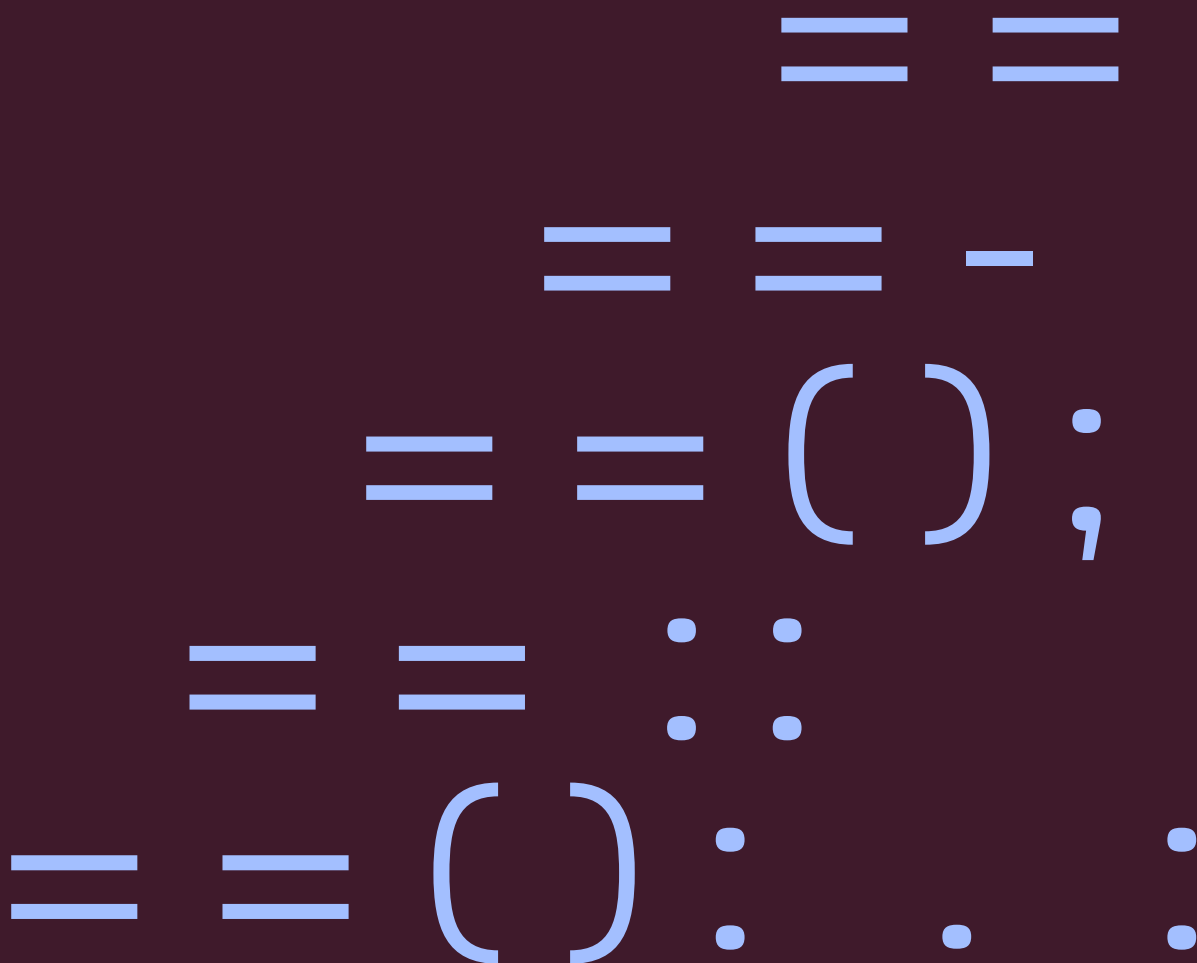


Digital Post

Principper for afsendelse af dokumenter og HTML v1.2



Indhold

1. Introduktion	4
1.1 Målgruppe	4
1.2 Vejledningens indhold	4
2. Generelle principper for afsendelse af meddelelser i Digital Post.....	4
2.1 Meddelelser skal være selvbærende – 'self-contained'	5
2.2 Sørg for bedst mulig tilgængelighed og funktionalitet.....	5
2.3 Retningslinjer for kodning af meddelelser skal overholdes	5
2.4 Sikring af ensartet udtryk på tværs af platforme.....	6
2.5 Overholdelse af regler til udformning af MeMo.....	6
2.6 Anvendt funktionalitet må ikke manipulere visning	6
2.7 Modtagers adfærd må ikke spores eller registreres	6
2.8 Bevar kontrol over websites, hvortil der linkes og beskyt forbindelsen	7
2.9 Materiale af blivende værdi vedhæftes i PDF-format.....	7
3. Kort om HTML i Digital Post	7
3.1 Validering af HTML-meddelelser	8
4. Særlige retningslinjer for brug af HTML	9
4.1 Uafhængighed af eksterne ressourcer	9
4.2 Anvendelse af CSS.....	9
4.3 JavaScript må ikke anvendes	9
4.4 Indlejrede objekter må ikke anvendes	9
4.5 Om hyperlinks – links til eksterne websteder	10
4.6 WCAG - Web Content Accessibility Guidelines 2.1.....	11
4.7 Om billeder i meddelelser	11
4.8 Skrifttyper/Typografi.....	12
4.9 Semantisk opmærkning.....	13
5. Særligt om brug af PDF	13

6. Appendiks 1 – Uddybende vejledning til brug af HTML	13
6.1 Billeder	13
6.2 Links	14
6.3 Tabeller.....	15
7. Appendiks 2 – WCAG	16
8. Appendiks 3 – WAI-ARIA.....	17
9. Appendiks 4 – HTML Whitelist.....	18

1. Introduktion

Denne vejledning introducerer en række principper, som skal overholdes af afsendere, når de sender dokumenter med Digital Post. Principperne er fastsat med henblik på den bedste oplevelse af tilgængelighed, robusthed og sikkerhed. Vejledningen har derudover særlig fokus på regler og retningslinjer for kodningen af HTML dokumenter, når disse medsendes i Digital Post meddelelser.

1.1 Målgruppe

Denne vejledning retter sig mod myndigheder og it-virksomheder, som udformer løsninger for afsendelse af Digital Post. Både den forretningsansvarlige og den teknisk ansvarlige vil finde nyttige råd og regler.

1.2 Vejledningens indhold

Afsnit 2 indeholder en beskrivelse af de generelle principper, som gælder for afsendelse af dokumenter ved brug af Digital Post.

Afsnit 3 er en kort introduktion til fordele og udfordringer ved brug af HTML baserede dokumenter i Digital Post.

Afsnit 4 indeholder en gennemgang af de retningslinjer, som gælder ved forsendelse af HTML baserede dokumenter i Digital Post.

Afsnit 5 beskriver de overvejelser man bør gøre sig i relation til brug af PDF baserede dokumenter, særligt i forhold til tilgængelighed.

Appendiks 1 indeholder uddybende vejledning om brug af HTML i relation til Digital Post.

Appendiks 2 beskriver WCAG på et overordnet niveau.

Appendiks 3 giver en kort introduktion til WAI-ARIA.

Appendiks 4 indeholder en HTML whitelist tabel. En oversigt over alle tilladte HTML tags, attributter og værdier der tillades i meddelelser.

2. Generelle principper for afsendelse af meddelelser i Digital Post

Modtagere af Digital Post har høje forventninger til sikkerhed, integritet og robusthed af meddelelserne. Derfor er der defineret en række principper, som afsendere skal respektere, når de sender dokumenter til modtagelse via Digital Post. Generelt er det altid afsenders ansvar at sikre, at indsendte dokumenter er tilgængelige og ikke påfører hverken Digital Post eller slutmodtager sikkerhedsmæssige risici.

2.1 Meddelelser skal være selvbærende – 'self-contained'

Meddelelser skal være uafhængige af eksterne ressourcer og understøttende infrastruktur i deres udformning, visning og funktion. Tolkning af indholdet af meddelelser må ikke være afhængige af ressourcer fra eksterne kilder via brug af links. Således skal eksempelvis billeder, grafik, typografi, formularer og skabeloner være indeholdt i dokumenterne ved afsendelsen.

Baggrunden herfor er dels at sikre en robust og konsistent visning for modtageren, dels at øge sikkerheden i løsningen, således den ikke åbnes for eventuelle kompromitterede websites eller sårbarheder i tredjeparts software.

2.2 Sørg for bedst mulig tilgængelighed og funktionalitet

Det påhviler afsenderen at sikre, at meddelelser kan læses af modtageren. Dette betyder dels, at meddelelser skal indkodes ud fra gængse standarder, så de kan læses på flest mulige fysiske enheder og browsere og dels sikre den størst mulige tilgængelighed for modtagere med læsevanskeligheder, handicap osv. Det gælder også modtagere som anvender skærmlæsere eller andre hjælpemidler. Specifikt skal afsendere respektere WCAG 2.1 – se mere herom i følgende afsnit og Appendiks 2. Tilsvarende gælder for websteder, hvortil der linkes i meddelelserne. Tilgængelighed skal overholde aktuelle lovkrav¹.

I praksis er det erfaringen, at en stigende kompleksitet i visningen af meddelelser kan medføre en faldende grad af tilgængelighed. Tilgængeligheden af konkrete eksempler på kodning kan testes med kommercielt tilgængelige værktøjer, hvor det er muligt at teste HTML på en lang række renderingsmotorer. PDF giver ligeledes anledning til en række overvejelser man bør gøre sig i relation til tilgængelighed, hvilket fremgår af afsnit 5 i dette dokument. Baggrunden herfor er at løsningen skal efterlade et indtryk af god funktionalitet og kvalitet hos modtageren, samt at løsningens tilgængelighed tilgodeser befolkningens diversitet.

2.3 Retningslinjer for kodning af meddelelser skal overholdes

Det påhviler afsenderen at overholde retningslinjerne til kodning af de enkelte meddelelser angivet i dette dokument. Eksempelvis ved valg af tegnsæt, indkodning af billeder, samt fravalg af eksempelvis JavaScript.

Baggrunden er at sikre læsbarheden på det størst mulige udvalg af fysiske enheder og klienter, og at oppebære et højt sikkerhedsniveau således, at hverken modtageren eller løsningen eksponeres for sikkerhedsrisici.

¹ Læs Lov nr. 692 af 08/06/2018 "Lov om tilgængelighed af offentlige organers websteder og mobilapplikationer" på adressen <https://www.retsinformation.dk/eli/Lta/2018/692>

2.4 Sikring af ensartet udtryk på tværs af platforme

Det påhviler afsenderen at meddelelser fremstår og fungerer med et ensartet udtryk på tværs af platforme. Der findes tilgængelige værktøjer for emulering af diverse elektroniske enheder, softwareprodukter og versioner, som kan bidrage til at afdække, hvordan en konkret meddelelse/skabelon fremstår og fungerer.

Ved brug af HTML i meddelelserne skal HTML5-standarden overholdes, således at meddelelserne er valide og semantisk korrekte – såkaldt 'well-formed'. Herved sikres at modtageren får præsenteret indholdet korrekt.

2.5 Overholdelse af regler til udformning af MeMo

Der findes særlige regler for, hvilke elementer MeMo må indeholde, samt deres datatyper. MeMo-lib² kan anvendes til at generere og validere MeMo-filer, hvilket hjælper med at sikre, at filerne både strukturelt og indholdsmæssigt lever op til de fastsatte regler for udformningen af filer.

2.6 Anvendt funktionalitet må ikke manipulere visning

Det er ikke tilladt at indlejre funktionalitet i meddelelsen, som manipulerer eller ændrer modtagerens brugerflade/visningsklient. Desuden skal afsenderen respektere integriteten i de klienter, som modtagerne anvender til at læse meddelelserne med. Afsenderen må således ikke anvende 'overlays', specificere faste størrelser på grafik eller billeder eller gennemtvinge overordnede design-restriktioner.

Afsenderen bør i stedet anvende procentvise størrelsesangivelser for grafik og billeder og tilstræbe, at meddelelser præsenteres i et format, som modtageren genkender som en digital meddelelse eller et klassisk brevformat. Baggrunden herfor er at sikre korrekt visning af meddelelser på tværs af forskellige visningsklienter, blandt andet således at visningsklienten dynamisk vil kunne tilpasse meddelelsen til den aktuelle enhed, som meddelelsen skal vises på.

Det forventes, at afsender grundigt overvejer, om medsendte elementer kan skabe utilsigtede konsekvenser, når modtagerens renderingsmotor fortolker meddelelsen.

2.7 Modtagers adfærd må ikke spores eller registreres

Det er ikke tilladt for afsender på nogen måde at spore modtagers aktivitet i forbindelse med modtagelsen af en meddelelse. Dette kunne eksempelvis have form som sporing af åbning, læsning eller sletning af post, fremstilling af statistik eller lignende.

Det er eksempelvis ikke tilladt at anvende links med unikke id-parametre med den hensigt at kunne registrere, om en specifik modtager har læst meddelelsen.

² MeMo-lib er et hjælpeværktøj, som stilles til rådighed af Digitaliseringsstyrelsen med henblik at standardisere og lette arbejdet med anvendelsen af MeMo. Både kildekode og kompilerede versioner af biblioteket findes på adressen <https://bitbucket.org/nc-dp/workspace/projects/DPPUB>

2.8 Bevar kontrol over websites, hvortil der linkes og beskyt forbindelsen

Afsendere af meddelelser i Digital Post kan indsætte links til eksterne websteder, f.eks. relevante selvbetjeningsløsninger. Dette forudsætter, at afsenderen har kontrol over disse websteders indhold og at forbindelsen til dem er sikker.

Hertil kommer, at afsenderen i en rimelig fremtid skal bevare kontrollen, så modtageren altid vil møde et meningsfyldt og sikkert indhold, når modtageren senere aktiverer links i tidligere modtagne meddelelser. Som udgangspunkt består pligten så længe, som nogen modtager kan forventes at aktivere links i ældre meddelelser.

Det anbefales, at afsender opsætter foranstaltninger for at sikre, at modtageren ikke bliver henvist til et inaktivt eller ukorrekt domæne, som siden afsendelse, er blevet overtaget af tredjeparter. Baggrunden herfor er målet om at give modtageren en sikker, konsistent og robust funktionalitet i en forudsebar fremtid. Specifikt skal modtageren beskyttes mod, at ondsindede aktører overtager domæner, som ikke længere er registreret til den tilsigtede brug og herefter udgør en sikkerhedsrisiko for besøgende på webstedet.

2.9 Materiale af blivende værdi vedhæftes i PDF-format

Meddelelser, der indeholder materiale af blivende værdi, bør vedhæftes som PDF. Da det kan være udfordrende at sikre, at alle visningsklienter over tid kan gengive HTML-meddelelser ensartet, anbefales det at vedhæfte meddelelsen i et alternativt format – for eksempel som klartekst eller PDF – for at sikre langtidsholdbarhed og konsistens.

Såfremt der er tale om et juridisk bindende dokument, f.eks. en afgørelse, anbefales det alene at sende dokumentet i PDF/A, idet der er større sikkerhed for, at visningen er ens uagtet modtagersystem og version af browser. Det er afsenders ansvar at sikre at afsendte Digital Post meddelelser indeholder al information, som modtager skal forholde sig til.

3. Kort om HTML i Digital Post

HTML er et af de godkendte filformater til afsendelse af meddelelser via Digital Post. Formatet tilbyder en række fordele, men indebærer også visse udfordringer.

I forbindelse med udformning af HTML-meddelelser der sendes via Digital Post, er det væsentligt at afsender er opmærksom på, at modtagere kan falde i to grupper – dem der læser deres post via en visningsklient f.eks. på borger.dk eller Virk og dem der henter posten hjem i egne systemer.

Visningsklient – En mobilapplikation eller en webløsning, som blandt andet viser modtagne og sendte meddelelser. Visningsklienterne, f.eks. borger.dk og Virk, er udviklet specifikt til at præsentere Digital Post og anvendes af alle borgere og virksomheder, der ikke har valgt at opsætte et modtagersystem. Disse visningsklienter vil fortolke og forestå rendering af HTML medsendt i MeMo.

Modtagersystem – En applikation installeret på modtagerens enhed – typisk pc, men potentielt også mobilenheder – som eksempelvis henter, sender og opbevarer meddelelser. Myndigheder og andre

organisationer, som i regi af Digital Post har status af offentlige afsender, er forpligtet til at anvende et modtagersystem til at modtage meddelelser. Det er også muligt for øvrige organisationer at opsætte et modtagersystem. Fortolkning og rendering af HTML afhænger af det enkelte modtager- eller fagsystem, og det må derfor forventes at der kan være varians i visningen.

Sammenlignet med de øvrige alternativer, såsom PDF eller tekstdokumenter, har HTML flere fordele, såsom god understøttelse af tilgængelighed for modtagere med specielle behov, større fleksibilitet samt større ensartethed i kommunikationen over forskellige kanaler. Derudover tilbyder HTML en vifte af avancerede funktioner til præsentation, som ligger ud over de faste formater, såsom grafikker, farver og lignende.

Meddelelser i HTML kan medvirke til at sikre en øget tilgængelighed for modtagere, som anvender hjælpemidler til læsning af meddelelserne. Skærmlæsere kan under anvendelse af de indbyggede opmærkninger hjælpe læseren til at identificere de enkelte komponenter i meddelelsen – på en måde, som er skjult for de modtagere, som ikke anvender hjælpemidler. Samtidig kan billeder, grafik, tabeller og specielle skrifttyper indlejres i HTML, som dermed kan tilsikre en ensartet præsentation på tværs af platforme.

Anvendelse af HTML kan dog også byde på vanskeligheder for post-modtagere, der anvender egne modtagersystemer til at læse deres Digital Post. Meddelelserne vil her blive præsenteret ved brug af egenudviklet eller proprietær programmel, som kan være konfigureret til ikke at render og vise HTML-indhold helt eller delvist af hensyn til sikkerhed eller lokal it-politik.

HTML er et fleksibelt opmærkningssprog, hvor indholdet opmærkes med information, som beskriver indholdet. Når indholdet skal vises, vil opmærkningen blive fortolket af en klient på modtagerens enhed.

Da Digital Post når ud til alle borgere og virksomheder i Danmark, vil der være rigtig mange forskellige varianter og konfigurationer af visningsklienter, mobilapplikationer, browsere og fagsystemer, det medfører risiko for, at meddelelserne vises forskelligt på forskellige enheder og i forskellige versioner af f.eks. browsere. Dette er en kendt risiko – og til afhjælpning heraf findes en række emulatorer, som kan illustrere hvordan konkrete HTML-filer præsenteres på forskellige enheder og browsere.

3.1 Validering af HTML-meddelelser

Der findes specifikke HTML tags og attributter der kan forårsage uønskede handlinger hos læseren, eller kompromittere sikkerheden for læseren, fordi HTML afvikles direkte i læserens browser. I Digital Post-løsningen valideres HTML-meddelelser derfor med udgangspunkt i en såkaldt "whitelist" over tags. Whitelisten er en tabel over alle de HTML tags og attributter, der tillades i en meddelelse. Tabellen kan læses i **Fejl! Henvisningskilde ikke fundet..**

Meddelelser der indeholder tags eller attributter, som ikke fremgår af whitelisten, vil blive afvist. Afsenderen vil få en fejlmeddelelse, der angiver årsagen til afvisningen. Det er desuden muligt at

afprøve HTML-valideringen ved at anvende den dedikerede valideringsservice udstillet af Digital Post til formålet.³

4. Særlige retningslinjer for brug af HTML

Hensyn til sikkerhed, brugeroplevelse, robusthed m.v. tilsiger, at afsenderen er forpligtet at respektere følgende retningslinjer, når de sender dokumenter i HTML-format. Retningslinjerne følger de generelle principper for afsendere af Digital Post, som anført i forrige afsnit.

I Appendiks 1-3 findes uddybende råd og vejledning til de anførte retningslinjer.

4.1 Uafhængighed af eksterne ressourcer

Meddelelser skal være uafhængige af eksterne ressourcer og må ikke i deres visning eller funktion være afhængige af eksempelvis eksterne billeder, typografi eller CSS-stylesheets.

Baggrunden herfor er dels at sikre en robust og konsistent visning for modtageren, dels at øge sikkerheden i løsningen, således den ikke åbnes for eventuelle kompromitterede websites eller eksterne plug-ins.

4.2 Anvendelse af CSS

Det er tilladt at benytte CSS for at "style" meddelelsen, medmindre man falder under *Strict* policy, som beskrevet i Appendiks 4 – det skal blot være inline, enten i et `<style>`-tag eller direkte på de individuelle elementer. Der må ikke specificeres absolutte størrelser, som enkelte klienter ikke kan håndtere – eksempelvis `'width: 5000px'` på en lille skærm. Ligeledes er det ikke tilladt at anvende `'!important'`-deklarationer. Baggrunden herfor er et ønske om at respektere visningsklientens muligheder for frit at give den bedst mulige præsentation af meddelelsen, uden at den bliver domineret eller underkendt af afsenderen.

Generelt gælder det, at anvendelsen af CSS skal ske i respekt for diversiteten af klienter således, at afsenderen respekterer klientens frie fortolkning af indhold. Desuden bør CSS anvendes med det forbehold, at meddelelser kan vises i browsere eller systemer, som ikke understøtter den konkrete funktionalitet. Dette indebærer også at CSS alene bør anvendes til grafiske formål, og ikke til funktionalitet.

4.3 JavaScript må ikke anvendes

JavaScript må ikke anvendes fra hverken en ekstern fil, eller via et inline script.

Baggrunden herfor er ønsket om at sikre en god og ensartet brugeroplevelse, idet flere modtagere forventeligt har valgt at slå JavaScript fra af sikkerhedshensyn.

4.4 Indlejrede objekter må ikke anvendes

Indlejrede objekter, som udfører eksekvering af kode og/eller tager kontakt til eksterne websites, er ikke tilladt. Dette kunne eksempelvis være i form af IFrame (Inline Frame), Java appletter/applets, eller Adobe Flash.

³ Servicen findes på stien `"/validations/"` og kan modtage og validere en enkelt HTML-fil via REST.

Baggrunden herfor er kravet til modtagerens platform, hvor der ofte kræves installation af tredjeparts software for at indlejrede objekter kan afvikles som tilsigtet. Dertil kommer sikkerhedsrisikoen ved afvikling af tredjeparts komponenter (med eventuelle sårbarheder uden for afsenders kontrol) på modtagerens klient.

Video og lyd må ikke anvendes

Video eller lyd er ikke tilladt – og må derfor ikke indlejres – i meddelelser. Såfremt der er behov for det, kan video- og lydfiler i stedet fremsendes som vedhæftninger. Baggrunden herfor er, at en række klienter må forventes ikke at være i stand til at præsentere meddelelsen som tilsigtet.

Billeder

Billeder må ikke anvendes som links, da det er svært se, at et billede faktisk er et klikbart link. Ligeledes kan det være svært at se, om et billedlink allerede har været klikket på.

4.5 Om hyperlinks – links til eksterne websteder

Hyperlinks kan anvendes i meddelelser som adgangsveje til eksterne websteder, eksempelvis selvbetjeningsløsninger. Hyperlinks skal anvende HTTPS-, må ikke fremstå som URL-forkortere, men skal tydeligt fremstå som en direkte vej til et separat, eksternt website.

Afsender skal bevare kontrollen over det website, der linkes til. Baggrunden for dette er af hensyn til dels brugervenlighed, dels sikkerhed. Hvis en modtager aktiverer et link i en mail modtaget langt tilbage i tiden, vil modtageren risikere enten at besøge et ikke-funktionelt website, eller en ondsindet aktør vil have kunnet overtaget eller registreret websitets adresse, og dermed præsentere ondsindet indhold på websitet, som kunne hentes ved modtagerens besøg.

Det skal herudover bemærkes, at der ikke bør inkluderes personhenførbare data – eksempelvis CPR/CVR-nummer, telefonnummer eller navn – i selve linket. Dette anbefales for at sikre, at personhenførbare oplysninger ikke bliver givet til tredjeparter. Dette kunne f.eks. ske, hvis afsenders domæne er forældet og en tredjepart har overtaget det. Man kan i stedet tildele modtager en unik nøgle, som afsender kan benytte til at referere til en given modtager.

Det anbefales, at afsender har sikret sit websteds domæne med DNSSEC eller tilsvarende, samt implementeret former for viderestillingsinformation for brugeren i tilfælde af inaktivt domæne. Dette kan være en viderestillingsservice, som håndterer domæner og ændringer. Alternativt kan afsender opsætte HTTP 302 statuskoder, som informerer modtager om, hvor informationen ellers kan findes. For uddybende vejledning omkring brug af links henvises til Appendix 1 sektion 6.2 Links.

Det anbefales ikke at anvende formularer indeholdt i meddelelser med henblik på modtagers udfyldelse af denne. I stedet bør afsenderen i meddelelsen inkludere et link til et eksternt website, hvor formular-funktionaliteten kan placeres. Som hovedregel bør det eksterne website ligge bag et sikkert login, jævnfør reglerne om persondata i databeskyttelsesforordningen, især hvor webstedet håndterer persondata. Hertil skal afsenderen iagttage retningslinjerne om hyperlinks, som angivet andetsteds i denne vejledning.

Selvom afsendelsen af formularen er krypteret er det stadigvæk ikke tilladt at afkræve modtageren at medsende persondata, såsom modtager identitet eller andre informationer som kan spores tilbage til modtageren.

4.6 WCAG - Web Content Accessibility Guidelines 2.1

WCAG⁴ er et sæt retningslinjer for tilgængeligt webindhold, der hjælper med at gøre det mere tilgængeligt for mennesker med forskellige former for funktionsnedsættelser. Retningslinjerne er bygget omkring fire hovedprincipper: opfattelig, anvendelig, forståelig og robust. Retningslinjerne er blevet til i et internationalt samarbejde, som blandt andet inkluderer Danske Handicaporganisationer.

Afsender er forpligtet til at tilstræbe en høj tilgængelighed, som nævnt andetsteds i denne vejledning. WCAG er i denne henseende en anbefaling, som kan hjælpe en afsender med at udvikle efter nogle vedtagne og meget bredt vedtagne principper.

For yderligere uddybning, se **Fejl! Henvisningskilde ikke fundet..**

WAI-ARIA – Web Accessibility Initiative-Accessible Rich Internet Applications

WAI-ARIA er en teknisk specifikation udgivet af World Wide Consortium⁵. WAI-ARIA specificerer, hvordan man øger tilgængeligheden af websider, især med henblik på personer med læsevanskeligheder og synsbesvær.

WAI-ARIA er udviklet for at tilføje mere kontekstuel information til HTML, så brugere af skærm-læsere lettere kan forstå opmærkningen og vurdere, hvad der er mest relevant. Det fungerer som et supplement til HTML og bør kun anvendes, når HTML's egne opmærkningsmuligheder ikke er tilstrækkelige. Afsendere og udviklere bør derfor først fokusere på korrekt og meningsfuld opmærkning i HTML og derefter benytte WAI-ARIA, hvis der er behov for yderligere detaljer i opmærkningen.

Det er ikke et direkte krav at følge WAI-ARIA-specifikationen i Digital Post, men den er et effektivt værktøj til at opfylde WCAG-retningslinjerne. Derfor anbefales det at sætte sig ind i standarden og implementere de dele, der er relevante og mulige.

For yderligere information se **Fejl! Henvisningskilde ikke fundet..**

4.7 Om billeder i meddelelser

Billeder i formaterne JPEG, GIF eller PNG kan indlejres ved Base64-enkodning. Billeder bør optimeres og størrelsesjusteres til webvisning. Billeder, grafik og typografi kan generelt inkluderes i meddelelsen via Base64-enkodning.

Billeder skal have tilknyttet en 'alt-tag' og indeholde en beskrivende tekst. Teksten anvendes – eksempelvis af skærm-læsere – hvis billedet ikke kan vises. Billeder, som udelukkende benyttes til udsmykning, skal ikke bære et 'alt-tag'. Hvor et modtagersystem anvendes til læsning af en meddelelse, kan det forekomme, at billeder – tillige med skrifttyper og grafik – ikke præsenteres som tilsigtet. Billeder bør derfor ikke være et bærende element i meddelelsen.

⁴ WCAG 2.1 er tilgængelig på adressen: <https://www.w3.org/TR/WCAG21/>

⁵ Se mere på adressen <https://www.w3.org/WAI/standards-guidelines/aria/>

Det er også en mulighed at vedhæfte et billede i MeMo og referere til det i HTML med et Content-ID (cid). Dette gør det muligt at genbruge billeder på tværs af HTML-filer, hvilket f.eks. kan være nyttigt ved understøttelse af flere sprog, da det vil mindske størrelsen på meddelelsen.

For uddybende vejledning henvises til sektion 6.1 **Fejl! Henvisningskilde ikke fundet.** i **Fejl! Henvisningskilde ikke fundet..**

4.8 Skrifttyper/Typografi

Anvendte skrifttyper skal Base64-kodes og tilføjes i CSS. De må ikke hentes eksternt.

Tegnsæt

Der skal altid benyttes UTF-8, som herved sikrer at danske, såvel som alle andre Unicode-tegn, vises korrekt.

Validt og well-formed HTML

HTML-dokumenter skal alle være valide og 'well-formed'. Et validt HTML-dokument deklarereres med en DOCTYPE og indeholder altid korrekte metadata, såsom sidetitel og beskrivelse.

HTML5 og CSS3

Meddelelser skal være kodet i HTML5, eventuelt suppleret med CSS3, dog skal de begrænsninger og retningslinjer, der fremgår af dette dokument, tages med i betragtning. Visningsklienter skal som udgangspunkt understøtte HTML5 og CSS3.

Ikke alle modtagersystemer kan forventes at håndtere al HTML5-opmærkning korrekt, enten på grund af systemvalg eller konfiguration hos modtageren. Derfor er det vigtigt at skrive CSS3 på en progressiv måde. CSS3 kan bruges til at forbedre layout og visuelle elementer, men der bør altid være et brugbart fallback i CSS2.1. For eksempel kan man anvende et farveforløb som baggrund på et element, men hvis dette ikke understøttes, bør der være en alternativ løsning som en ensfarvet baggrund.

CSS skal skrives i et style-tag, der allerhelst er placeret i sidens **<head>**, og må ikke henvise til en ekstern fil. Det er også tilladt at *style* direkte på det relevante element. Dette er ikke gældende for *Strict* policy, som beskrevet i Appendix 4.

Tabeller

Rene data kan med fordel præsenteres i en tabel, da tabeller ofte giver et klart og struktureret overblik. Dog kan tabeller også være pladskrævende og udfordrende at vise på små skærme, især hvis de indeholder lange tekster eller store mængder tal. For svagtseende kan HTML-tabeller være vanskelige at tolke og dermed svære at formidle på en måde, der er let at forstå for modtageren.

For uddybende vejledning omkring brug af tabeller henvises til sektion 6.3 Tabeller i **Fejl! Henvisningskilde ikke fundet.**

4.9 Semantisk opmærkning

Det bør altid tilstræbes at anvende semantisk opmærkning, da det hjælper brugere, herunder svagtseende, der benytter skærmlæsere. Semantisk opmærkning gør det muligt for skærmlæsere at identificere typen af information, hvilket sikrer en mere konsistent og forståelig præsentation for modtageren.

Ved at bruge semantisk opmærkning kan dokumentets indhold tolkes og præsenteres korrekt, selv uden styling, hvilket gør det mere tilgængeligt og robust.

Eksempelvis foretrækkes det at bruge et `<b>` eller `<strong>`-element for at gøre tekst fed, i stedet for `<p class="bold">`.

Et andet eksempel er at benytte `<address>`-elementet til at opmærke en adresse på denne vis:

`<address>`

Digitaliseringsstyrelsen

Landgreven 4,

1301 København K

`</address>`

Det er vigtigt altid at benytte retvisende elementer. Det kan være fristende at bruge button-elementet, hvis man ønsker, at et link skal ligne en knap. Men et link og en knap er to forskellige elementer, som hver har deres tiltænkte funktionalitet og semantiske mening. Her vil en skærmlæser risikere at identificere link-knappen, som en knap, og dermed give misvisende information til modtageren.

5. Særligt om brug af PDF

Ved brug af PDF er der flere faktorer, som afsender bør overveje, før afsendelsen.

For det første er tilgængelighed en vigtig problematik, da modtagere med f.eks. nedsat syn kan benytte skærmlæsere og andre tilgængelighedsværktøjer til at tolke meddelelsen.

For det andet skal PDF-filer kunne vises korrekt på alle platforme. Heldigvis sikrer den underliggende struktur, som gør det muligt for skærmlæsere at fortolke indholdet, også, at mobil enheder kan tilpasse PDF'ens indhold til mindre skærme.

Derudover skal lov om webtilgængelighed overholdes i udformning af PDF-dokumenter. Der kan læses mere om dette her: <https://digst.dk/digital-inklusion/webtilgaengelighed/vejledning/>.

6. Appendiks 1 – Uddybende vejledning til brug af HTML

I dette appendiks findes uddybende råd og vejledning til de anførte retningslinjer for brug af HTML-dokumenter i Digital Post – der er særlig fokus på billeder, links og brug af tabeller.

6.1 Billeder

Billeder kan indlejres Base64-enkodet direkte i HTML eller via CSS:

Indlejret Base-64

```

```

Et billede skal være Base64-enkodet, og må ikke hentes fra en ekstern kilde, og det skal altid have en `alt`-attribut, som forklarer så godt som muligt for svagtseende, hvad billedet forestiller. Denne tekst vil tillige blive vist, hvis modtagerens klient/konfiguration ikke kan vise billedet. Billeder, som udelukkende benyttes som udsmykning, skal ikke bære et 'alt-tag'. Man kan tilføje en ARIA `role="presentation"`, hvis grafikken ikke er relevant for indholdet, men kun for den visuelle præsentation.

Responsive billeder

Hvor der er behov for at vise forskellige billeder afhængig af skærmstørrelsen, kan det være nyttigt at anvende responsive billeder. Dette er muligt ved at angive nogle regler ved brug af `<picture>`- og `<source>`-elementer.

```
<picture>
  <source media="(max-width: 799px)"
    srcset="data:img/png;base64,i4€G...Jggg==">
  <source media="(max-width: 800px)"
    srcset="data:img/png;base64,iDFD...Jggg==">
  
</picture>
```

Herover ligger to Base64-billeder i hvert sit `<source>`-element. Begge elementer har også en `media`-attribut, hvor man kan indtaste et hvilket som helst validt media query. Parseren vil således vise det billede, som matcher bedst. Er der ingen, der matcher – eller hvis `<source>`-elementet ikke er understøttet – benyttes det billede, som vises i det almindelige `<img>`-element.

Det skal dog bemærkes, at da der anvendes indlejret Base64 (frem for eksterne ressourcer), vil al billededata – i samtlige varianter – blive hentet ned sammen med meddelelsen. Af hensyn til den samlede filstørrelse af meddelelsen, skal man derfor være varsom med at tilføje for mange billeder.

6.2 Links

Som nævnt i retningslinjer i afsnittet om hyperlinks, skal alle links af sikkerhedsmæssige årsager pege på et HTTPS-domæne.

```
<a href="https://digst.dk" target="_blank" aria-label="Gå til forsiden">
  Digitaliseringsstyrelsen
</a>
```

Det skal tydeligt fremgå for modtageren, at der er tale om et link – på en måde, hvor linket adskiller sig ud fra den øvrige tekst. Typisk har et link en anden skriftfarve og er understreget. Af samme årsag skal normal tekst heller ikke forveksles med et link. Almindelig tekst bør derfor ikke understreges eller markeres med skiftende skriftfarver. I stedet kan man fremhæve tekst ved eksempelvis at gøre den fed.

Et link kan have forskellige statusser, og disse bør også styles særskilt for at indikere, at linket er klik-bart. Dette kan gøres på flere måder, eksempelvis at der vises en pegefinger, når musen føres over linket, at der er et omrids rundt om linket, når det er aktiveret med keyboardet, eller har en

anden farve, når det allerede er blevet klikket på. Dette er standard i et normalt browser-stylesheet, så forholdet kræver alene opmærksomhed, hvis man har lavet ændringer til styling af links. Link-teksten skal fortælle modtageren, hvad linket drejer sig om. Anfør nøgleord og en beskrivelse, der giver et bud på hvad der sker på den side, som modtageren føres over til. Linket må gerne flettes ind i teksten organisk, når dette er muligt og det giver mening i den givne kontekst.

Det anbefales at bruge linkets `title`-attribut. Den kan enten bruges til at give modtageren en lidt mere uddybende beskrivelse af, hvad der vil ske, når der klikkes på linket. Alternativt kan man blot nøjes med at skrive linkets titel eller hostnavn. Dette kan dog undlades, hvis den ikke vil gavne brugeroplevelsen eller hvis det allerede er åbenlyst for modtageren, hvor linket peger hen.

Man kan anvende en `aria-label`, hvis man vil give kontekstuel information specifikt til skærmlæsere. I dette tilfælde skal man være opmærksom på, at de fleste skærmlæsere vil læse værdien i `aria-label` attributten – og ikke linkteksten – højt for modtageren.

Endeligt skal alle links have `target="_blank"`. Den betyder, at alle links vil blive åbnet i en ny fane, således at modtageren ikke bliver sendt væk fra den oprindelige meddelelse.

6.3 Tabeller

Rene data kan præsenteres i en tabel. Tabeller kan være en overskuelig måde at præsentere data på. Tabeller kan dog også være pladskrævende og dertil vanskelige at præsentere på fysisk små skærme, specielt hvor tabellen indeholder længere tekster.

For svagtseende kan HTML-tabeller være vanskelige at fortolke og dermed præsentere for modtageren i et forståeligt format. Eksempel:

```
<table>
  <caption>Byer i Danmark</caption>
  <tr>
    <th></th>
    <th>Indbyggertal (by)</th>
    <th>Indbyggertal (kommune)</th>
  </tr>
  <tr>
    <td>København</td>
    <td>613.288</td>
    <td>1.308.893</td>
  </tr>
  <tr>
    <td>Århus</td>
    <td>340.421</td>
    <td>273.077</td>
  </tr>
  <tfoot>
    <td>Samlet</td>
    <td>953.709</td>
    <td>1.581.970</td>
```

```
</tfoot>
</table>
```

Ovenstående kode, vil producere følgende tabel:

Byer i Danmark		
	Indbyggertal (by)	Indbyggertal (kommune)
København	613.288	1.308.893
Århus	340.421	273.077
Samlet	953.709	1.581.970

Layoutet i en tabel består ganske simpelt af rækker (<tr>), der kan indeholde celler (<td>). Der skal helst være lige mange celler i en række, dog har man mulighed for at undlade celler eller slå to celler sammen med en colspan-attribut. Det skaber dog problemer for skærmlæsere, der ikke altid har nemt ved at læse tabeller i et uortodokst format.

Tabeloverskrifter kan laves ved at indsætte <th> eller <thead> i stedet for <td>. Talsummer kan vises ved at skabe en 'table-footer', som samler flere rækker i <tfoot>.

Tabeller har desværre nogle ulemper, som kan være svære at arbejde udenom. I sagens natur, kan de være meget pladskrævende og ufleksible. Det giver nogle udfordringer, når de præsenteres på en mindre skærm. En nem løsning er, at man sørger for at tabellen har mulighed for at udvide sig i bredden så meget som nødvendigt og gøres scrollbar horisontalt. På den måde er man ikke begrænset af skærmbredden.

En mere sofistikeret løsning er, at man lader layoutet tilpasse sig skærmbredden, således at man har én visning på en stor skærm, og en anden visning på en mobiltelefon, som illustreret.

Byer i Danmark

København

– Indbyggertal (by):	613.288
– Indbyggertal (kommune):	1.308.893

Århus

– Indbyggertal (by):	340.421
– Indbyggertal (kommune):	273.077

Man kan også vælge, at vise færre kolonner, hvis pladsen ikke er tilgængelig og hvis man kan undvære dem.

7. Appendiks 2 – WCAG

Web Content Accessibility Guidelines (WCAG) er en række retningslinjer, der skal gøre indholdet tilgængeligt for modtagere med særlige behov eller handicap. Retningslinjerne er kategoriseret i fire hovedprincipper: Opfattelig, anvendelig, forståelig og robust.

Opfattelig

Det første princip i WCAG er, at indholdet skal præsenteres på en måde, som modtageren kan forstå, trods handicap. Det betyder for eksempel, at indhold der ikke er tekstbaseret, bør have en

alternativ tekstvisning. Den behøver ikke at være visuel, så længe den er tilgængelig for skærmlæsere. Har man f.eks. et billede, bør dette have en såkaldt alt-attribut med en beskrivelse af billedet. Ydermere kan man gøre brug af *aria-label*, til at beskrive elementer eller sektioner på siden, så modtageren nemmere kan navigere på siden.

Generelt set skal der være mere end én måde at forstå indhold på. Eksempelvis, kan man ikke nøjes med at adskille et element med en speciel farve. Det skal understøttes med en sekundær adskillelse. Det vil sige, at det ikke er nok at links har en anden tekstfarve end den øvrige brødtekst, da det vil blive overset af en farveblind modtager. Her kan man eksempelvis sætte en understregning under linkteksten.

Derudover bør dokumentet struktureres på en logisk måde, således at det stadig er forståeligt i en forsimplet visning. Hvis informationen står i den rigtige rækkefølge, vil en skærmlæser have en langt nemmere opgave i at præsentere indholdet korrekt til modtager.

Anvendelig

Det næste princip handler om, at indholdet og brugergrænsefladen skal kunne anvendes uden besvær. Det betyder blandt andet, at modtageren skal kunne navigere i dokumentet ved hjælp af tastaturet alene. Ergo er det vigtigt, som før nævnt, at dokumentet har en logisk struktur og rækkefølge. Har man elementer som skal kunne fokuseres, bør disse have en tabulator index-attribut (med mindre det drejer som et element, som eksplicit er fokusérbart) Elementer, som er i fokus, skal ligeledes markeres som sådan.

Forståelig

Indholdet skal forfattes på en letforståelig og letlæselig måde. Dette gælder sprogligt, såvel som teknisk. De steder, hvor der benyttes forkortelser eller fremmedord, bør der være en hjælpefunktion til at få de rigtige definitioner. Det kan eksempelvis gøres med en title-attribut, således at den rigtige definition vises, når man fører musen over. Derudover skal brugergrænsefladen generelt være nem, overskuelig og forudsigelig at bruge.

Robust

Det sidste princip i WCAG handler om, at brugergrænsefladen skal være robust, og tilgængelig i en bred vifte af brugeragenter – både de mest gængse, men også hjælpemidler, såsom skærmlæsere. Det opnås lettest ved, at koden er valid, velstruktureret og ikke afhænger af teknologier, som ikke understøttes af det brede flertal.

8. Appendiks 3 – WAI-ARIA

Her følger en række eksempler på opmærkning i overensstemmelse med WAI-ARIA.

```
<header role="banner">
```

En ARIA-rolle "banner" vil signalere, at indholdet inde i elementet er generelt for meddelelsen, og ikke hovedindholdet i beskeden i øvrigt. Det kan eksempelvis dreje sig om et logo og et billede.

```

```

En ARIA-rolle "presentation" vil signalere, at elementet kun er der for præsentationens skyld. Det kan være et grafisk element, som ellers ikke er relevant for selve meddelelsen i øvrigt.

<div role="complementary">

En ARIA-rolle "complementary" vil signalere, at indholdet er relevant for det omkringliggende indhold, men kun agerer som supplerende information. Det kan være en infoboks inde i en artikel.

Her kunne man med fordel have valgt at bruge et <aside>-element i stedet for ARIA, da dette element implicit har role="complementary".

9. Appendiks 4 – HTML Whitelist

HTML Whitelisten indeholder de tags, attributter og værdier der er godkendte brug i HTML-meddelelser.

Før en meddelelse leveres i Digital Post, både når de sendes fra et afsendersystem eller fra en visningsklient, bliver HTML-indholdet i meddelelsen valideret. Dette gøres for at sikre at dokumentet ikke indeholder skadeligt indhold, samt at det kun anvender HTML-elementer, som forventeligt kan renderes i langt de fleste systemer.

For at sikre dette anvender Digital Post en HTML whitelist, som er inddelt i to policies: *Strict*, som gælder for slutbrugere, der danner eller vedhæfter HTML via en visningsklient og *Lenient*, som anvendes for meddelelser der sendes via et afsendersystem.

Strict

- Ingen kommentarer tilladt i HTML dokumentet.
- Kun inline-styling af elementerne er tilladt.

Elements	Attributes
	Inline styling is allowed on all elements that support styling
Global	
All elements	
<i>Elements where attribute is relevant</i>	• role • title • aria-hidden • aria-label • aria-level • aria-orientation • aria-placeholder • aria-sort • aria-relevant • aria-activedescendant • aria-colcount • aria-colindex • aria-colspan • aria-describedby • aria-details • aria-labelledby • aria-posinset • aria-rowcount • aria-rowindex • aria-rowspan

Main Main blocks such as <html>, <body>, etc.	
html	• xmlns • lang
head	
meta	• charset • content • name • http-equiv ○ content-security-policy ○ content-type
title	
body	• lang
Semantic Html 5 semantic elements	
address	
article	
aside	
details	
figcaption	
figure	
footer	
header	
main	
mark	
nav	
section	
summary	
time	
Blocks Allow common block elements including <p>, <h1>, etc.	
p	
div	
h1	
h2	
h3	
h4	
h5	

h6	
hr	
ul	
ol	
li	
blockquote	
Formatting Allows common formatting elements including , <i>, etc.	
b	
i	
font	color, face, size
s	
u	
o	
sup	
sub	
ins	
del	
strong	
strike	
tt	
code	
big	
small	
br	
span	
em	
Tables Allow common table elements	
table	• summary • align • valign
tr	• align • valign
td	• align • valign
th	• align

	• valign
colgroup	• align • valign
caption	
col	• align • valign
thead	• align • valign
tbody	• align • valign
tfoot	• align • valign
Links	
a	• href ◦ https, mailto • target ◦ _blank
Images	
Allow elements from embedded sources only	
img	• alt • src ◦ data:image • border • height • width
Styles	Properties
Allow certain safe CSS properties in style="..." attributes. <style> element is not allowed	
• -moz-border-radius • -moz-border-radius-bottomleft • -moz-border-radius-bottomright • -moz-border-radius-topleft • -moz-border-radius-topright • -moz-box-shadow • -moz-outline • -moz-outline-color • -moz-outline-style • -moz-outline-width • -o-text-overflow • -webkit-border-bottom-left-radius • -webkit-border-bottom-right-radius	• url ◦ data uri only • -moz-inline-box, -moz-inline-stack, -moz-pre-wrap, -o-pre-wrap, -pre-wrap, 100, 200, 300, 400, 500, 600, 700, 800, 900, above, absolute, aliceblue, all-scroll, always, antiquewhite, aqua, aquamarine, armenian, at, auto, avoid, azure, baseline, behind, beige, below, bidi-override, bisque, black, blanchedalmond, blink, block, blue, blueviolet, bold, bolder, border-box, both, bottom, break-word, brown, burlywood, cadetblue, capitalize, caption, center, center-left, center-right, chartreuse, child, chocolate, circle, cjk-decimal, clip, closest-corner, closest-side, code, col-resize, collapse, condensed, contain, content-box, continuous, coral, cornflowerblue, cornsilk, cover, crimson, crosshair, cursive, cyan, darkblue, darkcyan, darkgoldenrod, darkgray, darkgreen, darkkhaki, darkmagenta, darkolivegreen, darkorange, darkorchid, darkred, darksalmon, darkseagreen, darkslateblue, darkslategray, darkturquoise, darkviolet, dashed, decimal, decimal-leading-zero, deeppink, deepskyblue, default, digits, dimgray, disc, disclosure-

• -webkit-border-radius • -webkit-border-radius-bottom-left • -webkit-border-radius-bottom-right • -webkit-border-radius-top-left • -webkit-border-radius-top-right • -webkit-border-top-left-radius • -webkit-border-top-right-radius • -webkit-box-shadow • azimuth • background • background-attachment • background-color • background-image • background-position • background-repeat • border • border-bottom • border-bottom-color • border-bottom-left-radius • border-bottom-right-radius • border-bottom-style • border-bottom-width • border-collapse • border-color • border-left • border-left-color • border-left-style • border-left-width • border-radius • border-right • border-right-color • border-right-style • border-right-width • border-spacing • border-style • border-top • border-top-color • border-top-left-radius • border-top-right-radius • border-top-style • border-top-width • border-width	closed, disclosure-open, dodgerblue, dotted, double, e-resize, ellipse, ellipsis, embed, ethiopic-numeric, expanded, extra-condensed, extra-expanded, fantasy, far-left, far-right, farthest-corner, farthest-side, fast, faster, female, firebrick, fixed, floralwhite, forestgreen, fuchsia, gainsboro, georgian, ghostwhite, gold, goldenrod, gray, green, greenyellow, groove, hand, hebrew, help, hidden, hide, high, higher, hiragana, hiragana-iroha, honeydew, hotpink, icon, indianred, indigo, inherit, inline, inline-block, inline-table, inset, inside, invert, italic, ivory, japanese-formal, japanese-informal, justify, katakana, katakana-iroha, khaki, korean-hangul-formal, korean-hanja-formal, korean-hanja-informal, large, larger, lavender, lavenderblush, lawngreen, left, left-side, leftwards, lemonchiffon, level, lightblue, lightcoral, lightcyan, lighter, lightgoldenrodyellow, lightgreen, lightgrey, lightpink, lightsalmon, lightseagreen, lightskyblue, lightslategray, lightsteelblue, lightyellow, lime, limegreen, line-through, linen, list-item, local, loud, low, lower, lower-alpha, lower-greek, lower-latin, lower-roman, lowercase, ltr, magenta, male, maroon, medium, mediumaquamarine, mediumblue, mediumorchid, mediumpurple, mediumseagreen, mediumslateblue, mediumspringgreen, mediumturquoise, mediumvioletred, menu, message-box, middle, midnightblue, mintcream, mistyrose, mix, moccasin, monospace, move, n-resize, narrower, navajowhite, navy, ne-resize, no-content, no-display, no-drop, no-repeat, none, normal, not-allowed, nowrap, nw-resize, oblique, oldlace, olive, olivedrab, once, orange, orangered, orchid, outset, outside, overline, padding-box, palegoldenrod, palegreen, paleturquoise, palevioletred, papayawhip, peachpuff, peru, pink, plum, pointer, powderblue, pre, pre-line, pre-wrap, progress, purple, red, relative, repeat, repeat-x, repeat-y, ridge, right, right-side, rightwards, rosybrown, round, row-resize, royalblue, rtl, run-in, s-resize, saddlebrown, salmon, sandybrown, sans-serif, scroll, se-resize, seagreen, seashell, semi-condensed, semi-expanded, separate, serif, show, sienna, silent, silver, simp-chinese-formal, simp-chinese-informal, skyblue, slateblue, slategray, slow, slower, small, small-caps, small-caption, smaller, snow, soft, solid, space, spell-out, springgreen, square, static, status-bar, steelblue, sub, super, suppress, sw-resize, table, table-caption, table-cell, table-column, table-column-group, table-footer-group, table-header-group, table-row, table-row-group, tan, teal, text, text-bottom, text-top, thick, thin, thistle, to, tomato, top, trad-chinese-formal, trad-chinese-informal, transparent, turquoise, ultra-condensed, ultra-expanded, underline, unrestricted, upper-alpha, upper-latin, upper-roman, uppercase, vertical-text, violet, visible, w-resize, wait, wheat, white, whitesmoke, wider, x-fast, x-high, x-large, x-loud, x-low, x-slow, x-small, x-soft, xx-large, xx-small, yellow, yellowgreen
--	--

• box-shadow • caption-side • color • cue • cue-after • cue-before • direction • elevation • empty-cells • font • font-family • font-size • font-stretch • font-style • font-variant • font-weight • height • image() • letter-spacing • line-height • linear-gradient() • list-style • list-style-image • list-style-position • list-style-type • margin • margin-bottom • margin-left • margin-right • margin-top • max-height • max-width • min-height • min-width • outline • outline-color • outline-style • outline-width • padding • padding-bottom • padding-left • padding-right • padding-top • pause • pause-after • pause-before • pitch • pitch-range • quotes • radial-gradient() • rect() • repeating-linear-gradient()	
---	--

• repeating-radial-gradient() • rgb() • rgba() • richness • speak • speak-header • speak-numeral • speak-punctuation • speech-rate • stress • table-layout • text-align • text-decoration • text-indent • text-overflow • text-shadow • text-transform • text-wrap • unicode-bidi • vertical-align • voice-family • volume • white-space • width • word-spacing • word-wrap	
--	--

Lenient

- Kommentarer er tilladt i HTML-dokumentet.
- Global styling via `<style>`-elementet er tilladt med den ene begrænsning at eksterne henvisninger fortsat ikke tillades.
- Inline-styling kan på samme vis anvendes uden restriktioner, ud over blokeringen af eksterne henvisninger.
- Atributterne `id` og `class` tillades for alle elementer, som understøtter dem.

Alt, som tillades for strict, er også tilladt for lenient, derudover er det muligt at anvende følgende elementer og attributter:

Elements	Attributes
Global	
All elements	
<i>Elements where attribute is relevant</i>	• id • class • lang • aria-setsize • aria-busy • aria-atomic • aria-controls • aria-current • aria-description • aria-disabled

	- aria-errormessage - aria-flowto - aria-haspopup - aria-invalid - aria-keyshortcuts - aria-live - aria-owns - aria-roledescription
Main Further options on elements such as <html>, <body>, etc.	
html	- xmlns:v - xmlns:o - xmlns:w - xmlns:m
body	- link - vlink
Blocks Further options on block elements	
span	
o:p	
p	align
div	align
hr	- size - width - align
picture	
source	- srcset - data:image - src - data:image - media - type
pre	
cite	
ol	- type - start
ul	type
Links	
a	name
Tables	
table	- border - cellspacing - cellpadding - width
td	- scope - headers

	• colspan • width • rowspan • nowrap • height
th	• scope • headers • colspan • width • rowspan • nowrap • height
colgroup	• width
col	• width • height • span
Styles	
style	Global style tag allowed. Local style attribute unrestricted. Only URLs to external resources are blocked.

