

# 1: Værktøjer til test af tilgængelighed

Præsenteret af Lars Michael Sørensen, CPWA

## Værktøjer til test af tilgængelighed

Lars Michael Sørensen, CPWA



Netvenlig.dk

Velkommen til jer alle!

## 2: Om Lars Michael

### Lars Michael Sørensen, CPWA

Specialist i anvendeligt indhold og tilgængeligt design

- Certified Professional in Web Accessibility
- Medforfatter på WCAG 2.1 (Dansk)
- Undervisning og rådgivning
  - Webtilgængelighed
  - Brugervenlighed
  - Skriv til nettet
  - Informationsarkitektur

LinkedIn: /in/lmsoren

[LMSOREN@netvenlig.dk](mailto:LMSOREN@netvenlig.dk)



Jeg hedder som sagt Lars Michael og jeg er specialist i anvendeligt indhold og tilgængeligt design.

Da jeg skulle lave præsentationen her, så gik det op for mig, at jeg har en slags jubilæum: Det er nemlig 30 år siden i år, at jeg begyndte at arbejde med digitalt indhold.

Jeg er også certificeret i webtilgængelighed. Det er det, jeg arbejder mest med og også underviser i, især på Teknologisk Institut og hos virksomheder og myndigheder, der vil lære om webtilgængelighed.

Jeg er – blandt andre sammen med Anne – medforfatter på den danske oversættelse af WCAG 2.1

Hvis du har spørgsmål til noget jeg fortæller om, og du først kommer i tanke om det i morgen eller efter mødet, så er du velkommen til at finde mig på LinkedIn eller sende en email. Jeg synes kun det er hyggeligt at høre fra folk, så hold dig ikke tilbage!

### 3: Indhold i præsentationen

#### Jeg vil tale om

- Mest almindelige tilgængelighedsfejl på offentlige websteder
- Værktøjer til at undersøge
  - Kontrastforhold
  - Alt-tekster
  - Tastaturfokus og læserækkefølge
  - Overskrifter
  - Landmærker og regioner
- Tilgængelighedstræet
- Automatisk test af websider

Jeg skal fortælle jer lidt om værktøjer til at teste webtilgængelighed med.

Først vil jeg lige give en oversigt over de mest almindelige fejl på websteder der er omfattet af webtilgængelighedsloven.

Med det i baghovedet, så kigger vi på nogle konkrete værktøjer til at lede efter tilgængelighedsfejl med.

Jeg vil tale om værktøjer til at undersøge Kontrastforhold, Alt-tekster, Tastaturfokus og læserækkefølge, Overskrifter, landmærker og regioner.

Jeg vil også fortælle kort om det, der kaldes Tilgængelighedstræet. Og når vi det, så slutter jeg af med at vise to værktøjer, som kan teste hele sider på en gang.

Hvis du henter præsentationen på Digitaliseringsstyrelsens websted, så er der til sidst i dokumentet en liste med links til de forskellige værktøjer, jeg præsenterer.

## 4: Mest almindelige tilgængelighedsfejl i 2024 og 2025

### Mest almindelige tilgængelighedsfejl i 2024 og 2025

Dybdegående monitorering er en manuel evaluering af websted

Mest fejlede WCAG-kriterier	2025	2024
1.4.4 Ændring af tekststørrelse	-	95%
1.3.1 Information og relationer	82 %	91%
4.1.2 Navn, rolle, værdi	74 %	86%
1.4.11 Kontrast for ikke-tekstbaseret indhold	65 %	80%
3.1.2 Sprog, der angives til dele af indholdet	65 %	-
1.1.1 Ikke-tekstbaseret indhold	59 %	-
2.4.3 Fokusrækkefølge	57 %	-
1.4.10 Ombrydningsvisning	54 %	90%

Digitaliseringsstyrelsen: Overordnede resultater, digst.dk

Her er en tabel over de mest almindelige tilgængelighedsfejl i de dybdegående monitoreringer af websteder, der er udført over de sidste to år.

Hvert år udføres 23 manuelle evalueringer, hvor man går webstederne igennem for alle 50 kriterier i WCAG 2.1, plus eventuelle kriterier fra EN 301 549.

Man skal lige vide, at tallene for 2024 er en Top 5, men tallene for 2025 er en Top 7, så det er ikke alt, vi kan afkode af tallene her, men det er stadig tydeligt, at der faktisk sker fremskridt.

Især på forhold omkring zoom og små skærmstørrelser kan vi se en ret markant forbedring. Det er kriterierne 1.4.4, der fejlede på 95 procent af de testede websteder i 2024, men slet ikke er på top 7 i 2025 og 1.4.10, som fejler på 90 procent i 2024 men kun 54 procent i 2025.

Der er også gode fremskridt med kontrastforhold for brugergrænsefladeelementer og grafiske objekter. Det er 1.4.11 jeg taler om. I 2024 fejlede kriteriet på 80 procent af de testede websteder, men sidste år var det faldet til 65 procent.

Der er et behov for at blive bedre i forhold til information, struktur og sammenhæng i indholdet. Det er 1.3.1 jeg taler om nu, men en årsag til at den slår så højt ud med 91 og 82 procent er, at en fejl i 1.3.1 ofte

afspejler, at der også er fejl i andre kriterier, for eksempel kriterier der vedrører sidens visuelle struktur – at overskrifter er mærket op med en forkert overskriftkode, eller at inputfelter og deres etiketter ikke er mærket op, så det er tydeligt for alle, at de hører sammen.

Lidt senere vil vi se på værktøjer til at undersøge nogle af de ting nærmere, men lad os begynde med kontrastforhold

## 5: Kontrastkrav til tekst og billeder af tekst

### Kontrastkrav til tekst og billeder af tekst


Kontrastforhold for tekst og billeder af tekst:

- Mindst 4,5:1 for normal tekst og mindst 3:1 for stor tekst
- Stor tekst = mindst 14 punkt fed eller 18 punkt normal

Undtagelser:

- Pynt og udsmykning
- Tekst ved inaktive brugergrænsefladekomponenter
- Logotyper er ikke omfattet af kravene

Minimum 4,5:1



Hvis vi lige skal tage reglerne for kontrastforhold først, så siger de, at ...

Kontrast for tekst og billeder af tekst skal være mindst 4,5:1 for almindelig tekst og mindst 3:1 for stor tekst

Stor tekst er defineret som mindst 14 punkt fed eller mindst 18 punkt normal.

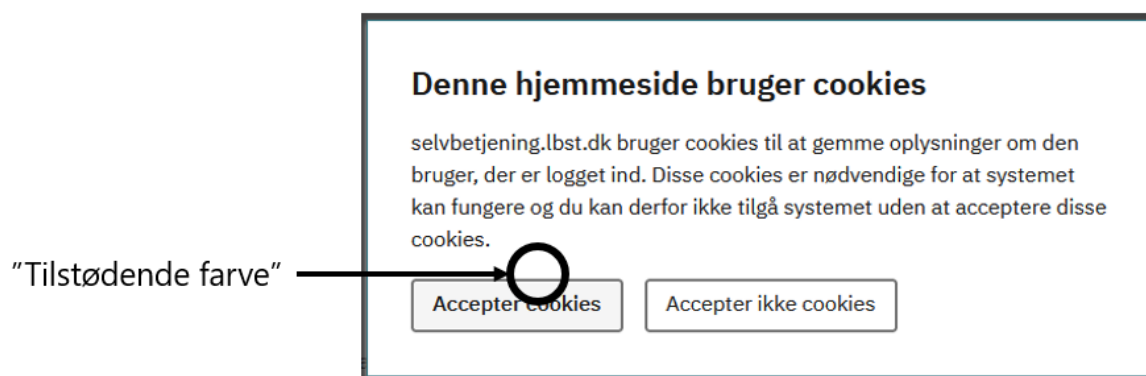
Og fordi ingen kan regne i punkt længere, så er 14 punkt næsten det samme som 19 pixel og 18 punkt svarer til 24 pixel.

## 6: Kontrastkrav for brugergrænsefladeelementer

### Kontrastkrav til brugergrænsefladeelementer

Minimum 3:1 mod "tilstødende farver"

Ingen kontrastkrav til inaktive elementer



Brugergrænsefladekomponenter som knapper og inputfelter har også kontrastkrav. Det er 3:1 "mod tilstødende farver". De tilstødende farver er som oftest en ensfarvet baggrundsfarve, men hvis knappen her nu lå oven på et fotografi med mange farver, så kan det godt blive nødvendigt at måle kontrastforholdet mod tilstødende farver flere steder rundt om knappen.

Der er ikke en fast regel for, hvor meget der skal være synligt af et element, eller hvor meget det må forsvinde ind i baggrunden - og det er ikke ulovligt at være en dårlig designer - men sørg for at gøre det let for folk at finde det, I gerne vil have dem til at klikke på!

Inaktive grænsefladeelementer er ikke omfattet af kontrastkrav. Så kan man ikke klikke på knappen før man har gjort noget andet først, er det ikke forkert at gøre den næsten usynlig.

Der er efterhånden en del gode værktøjer til at måle farvekontrast:

Flere af jer er sikkert godt inde i Adobe og Figma-verden, og begge har rigtig fine værktøjer til kontrastmåling.

## 7: Adobe Color Contrast Checker

### Adobe Color Contrast Checker

- [Adobe Color Contrast Checker](#)
- [Figma Contrast Plugin \(WillowTree\)](#)
- [TPGi Color Contrast Analyser](#) (min favorit)

**Adobe Express**

### Color contrast checker (AA/AAA).

Check foreground and background color contrast ratios against WCAG 2.1 accessibility guidelines. Test combinations for AA and AAA standards to help ensure readable, accessible design.

Contrast ratio 4.55 : 1 [Compare entire palette](#)

Foreground color #8DA032 Background color #002E66

0.6 0.4

Summary Contrast suggestions Set a contrast ratio

Category	AA	AAA
Large text	Pass ✓	Pass ✓
Small text	Pass ✓	Fail ✗
Graphics and UI	Pass ✓	N/A

Adobes værktøj er webbaseret og alle kan bruge det, også selv om man ikke bruger adobes produkter til daglig.

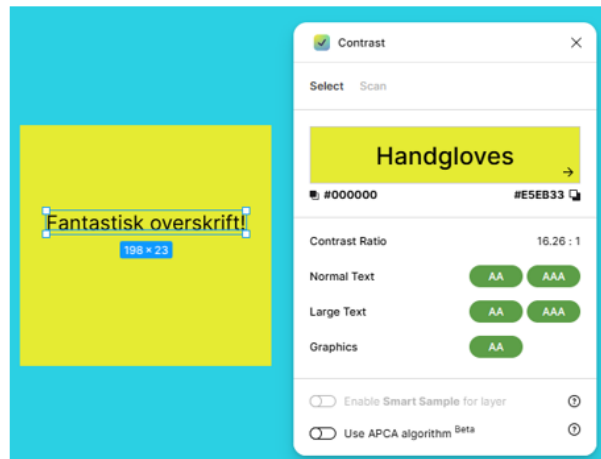
Man finder de hexadecimale farveværdier for baggrund og forgrundsfarve og sætter dem ind i felterne, og så beregner værktøjet selv kontrastforholdet og fortæller, hvordan de passer ind i WCAG-kravene.

Ulempen her er, at man skal selv finde en måde at få fat i farvekoderne. Enten med en farvepipette af en slags, eller man skal ind og grave i webstedets designark – det kan lade sig gøre, men det er ikke altid sjovt.

## 8: Figma Contrast Plugin (WillowTree)

### Figma Contrast Plugin (WillowTree)

- Adobe [Color Contrast Analyzer](#)
- **Figma Contrast Plugin (WillowTree)**
- [TPGi Color Contrast Analyser](#) (min favorit)



Det er lidt det samme i Figma. Men måske har man allerede let adgang til farvepaletten, hvis man er involveret i designprocessen på dette niveau.

Figma har forskellige communityudviklede plugins til kontrastmåling, og her ser vi et af de mest populære, lavet af brugeren WillowTree.

Som sagt er der links til alle værktøjer jeg omtaler, i slutningen af præsentationen.

## 9: Vispero Color Contrast Analyser

### Vispero Color Contrast Analyser

- Adobe [Color Contrast Analyzer](#)
- [Figma Contrast Plugin \(WillowTree\)](#)
- **[Vispero Color Contrast Analyser \(min favorit\)](#)**



#### Ansøgningssystem

På Landbrugsstyrelsens hjemmeside er indgangen til ansøgningsskemaet er via "Tast selv".

Det er en god idé at læse Plan og landdistriktsstyrelsens [vejledning](#), inden du påbegynder.

Find hjælp til at oprette din ansøgning i [hjelpecenteret](#)

**GÅ TIL ANSØGNINGSSYSTEMET**

#### VEJLEDNING

Det er en god idé at læse Plan og landdistriktsstyrelsens. Her kan du

Selv foretrækker jeg at bruge Color Contrast Analyser fra Vispero, som er en selvstændig applikation, der også gør det muligt at måle farvekontrast udenfor webbrowseren, for eksempel i dokumenter.

Color Contrast Analyser har det hele indbygget, inklusiv en farvepipette, der kan suge farvekoden fra ethvert sted på computerskærmen.

Resultatet kommer med det samme i bunden af vinduet, og har man brug for at justere en farve til den overholder mindstekravet, kan man gøre det direkte i panelet.

## 10: Simulering af farveblindhed

### Vispero Color Contrast Analyser

- Adobe [Color Contrast Analyzer](#)
- [Figma Contrast Plugin \(WillowTree\)](#)
- **[Vispero Color Contrast Analyser \(min favorit\)](#)**



#### Ansøgningssystem

På Landbrugsstyrelsens hjemmeside er indgangen til ansøgningsskemaet er via "Tast selv".

Det er en god idé at læse Plan og landdistriktsstyrelsens [vejledning](#), inden du påbegynder.

Find hjælp til at oprette din ansøgning i [hjelpecenteret](#)

**GÅ TIL ANSØGNINGSSYSTEMET**

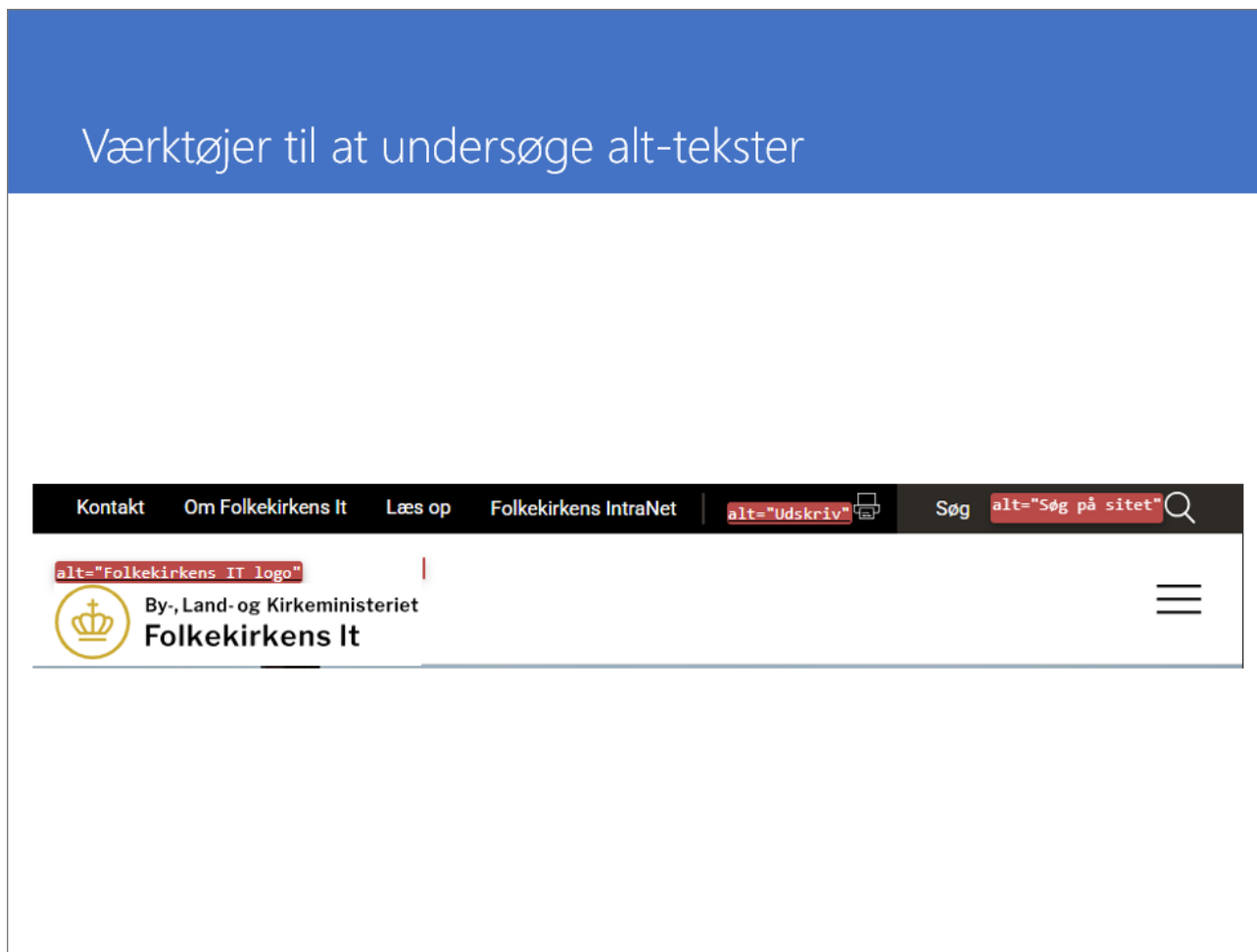
#### VEJLEDNING

Det er en god idé at læse Plan og landdistriktsstyrelsens. Her kan du

Har man lyst, kan man også bruge Color Contrast Analyser til at undersøge, hvordan forskellige farvekombinationer fungerer for personer med nedsat farvesyn – altså farveblinde

Programmet er både til Windows og Mac, men det forudsætter jo, at I kan - og må - installere software på jeres computere, og det er ikke alle som har lov til det.

## 11: Værktøjer til at undersøge alt-tekster



Alt-tekster eller "Alternative tekstbeskrivelser" kan have stor betydning for, hvordan man opfatter indhold på en webside, hvis man ikke kan se det.

Gode alt-tekster formidler den nødvendige information til den der hører den, og ikke andet.

Nogle gange skal der være alt-tekster, og andre gange skal der ikke. Alt-tekster er ikke "one size fits all". De kræver, at man tager stilling til, hvad formålet med et billede eller et stykke grafik er, på den side hvor man viser det.

Alle der har været på kursus hos mig ved, at jeg bruger ret meget tid på at tegne og fortælle om alt-tekster, for der er mange forskellige scenarier at tage fat i.

Men det vil jeg altså ikke gøre her.

## 12: Kodeinspektør i browseren

The screenshot shows a web browser displaying the website 'FOLKEKIRKEN'. The page has a blue header with the logo and navigation links: 'Livets begivenheder', 'Årets højtid', 'Eksistens', 'Tro', and 'Om folkekirken'. The main content area features a podcast player on the left and a news article on the right. The news article is titled 'Kirken får teolog og salmedigter som ministre' and includes a sub-headline 'Danmark har fået ny regering, og nøglerne til'. A code inspector is overlaid on the bottom half of the browser, showing the HTML structure of the page. The code includes a grid layout and an image element with the following attributes: ``

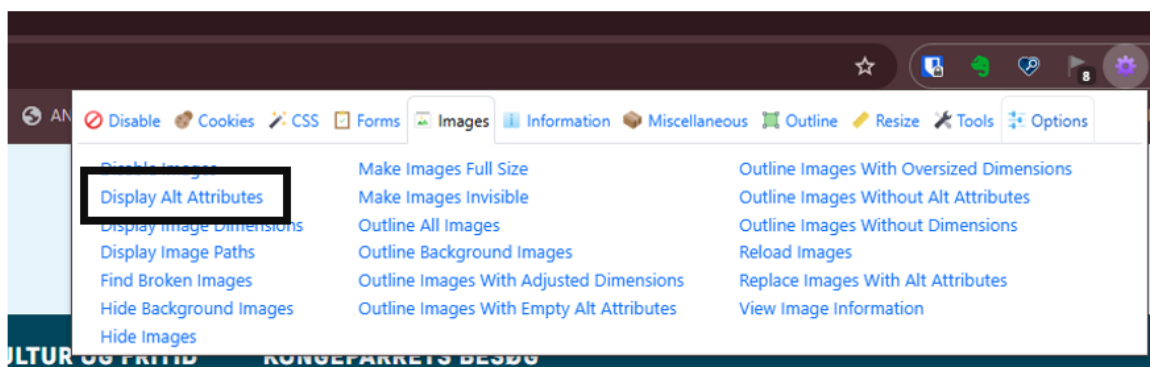
Der er flere måder at undersøge alt-tekstbeskrivelser på.

Den simpleste måde er at åbne browserens Kodeinspektør og gå billederne igennem, et for et. Det kan godt tage en del tid, hvis der er meget grafik på siden,

så en anden, mere effektiv måde er at bruge den browserudvidelse, der hedder Web Developer.

## 13: Web Developer Plugin

### Web Developer plugin



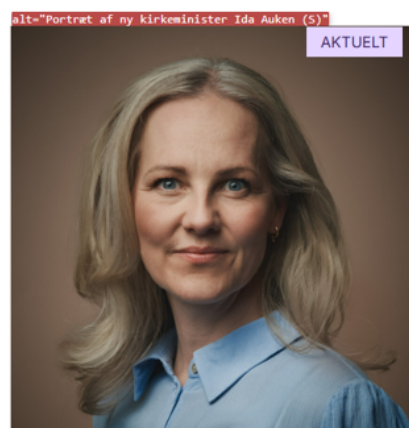
Web Developer har rigtig mange indstillinger, også når det kommer til visning og håndtering af billeder og grafik.

Jeg bruger meget den indstilling vi ser her: Display Alt Attributes – vis alt-tekster.

## 14: Alt-tekster med Web Developer Plugin

### Alt-tekster med Web Developer plugin

#### Stof til eftertanke



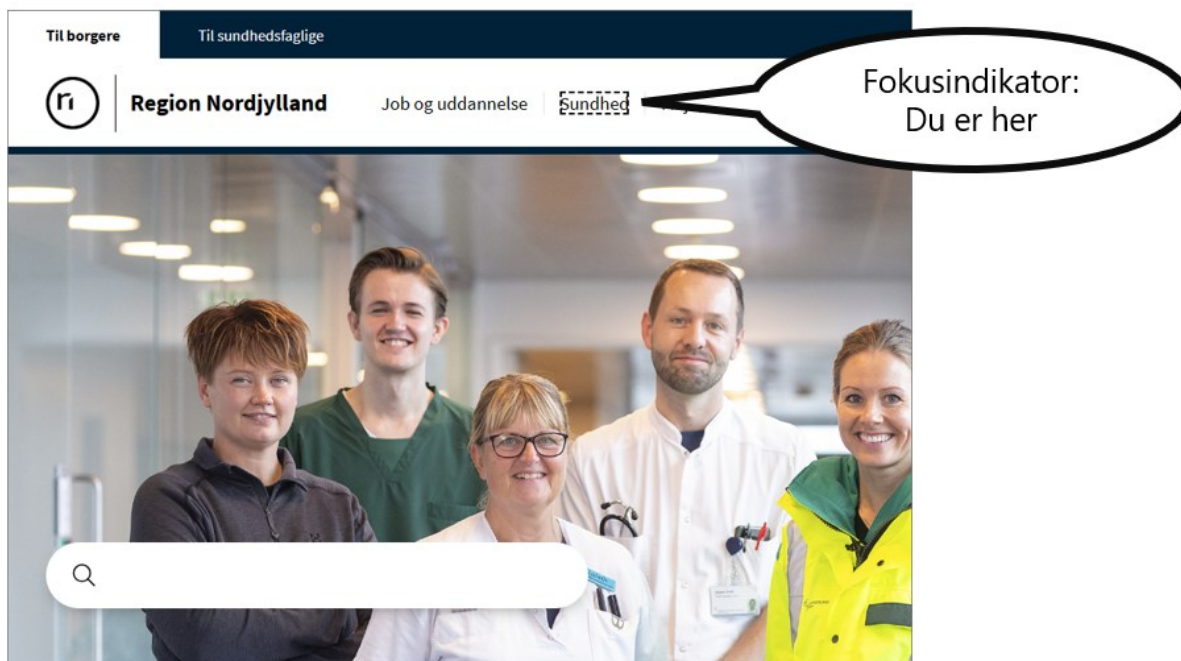
#### Kirken får teolog og salmedigter som ministre

Danmark har fået ny regering, og nøglerne til

Når den er valgt, får man alt-teksterne vist sammen med billedet, de hører til. Det er bare en af måderne, Web Developer kan vise alt-tekstindhold på. Du kan også slå billeder på siden fra, så der kun er alt-tekster og det giver en rigtig god ide om, hvor informerende dine alt-tekster faktisk er.

## 15: Tastatur og læserækkefølge

### Tastaturfokus og læserækkefølge



Tastaturfokus og læserækkefølge er ikke direkte med i listen over mest almindelige tilgængelighedsfejl, men at kunne orientere sig når man navigerer med tastatur er essentielt for alle, der ikke anvender en mus.

Tastaturfokus er let at teste. Du trykker på TAB-tasten og observerer, at fokusindikatoren flytter sig fra det ene klikbare element på siden, til det næste, indtil du har været alle igennem.

Der må ikke være interaktivt indhold på siden, du ikke kan nå med tastaturet alene.

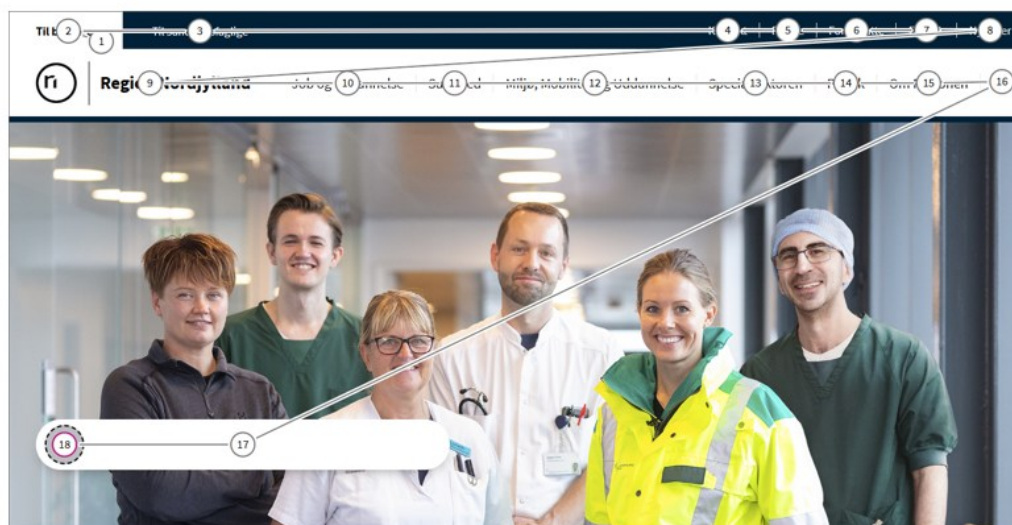
Kommer du til en komponent, så betjenes de nogle gange med piletasterne eller med Retur og mellemrum.

Det sker, at jeg har brug for at kunne dokumentere en bestemt tastaturrekkefølge, eller antallet af tryk fra et sted på siden til et andet. Og til det bruger jeg Microsoft-udvidelsen Accessibility Insights for Web.

## 16: Værktøj: Accessibility Insights for Web

### Værktøj: Accessibility Insights for Web

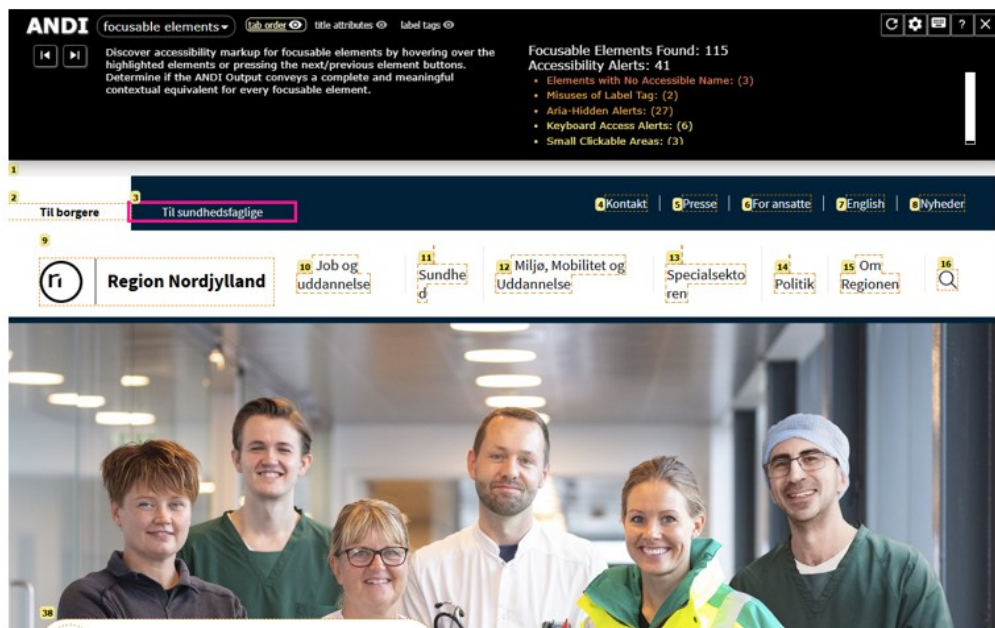
- Plugin fra Microsoft, virker i alle browsere
- Kan også udføre semi-automatiske tests / Assesments



Med det kan jeg vælge at illustrere tabulatortryk visuelt som tal på en række, sådan som vi ser det her. Udvidelsen har flere gode funktioner og jeg er i det hele taget meget glad for den, kan jeg hilse at sige!

## 17: Tastaturrekkefølge med ANDI

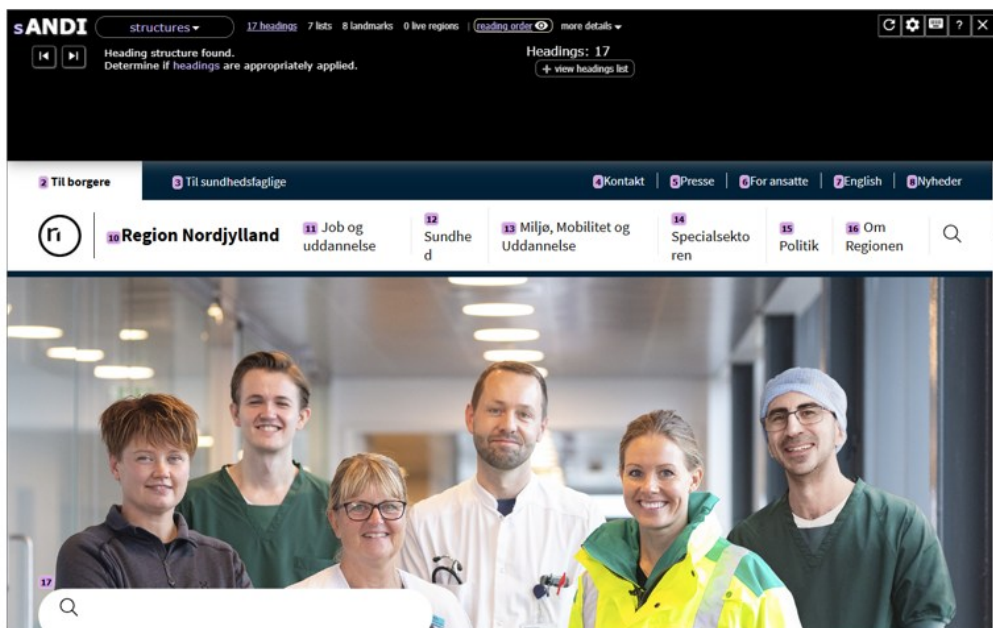
### Tastaturrekkefølge med ANDI



Vil man hellere noget andet, så er udvidelsen ANDI – der teknisk set er en såkaldt Bookmarklet - et meget kompetent alternativ

## 18: Læserækkefølge med ANDI og Web Developer

### Læserækkefølge med ANDI og Web Developer



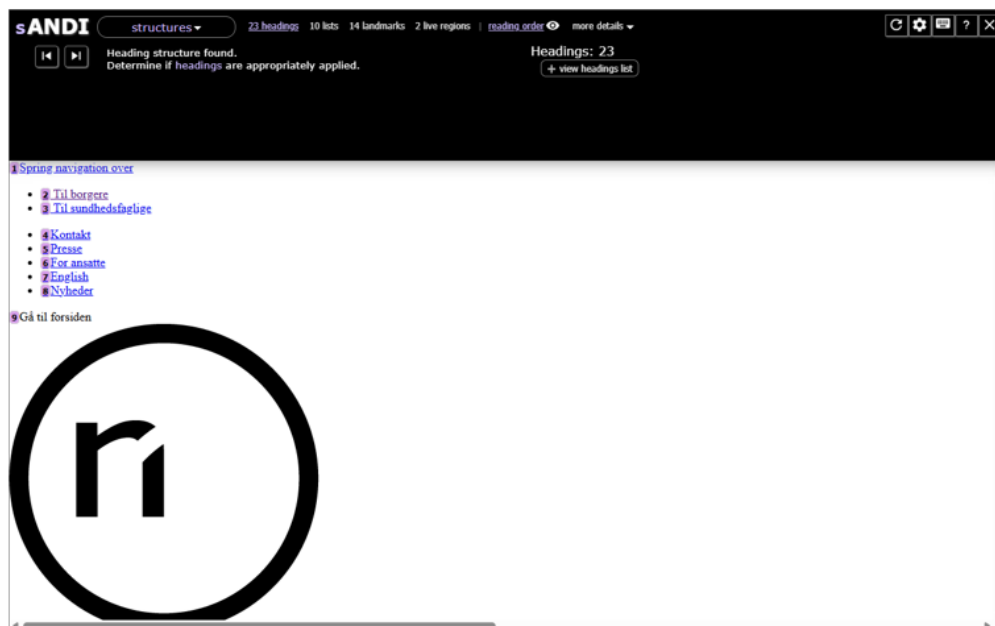
ANDI kan vise en masse forhold, man ellers ikke lige kan hive frem, omkring en webside.

Tallene ud for de forskellige elementer viser den læserækkefølge, vi ville forvente.

Men vil man gerne længere ned i materien med det spørgsmål, så kan vi fjerne det visuelle design ved hjælp af Web Developer-udvidelsen. Og så kommer siden til at se sådan ud:

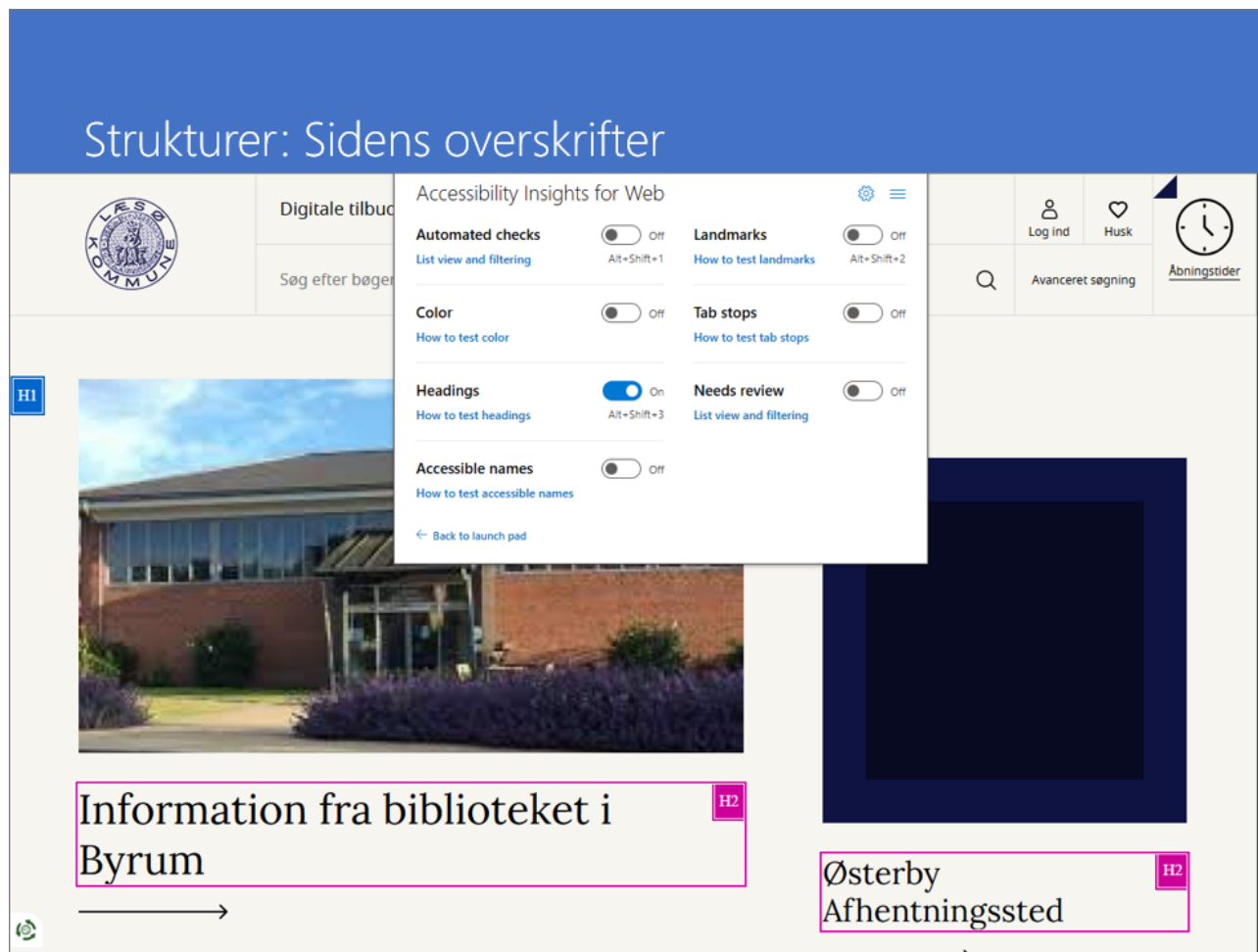
## 19: ANDI og Web Developer

### ANDI og Web Developer



Nu har vi kun indhold tilbage. I kan se en talrække og den repræsenterer ret præcist den rækkefølge som for eksempel en skærmlæser vil præsentere indholdet på.

## 20: Strukturer: Sidens overskrifter



To ting, man hurtigt kan tjekke, når man har installeret et af de tre plugins, vi har kigget på allerede, er overskrifter på siden, og de skjulte landmærker og regioner, der ligger i HTML-koden.

Begge dele fungerer som vejskilte, når man orienterer sig på en side.

Overskrifter foretrækker jeg at opdage med Accessibility Insights for Web.

Hver overskrift får en farvet boks og farven indikerer også, hvilket niveau overskriften er på.

I eksemplet her, kan vi faktisk opdage, at forsiden til Læsø Bibliotek har en skjult overskrift. Der er en blå H1 uden indhold, ude til venstre. Det ville jeg naturligvis gerne kigge nærmere på, hvis det var rigtigt.

Når jeg ruller ned ad siden, ville jeg også være interesseret, hvis jeg ser indhold, der visuelt ligner en overskrift, men som ikke bliver markeret med en farvet kasse. I de fleste tilfælde ville det udløse en fejl.

## 21: Regioner og landmærker

Strukturer: Regioner og landmærker

biblioteket.laesoe.dk

Digitale tilbud GO - Børnenes digitale bibliotek

Søg efter bøger, film, musik og mere

Information fra biblioteket i Byrum

Østerby

Elementliste

Type:  Lings  Overskrifter  Formularfelter  Knapper  Landmærker

- banner
  - Primary site navigation; navigation
  - formular
  - hovedindhold
  - information om indhold

Filtrér efter:

Aktivér Elyt til Annuller

Og jeg foretrækker også at bruge Accessibility Insights for Web til at finde landmærker og regioner.

Landmærker og regioner ses ikke på ydersiden. De er en del af koden, men de kan guide skærmlæserbrugere til at orientere sig og finde hurtigere frem til bestemte områder på en side.

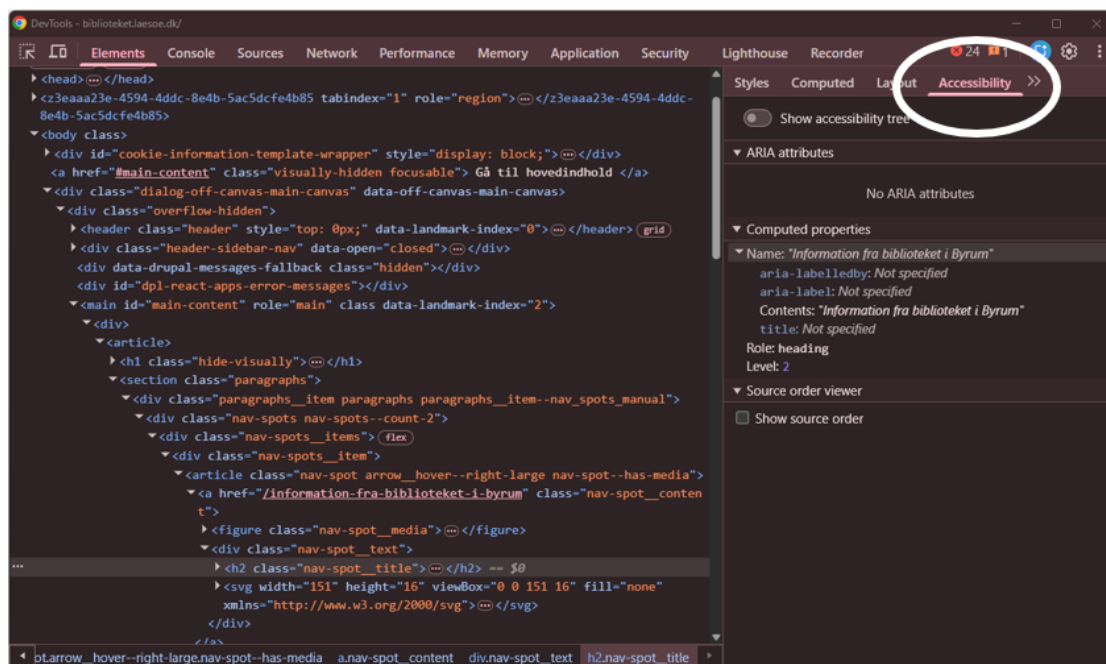
Som sagt vælger jeg gerne mit favorit-plugin til det her job, men I ser her et billede hvor jeg har slået skærmlæseren NVDA til, og åbnet det som kaldes Elementlisten.

Elementlisten er en genvejsliste til forskellige elementer på webstedet. Den kan vise alle links, overskrifter, inputfelter, knapper. Og også landmærkerne, som vi ser her.

Landmærkerne kan i øvrigt navngives med præcise angivelser af, hvad de indeholder, for eksempel at det er en hovednavigation eller en søgefunktion.

## 22: Tilgængelighedstræet 1

### Tilgængelighedstræet 1



I alle moderne webbrowsere findes der noget, som kaldes Tilgængelighedstræet.

Når en bruger taster en webadresse ind, så hentes kildekoden fra webserveren og bliver til det, man kan se i vinduet. Ud fra det, man aktuelt ser på siden, bliver der i baggrunden lavet en struktur, der afspejler alle elementer på siden, visuelle og indholdsmæssige. Skifter indholdet på siden automatisk, eller foretager brugeren selv en ændring, opdateres strukturen med det samme.

Denne struktur kaldes Tilgængelighedstræet og det er her, at hjælpeteknologier – ikke bare skærmlæsere, men også alle andre – får deres oplysninger om websidens aktuelle tilstand fra.

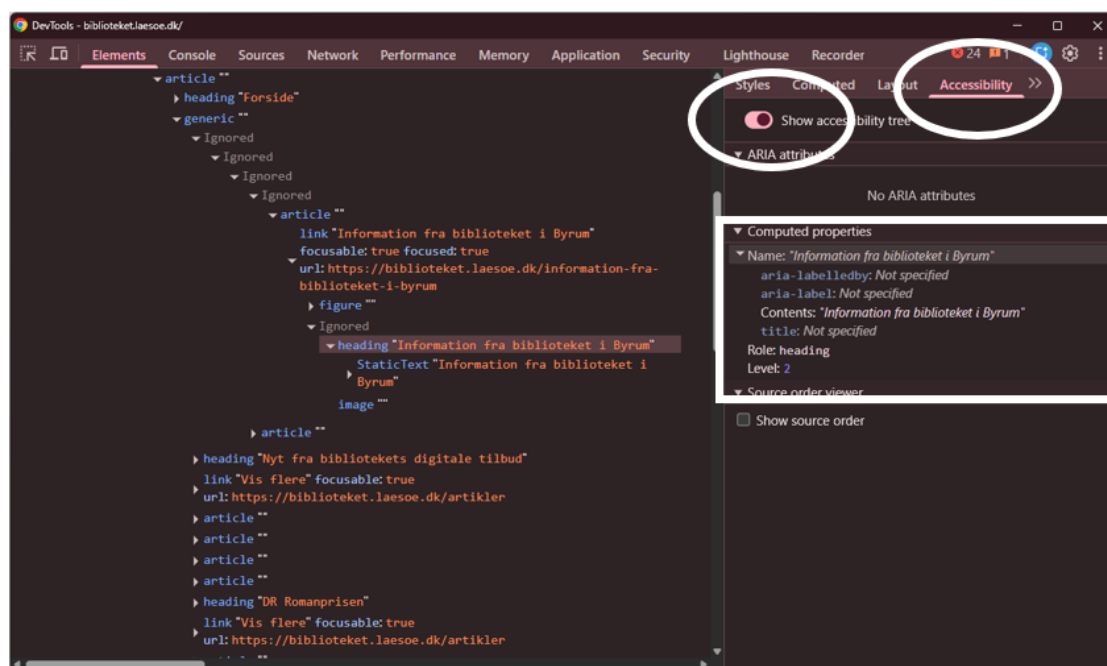
Det kan være en fordel at kunne arbejde med Tilgængelighedstræet, for de fleste hjælpeteknologier er ikke lette at blive gode til at anvende på en rimelig måde, hvis man ikke bruger dem til daglig.

Man kan kigge i Tilgængelighedstræet, og selv om det kan tage lidt tid at blive skarp på at bruge og forstå det, er det et supergodt værktøj, hvis man gerne vil vide, hvad en webside formidler videre til hjælpeteknologierne.

Man finder typisk Tilgængelighedstræet ved at åbne Kodeinspektøren. I Google Chrome kan man højreklikke på en side og vælge Undersøg eller bare trykke på F12.

## 23: Tilgængelighedstræet 2

### Tilgængelighedstræet 2



Afhængig af Kodeinspektørens opsætning, vil det se forskelligt ud, men du skal først finde knappen "Vis tilgængelighedstræet" og når du aktiverer den, skifter HTML-vinduet til at vise en struktureret form af websiden.

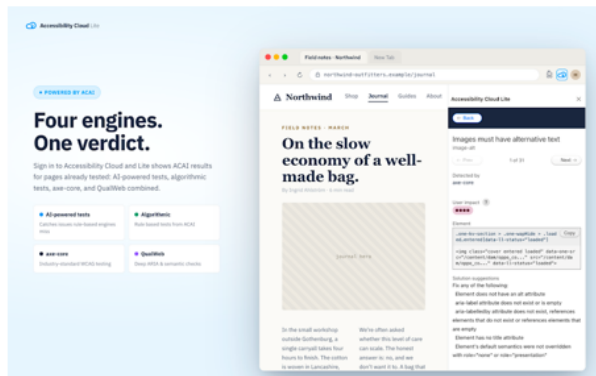
Du kan med fordel også slå fanen "Tilgængelighed" til, hvis du kan finde den.

Vælg et element og se i vinduet "Beregnete egenskaber" eller "Computed Properties". Hvis Tilgængelighedstræet kan formidle noget om det element du har valgt, kommer informationerne ud der.

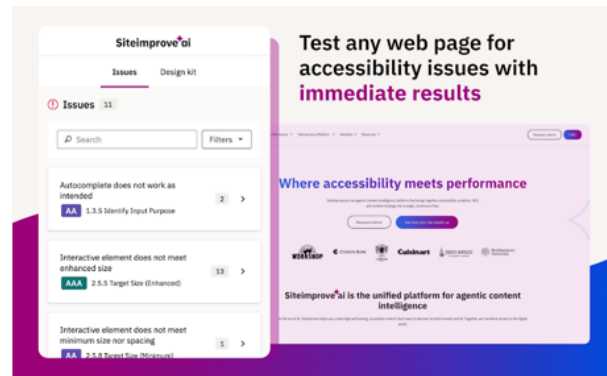
På billedet kan vi for eksempel se, at vi kigger på en overskrift, der er mærket op med H2 og hedder "information fra biblioteket i Byrum".

## 24: Værktøjer, som kan teste en hel side ad gangen

### Værktøjer, som kan teste en hel side ad gangen



Accessibility Cloud Lite



Siteimprove Accessibility Checker

I stedet for at teste en enkelt funktion ad gangen, så findes der værktøjer, som kan tage en hel side i et hug og give pålidelige informationer om mulige fejl.

Her er to af slagsen. Der findes ret mange på markedet, men jeg har valgt disse to ud, Accessibility Cloud Lite og Siteimprove Accessibility Checker, fordi jeg kender dem lidt og bruger dem ind imellem.

Fælles for den type værktøjer er, at de skal anvendes med forbehold: Semi-automatiske testværktøjer, som disse, kan ikke teste alle 50 WCAG 2.1-kriterier.

De kan teste en god tredjedel. Resten er man nødt til at undersøge manuelt.

For eksempel kan de godt afgøre, om der er en alt-tekst til et billede. Men de kan ikke sige noget om kvaliteten af alt-teksten. De kan advare om mulige kontrastfejl, men kan ikke altid afgøre det med sikkerhed.

De kan også se, om en side har en overskrift, men ikke om overskriften fortæller noget om sidens emne eller funktion. Og de har ingen mening om undertekster eller synstolkning til video og lyd.

Det gør dem ikke til dårlige værktøjer, det er bare en naturlig begrænsning.