifter og links.
- Juicy Studio Accessibility Toolbar
Plugin til Firefox (<https://addons.mozilla.org/da/firefox/addon/juicy-studio-accessibility-too/>) som kan vise ARIA opmærkningen i DOM'en i form af områder og roller. Derudover kan man få en analyse af relationen mellem alle kombinationer af color/background color.

Bemærk: Man kan få forkerte resultater i forbindelse med Colour Contrast Analyse, da denne plugin fx ikke kan tolke baggrund angivet som gradient korrekt.

- Web Developer
Plugin til Firefox (<http://chrispederick.com/work/web-developer/>) indeholder gode værktøjer til forståelse af en sides html struktur også i nogen udstrækning i relation til tilgængelighed.
- Accessibility Evaluation Toolbar
Plugin til Firefox (<https://addons.mozilla.org/en-US/firefox/addon/5809/>) indeholder værktøjer, som kan validere mod de retningslinjer der gælder for University of Illinois at Urbana-Champaign. Værktøjerne kan dog med fordel bruges enkeltvis fx til at vise alle access keys på en side.
- Navigation udelukkende ved brug af tastatur (procedure)
Man skal sikre sig, at man kan ramme alle funktioner/links på siden udelukkende ved brug af tastatur. Normalt er det en kombination af TAB og ENTER der skal bruges; men man skal også teste at eventuelle access keys virker. Det indbefatter også at det aktive element er klart visuelt markeret og at rækkefølgen man kommer til elementerne er logisk.
Web Developer Toolbar har et menupunkt 'Information', der har mulighed for at vise disse forskellige elementers placering på siden. Da nogle elementer kan være placeret uden for det synlige område, så kan man også med fordel anvende Accessibility Evaluation Toolbar til at identificere disse.
- Kontrol af opbygningen af forms og tabeller (procedure)
I forms skal man sikre sig at der er knyttet en forklarende label til alle brugertilgængelige input elementer. Elementerne skal have en type der afspejler deres mulige indhold (brug de nye html5 typer). Man skal også kontrollere, at link og button er brugt på en naturlig måde i forhold til omgivelserne og at der i tvivlstilfælde er foretaget en ARIA-opmærkning.
Web Developer har to værktøjer i menuen Forms, der kan bruges. Outline Form Fields without Labels og View Form Information.

Juicy Studio Accessibility Toolbar har en 'Table Inspector' som viser summary og headings for søjler og rækker. Web Developer har mulighed for visuelt at vise tabeller og deres struktur; men der skelnes fx ikke mellem header og celler.

- Kontrol af alt attribute på billeder (procedure)
Web Developer har et menupunkt 'Images', hvor man kan markere, at man vil have Show Alt Attribute. Det bevirker, at alt-tekst vises i forbindelse med billedet, så det er nemt at se om der er en og om den angivne tekst er relevant i forhold til billedet i den aktuelle sammenhæng. Husk, at dekorative billeder skal være forsynet med en tom tekst.
- adgangforalle.dk
(<http://www.adgangforalle.dk/>) er et slutbruger værktøj som kan læse dele af en side op i form af syntetisk tale. Adgangforalle kan bruges af ordblinde og dårlige læsere, men ikke af blinde og meget svagtseende personer, da de ikke kan markere en tekst, som er en forudsætning for at få oplæst noget indhold.

Et af de bedre værktøjer til automatisk analyse af sider var www.cynthiasays.com. Den indgår i Web Developer Toolbar, men den har dog i en periode været umulig at bruge, da der en IP baseret blokering som bl.a. rammer Danmark. Derfor anbefales AChecker og Total Validator.