

VERKKOPALVELUIDEN SAAVUTETTAVUUSVAATIMUKSET



Ammattikorkeakoulututkinnon opinnäytetyö

Tietotekniikan koulutusohjelma

Syysy 2021

Leo Hyytiä

Tieto- ja viestintäteknikka

Tekijä Leo Hyytiä

Työn nimi Verkkopalveluiden saavutettavuusvaatimukset

Ohjaaja Toni Laitinen

Tiivistelmä

Vuosi 2021

Tässä opinnäytetyössä selvitetään, mitä vaatimuksia EU:n saavutettavuusdirektiivi ja WCAG 2.1 -saavutettavuusohjeistus asettavat verkkopalveluille. Työn päätarkoituksena on, että lukijalle tulee sen verran hyvä ymmärrys saavutettavuudesta, että hän osaa työn luettuaan tehdä verkkosivuistaan saavutettavuusohjeiden mukaiset.

Työssä käydään aluksi läpi esteelliset käyttäjäryhmät ja mitä erityistarpeita heillä on verkkosivujen käytössä. Sen jälkeen sivutaan joitain yleisiä asioita, joita saavutettavuustöitä tehdessä on hyvä tietää, kuten testaustyökaluja, saavutettavuusarvioinnin tekemistä ja aiheeseen liittyvää lainsäädäntöä. Sen jälkeen syvennytään tarkemmin WCAG 2.1 -saavutettavuusohjeistuksen sisältöön ja avataan sen asettamia vaatimuksia verkkopalveluille käytännön näkökulmasta ja esimerkkejä käyttäen.

Johtopäätöksenä on, että WCAG 2.1 -ohjeistusta noudattamalla verkkopalvelut ovat useimpien käyttäjäryhmien saavutettavissa. Yhdenvertaisuus paranee, kun mahdollisimman monet käyttäjät kykenevät toimimaan verkossa itsenäisesti mahdollisista rajoitteistaan huolimatta. Saavutettavuuden huomiointi verkkopalveluissa parantaa usein myös käytettävyyttä, joten saavutettavista verkkopalveluista hyötyvät kaikki käyttäjät.

Avainsanat saavutettavuus, WCAG 2.1, avustavat teknologiat

Sivut 20 sivua

This thesis explains the requirements that the accessibility directive of the European Union and WCAG 2.1 accessibility guidelines impose on web services. The main goal of this thesis is that the reader will gain enough knowledge about accessibility that they can make their own web pages comply with the accessibility guidelines.

The thesis starts with an explanation of different disabled user groups and any special needs they may have when using web services. Then it briefly touches on some general things that you should know about when working on accessibility, such as testing tools, accessibility evaluation and related legislation. Then it takes a deep dive into the contents of WCAG 2.1 accessibility guidelines, explaining the requirements it imposes on web services in a practical way and introduces multiple examples.

The conclusion is that web services which are compliant with WCAG 2.1 guidelines are accessible to most user groups. Equality increases when as many people as possible are able to use web services independently, regardless of their possible disabilities. Making web pages more accessible also often improves usability, so all users benefit from accessible web services.

Keywords Accessibility, WCAG 2.1, assistive technologies

Pages 20 pages

Sisälllys

1	Johdanto	1
2	Yleistä	1
2.1	Verkkopalveluiden saavutettavuus.....	1
2.2	Käyttäjien erityisryhmät.....	2
2.3	Saavutettavuuteen liittyvä lainsäädäntö	3
3	Työkaluja ja tekniikoita.....	4
3.1	Ruudunlukijat	4
3.2	WAVE.....	6
3.3	WAI-ARIA.....	6
4	Saavutettavuusarviointi.....	7
5	WCAG 2.1 -saavutettavuusohje	8
5.1	Yleistä WCAG 2.1 -saavutettavuusohjeesta	8
5.2	WCAG 2.1:n rakenne	8
5.3	WCAG:n tulevaisuus.....	9
5.4	Tekninen toteutus	10
5.4.1	Näppäimistökäyttö	10
5.4.2	Skaalautuvuus	10
5.4.3	Virhetilanteet ja ilmoitukset	11
5.4.4	Aikarajat	12
5.4.5	Tunnistautuminen	13
5.5	Rakenne, ulkoasu ja sisältö	14
5.5.1	Värit ja kontrasti	14
5.5.2	Tekstivastineet	16
5.5.3	Otsikot ja otsikkotasot	18
5.5.4	Maamerkit ja hyppylinkit	19
6	Yhteenveto	20
	Lähteet.....	21

1 Johdanto

Olen itse tehnyt saavutettavuuskorjauksia töissä ja tämän työn päätarkoituksena on olla dokumentti, joka olisi hyödyllinen ohje vastaavassa työssä. Työ käsittelee pääasiassa WCAG 2.1 -saavutettavuusohjeistusta käytännönläheisesti esimerkkejä käyttäen. Lisäksi siinä esitellään saavutettavuusarvioinnin prosessi, hyödyllisiä työkaluja ja tekniikoita, joista saavutettavuustöitä tehdessä on apua, ja käydään läpi joitain aiheeseen liittyviä yleisiä asioita, jotka on hyvä tietää.

WCAG 2.1 -ohjeistus on W3C:n julkaisema lista suosituksista, joita noudattamalla verkkosisällön saavutettavuus paranee huomattavasti käyttäjien erityisryhmien, kuten näkö-, kuulo- tai liikuntarajoitteisten henkilöiden kohdalla. Ohjeistus on kattava, mutta pitkä, melko hankalasti lähestyttävä dokumentti, jossa oleellinen informaatio ja toisiinsa liittyvät kriteerit ovat usein hajallaan pitkin sivua. Tämän opinnäytetyön päätarkoitus onkin purkaa dokumentti helposti sisäistettävään muotoon pääasiassa järjestelemällä sen olennaiset onnistumiskriteerit aihepiireittäin järkevään muotoon, käydä ne lyhyesti selkein esimerkein läpi ja tutkia suositeltuja toteutustapoja sekä mahdollisia saavutettavuusvirheitä.

2 Yleistä

2.1 Verkkopalveluiden saavutettavuus

Saavutettavuudella tarkoitetaan sitä, että erilaisiin erityisryhmiin kuuluvat käyttäjät, kuten näkö-, kuulo- tai liikuntarajoitteisten henkilöt, pystyvät käyttämään verkkopalveluja helposti ja itsenäisesti. Suuren osan palveluista siirtyessä verkkoon on tärkeää, että niitä pystyy käyttämään mahdollisimman suuri osa väestöstä itsenäisesti riippumatta heidän mahdollisista rajoitteistaan. (Aluehallintovirasto, n.d.-a)

On kuitenkin hyvä huomioida, että saavutettavuuden huomiointi verkkosuunnittelussa hyödyttää kaikkia käyttäjiä. Myös käyttäjän, jolla ei ole erityistarpeita, on helpompaa käyttää verkkopalvelua, joka on toteutettu selkeästi ja riittävää kontrastia käyttäen ja tekstitykset videoissa auttavat käyttäjiä, jotka eivät esimerkiksi teknisten ongelmien takia voi kuunnella videon ääntä.

(Aluehallintovirasto, n.d.-b)

2.2 Käyttäjien erityisryhmät

Vaikka saavutettavista verkkopalveluista hyötyy lähtökohtaisesti kaikki käyttäjät, erityisen tärkeää saavutettavuuden huomiointi on käyttäjille, joille verkon ja laitteiden käyttö on hankalaa esimerkiksi vamman tai korkean iän takia (Kehitysvammaliitto, n.d.-a). Arvion mukaan verkkopalvelujen heikko saavutettavuus vaikuttaa yli miljoonaan suomalaiseen (Aluehallintovirasto, n.d.-b). Näkö-, kuulo- ja puhevammat, liikunnalliset sekä kognitiiviset rajoitteet ja neurologiset sairaudet ovat tyypillisimpiä verkkopalvelujen käyttöä vaikeuttavia tekijöitä. Lisäksi on hyvä huomioida, että myös käyttäjän tietoteknisten taitojen puute kuuluu rajoitteisiin, sillä monet ikääntyneet käyttävät tietokonetta harvoin. (Kehitysvammaliitto, n.d.-a)

Suomessa on arviolta 80 000 näkövammaista, joista sokeita on 10 000. Myös näkörajoitteiset kykenevät käyttämään verkkoa itsenäisesti, mikäli saavutettavuus on otettu huomioon. Sokeille tärkeä apuväline verkon selaamiseen on ruudunlukuohjelma, joka lukee sivun sisällön käyttäjälle ääneen. Lisää ruudunlukijoista kappaleessa 3.1. Näkövammaisille käyttäjille tulee tarjota visuaaliselle sisällölle, kuten kuville ja videoille, vaihtoehtoinen esitystapa, joka yleensä on sisältöä kuvaava tekstivastine, jonka ruudunlukija tai muu avustava teknologia lukee käyttäjälle. Lievemmin näkörajoittuneet huomioidaan parhaiten, kun sivut ovat skaalautuvia eli niitä voi suurentaa reilusti, jotta niistä tulisi havaittavampia, ja värien kontrasti sivuilla on riittävä. (Kehitysvammaliitto, n.d.-b)

On arvioitu, että Suomessa on 740 000 kuulovammaista, joista täysin kuuroja on 8 000. Heille suurimmat saavutettavuushaasteet liittyvät videoihin, äänitiedostoihin ja erilaisiin ääniefekteihin. Tekstitysten tarjoaminen on hyvä tapa parantaa mediasisällön saavutettavuutta kuulorajoitteisille, ja niistä on hyötyä muillekin käyttäjille, sillä esimerkiksi sosiaalisessa mediassa videoita katsotaan usein ilman ääntä. On myös hyvä huomioida, että suuri osa täysin kuuroista on äidinkieleltään viittomakielisiä. Heidän suomen kielen taitonsa voi olla muuta väestöä huomattavasti heikompa, joten on hyvä pitää huoli siitä, että sisältö on kirjoitettu selkeästi. Verkkosivujen sisällön voi tarjota heille myös viittomakielellä, mikäli mahdollista. (Kehitysvammaliitto, n.d.-c)

Verkon käyttöä vaikeuttavia fyysisiä ja motorisia rajoitteita ovat esimerkiksi käsien vapina tai halvaantuminen, raajojen menetykset, tuntoaistin rajoitteet sekä tietyt sairaudet, kuten nivelreuma. Mikäli rajoitteet ovat lieviä, voi käyttäjä käyttää sivuja normaalisti hiirellä tai kosketusnäytöllä, mutta haasteita ilmenee sorminäppäryyden kanssa. Jos rajoitteet ovat

vaikeampia, saattaa käyttäjä selata sivua pelkällä näppäimistöllä tai jollain apuvälineellä, joihin lukeutuvat esimerkiksi ohjaaminen silmänliikkeillä, puheella tai päähän asetettavalla osoittimella. Kosketusnäytön käyttöä varten on saatavilla suuhun pantavia tikkuja. (Kehitysvammaliitto, n.d.-d)

Verkkosivujen suunnittelussa on otettava vaihtoehtoiset käyttötavat huomioon ja on pidettävä huoli siitä, että sivut ovat käytettävissä pelkällä näppäimistöllä. Käsien vapinasta kärsivien käyttäjien verkkoselaamista helpottaa parhaiten se, että klikattavat elementit ovat tarpeeksi suurikokoisia eivätkä ne ole liian lähekkäin. Motoriset rajoitteet vaikuttavat usein myös reaktionopeuteen, joten käyttäjille on annettava riittävästi aikaa sivun selaamiseen, mikä on huomioitava esimerkiksi sivulla mahdollisesti olevien aikarajojen suunnittelussa. Näiden rajoitusten huomioinnista on hyötyä myös ikääntyneille käyttäjille. (Kehitysvammaliitto, n.d.-d)

Verkon käyttöä vaikeuttavat kognitiiviset ja kielelliset rajoitteet sisältävät esimerkiksi luetun ymmärtämiseen, puheeseen ja oppimiseen liittyviä vaikeuksia. Ongelmien kirjo on laaja, joten kaikkien tarpeiden huomioon ottaminen on hankalaa. Lähtökohtaisesti kognitiivisten vaikeuksien huomiointi kuitenkin alkaa siitä, että sivujen käytettävyys on hyvä. Tämä hyödyttää myös muita käyttäjäryhmiä. Käyttäjät, joilla on kielellisiä vaikeuksia, hyötyvät siitä, että sivut on kirjoitettu sekokielellä tai selkeällä yleiskielellä. Selkokielen käyttö hyödyttää myös käyttäjiä, joiden äidinkieli on jokin muu kuin suomi. (Kehitysvammaliitto, n.d.-e)

2.3 Saavutettavuuteen liittyvä lainsäädäntö

Laki digitaalisten palvelujen tarjoamisesta tuli voimaan 1. huhtikuuta 2019 ja sen yksi päätarkoituksista on parantaa vammaisten ihmisten mahdollisuuksia käyttää verkkopalveluita yhdenvertaisesti. Käytännössä se velvoittaa julkista sektoria ja tiettyjä yksityisiä toimijoita, kuten liikenne-, posti-, pankki- ja rahoituspalvelujen tarjoajia, varmistamaan sähköisten palvelujensa saavutettavuus. Lain avulla pantiin täytäntöön Euroopan parlamentin ja neuvoston direktiivi (EU) 2016/2102, jossa on linjattu julkisen sektorin vastuulla olevien verkkopalvelujen saavutettavuusvaatimukset. (Laki digitaalisten palvelujen tarjoamisesta 306/2019 § 1, § 3)

Laissa saavutettavuusvaatimukset on määritelty niin, että palveluntarjoajien, joita laki koskee, on varmistettava palvelujensa havaittavuus, ymmärrettävyys, hallittavuus ja toimintavarmuus (Laki digitaalisten palvelujen tarjoamisesta 306/2019 § 7). Käytännössä palveluilta vaaditaan WCAG 2.1

-ohjeistuksen A- ja AA-tason kriteerien noudattamista, pois lukien suorat verkkolähettykset. AAA-tason kriteerien toteuttamista ei laissa vaadita. (Aluehallintovirasto, n.d.-c) Mikäli vaatimusten toteuttaminen aiheuttaa saavutettavuusarvioinnin perusteella palveluntarjoajalle kohtuuttoman rasitteen, voidaan vaatimuksista tietyiltä osin poiketa. On erityisen tärkeää ottaa huomioon vammaisten käyttäjien tarpeet palvelun suhteen kohtuutonta rasitetta arvioidessa, mutta myös palveluntarjoajan koko, taloudellinen asema ja toiminnan luonne sekä laajuus ovat vaikuttavia tekijöitä. Käytännössä siis isommilla toimijoilla ja keskeisten palveluiden tarjoajilla on vaikeampaa vedota kohtuuttomaan rasitteeseen. (Laki digitaalisten palvelujen tarjoamisesta 306/2019 § 8)

Laki velvoittaa palveluntarjoajaa tekemään saavutettavuusarvion ja kirjaamaan ylös, miltä osin palvelu ei täytä saavutettavuusvaatimuksia ja perustelut sille, miksi vaatimuksista on poikettu. Palveluntarjoajan tulee julkaista tiedot verkkosivuillaan saavutettavuusselosteessa, johon tulee lisäksi kirjata ohjeet vaihtoehtoiseen tapaan palvelun käyttämiseen, mikäli se ei ole käyttäjälle saavutettava, ja palveluntarjoajan yhteystiedot saavutettavuuspalautteen lähettämistä varten. Käyttäjää tulee myös ohjeistaa siitä, missä hän voi tehdä saavutuskantelun tai -selvityspyynnön valvontaviranomaiselle. Lain valvontaviranomainen on Etelä-Suomen aluehallintovirasto. (Laki digitaalisten palvelujen tarjoamisesta 306/2019 § 9, § 12)

3 Työkaluja ja tekniikoita

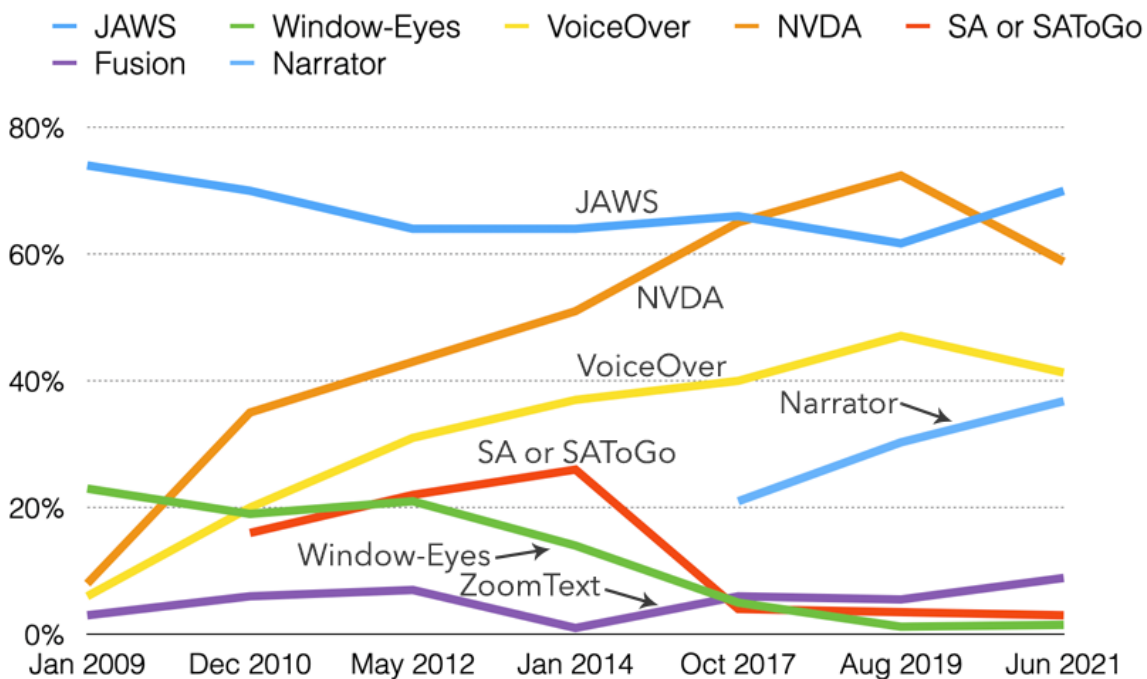
3.1 Ruudunlukijat

Ruudunlukija tai ruudunlukuohjelma on näkövammaisten henkilöiden avuksi suunniteltu apuohjelma, joka lukee näytöllä näkyvän sisällön käyttäjälle ääneen. Ruudunlukuohjelmia on saatavilla sekä tietokoneille että mobiililaitteille. Mobiililaitteilla ruudunlukijaa ohjataan sormieleillä, kuten napsautuksilla ja pyyhkäisyillä, ja tietokoneella ohjainlaitteena toimii näppäimistö. (Kehitysvammaliitto, n.d.-f)

Ruudunlukijan käyttö verkkosivuilla edellyttää, että verkkosivut on suunniteltu havaittaviksi sekä käytettäviksi pelkällä näppäimistöllä. Havaittavuus tarkoittaa sitä, että esimerkiksi kuvilla on tekstivastineet, jotka ruudunlukija voi lukea käyttäjälle ja näppäimistöikäytöllä tarkoitetaan sitä, että kaikki sivun toiminnallisuus, kuten verkkolomakkeet tai valikot, tulee olla käytettävissä pelkällä näppäimistöllä. WCAG-ohjeistus kattaa nämä alueet laajasti. (Kehitysvammaliitto, n.d.-f)

Ruudunlukuohjelmia on tarjolla monia. Jokaisen suuren käyttöjärjestelmän mukana tulee oma ruudunlukijansa: Applen laitteissa on mukana VoiceOver, Android-laitteissa TalkBack ja Windows-laitteissa Narrator. Lisäksi markkinoilla on useita itsenäisiä ruudunlukuohjelmia, joista suosituimpia ovat ilmainen NVDA ja kaupallinen JAWS. WebAIM:in tekemän kyselytutkimuksen (2021) mukaan käytetyin ruudunlukija on JAWS ja toiseksi suosituin on NVDA. Suosituimpien ruudunlukijoiden käyttöastetta on kuvattu kuvassa 1. Useimmat ruudunlukijan käyttäjät käyttävät kyselyn mukaan useampaa kuin yhtä ruudunlukuohjelmaa. Selainten osalta käytetyimmäksi selaimeksi on noussut Google Chrome, kun taas Mozilla Firefoxin ja Internet Explorerin osuus on huomattavasti laskenut. Suosituin ruudunlukija-selainyhdistelmä on Google Chrome ja JAWS. Mobiililaitteista ruudunlukijoiden käyttäjien keskuudessa suosituimpia ovat Applen laitteet lähes kolmen neljäsosan osuudellaan, mikä näkyy myös siinä, että Applen VoiceOver on ylivoimaisesti suosituin mobiiliruudunlukija, Androidin TalkBackin ollessa toisena.

Kuva 1: Suosituimmat ruudunlukijat (WebAim, 2021)



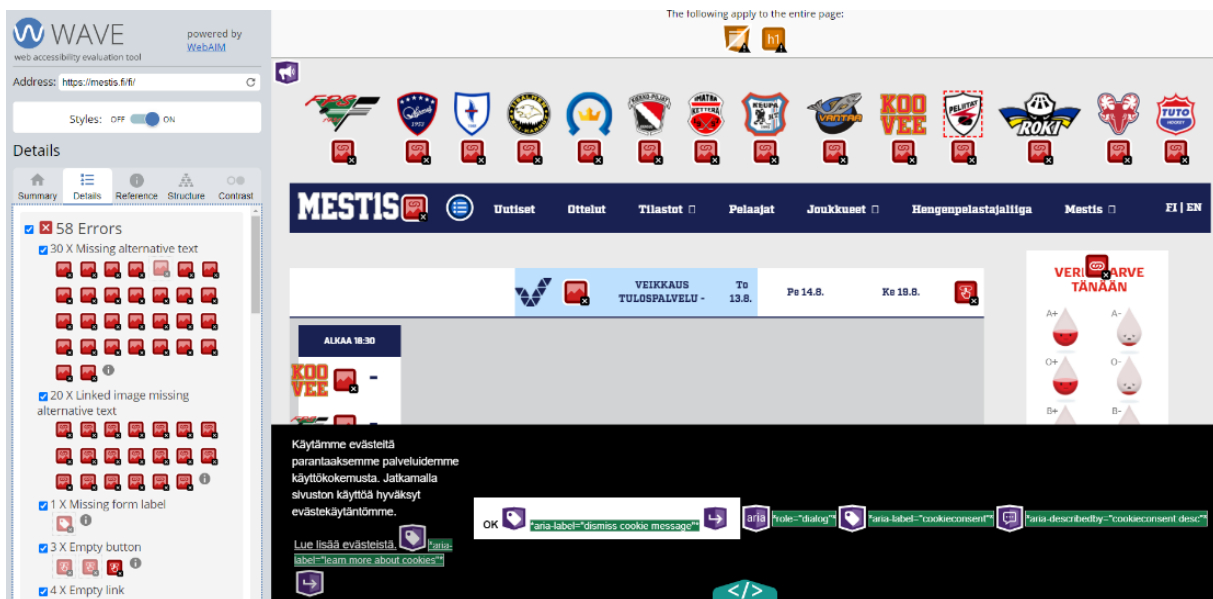
Ruudunlukijoiden laaja kirjo tekee verkkopalvelujen saavutettaviksi tekemisestä hankalaa, sillä kaikki ruudunlukija-selainyhdistelmät eivät tue kaikkia web-saavutettavuustekniikoita ja toiminnallisuudessa on usein merkittäviä eroja. Erityisesti Internet Explorerissa on useiden saavutettavuustekniikoiden tuen osalta huomattavia puutteita, mutta kyselytutkimus näyttää, että

sillä on edelleen kohtuullisesti käyttäjiä, Edgen tosin vietyä jonkin verran sen käyttäjistä. (WebAIM, 2021)

3.2 WAVE

Kuvassa 2 esitelty WAVE on web-työkalu, joka etsii käyttäjän antamalta verkkosivulta saavutettavuusongelmia. Se etsii esimerkiksi kontrastivirheitä, taulukko-ongelmia ja puuttuvia kuvatekstejä. Lisäksi se näyttää otsikkorakenteen arviointia varten ja sisältää kontrastilaskurin, jolla voi testata vaihtoehtoisia värejä suoraan verkkosivulla. Työkalu on hyödyllinen apuväline niin saavutettavuusarviointia kuin kehitystyötä tehdessä, mutta yksinään se ei osaa tunnistaa kaikkia saavutettavuusongelmia. WAVE löytyy osoitteesta <https://wave.webaim.org/>.

Kuva 2: Web-sivun saavutettavuusongelmat WAVE-työkalun listaamina



3.3 WAI-ARIA

WAI-ARIA, tai Accessible Rich Internet Applications, on tapa saada saavutettavuusvaatimusten vaatimaa semantiikkaa html-sivuille, joilla se muuten olisi mahdotonta. Tavallisella html:llä kirjoitetuille verkkosivuille tulee saavutettavuushaasteita, kun siihen lisää esimerkiksi muuttuvaa sisältöä tai visuaalista informaatiota, kuten kuvia. WAI-ARIA:n avulla html laajentuu niin, että saavutettavuuden näkökulmasta haasteellisten käyttöliittymäelementtien toteuttaminen on mahdollista saavutettavalla tavalla. (W3C, 2017)

WAI-ARIA:an kuuluu esimerkiksi navigointimerkkejä alueille, joihin käyttäjä haluaa navigoida nopeasti, kuten esimerkiksi navigointipalkkiin, pääsisältöön, lomakkeisiin tai hakupalkkiin. WAI-ARIA:an kuuluu myös rooleja, joilla voi muuttaa html-elementin semantiikkaa niin, että se vastaa käyttäjän odotuksia myös ruudunlukijan kanssa selatessa, kuten div-elementein toteutetun taulukon tai valikon tulkitseminen asianmukaisella tavalla, sekä mahdollisuus tarvittaessa piilottaa tiettyjä käyttöliittymäelementtejä ruudunlukijoilta. Myös päivittyvien alueiden, kuten chatin, tai uuden, näkyviin tulevan elementin, kuten virheilmoituksen, käsittely saavutettavasti onnistuu WAI-ARIA:n avulla. (W3C, 2017)

4 Saavutettavuusarviointi

Saavutettavuusarviointi alkaa suunnittelulla, jossa päätetään arvionnin tavoitteet. Olennaisia kysymyksiä ovat, mille tasolle saavutettavuus kohdesivuilla halutaan eli käytetäänkö esimerkiksi jotain WCAG 2.1 -tasoa, ja osallistuuko arviointiin sivuston oikeat käyttäjät vai tekeekö sen asiantuntija. Tärkeää on myös rajata arvionnissa käytettävien verkkoselainten, päätelaitteiden ja avustavien teknologioiden määrää mielekkääksi, sillä niiden mahdollisia yhdistelmiä on paljon. Läpi kannattaa käydä myös se, miten tarkasti arviointi halutaan raportoida, arvioidaanko saavutettavuusongelmien vakavuutta ja tarjotaanko myös korjausehdotukset. Suunnittelun yhteydessä arvioidaan myös työmäärä saavutettavuusarviointille ja korjauksille. Aikataulu riippuu siitä, kuinka paljon saavutettavuusongelmia sivulla on. (Kehitysvammaliitto, n.d.-g)

Itse saavutettavuustestaamisessa ja korjausehdotuksia kannattaa sivustoa käyttää avustavien teknologioiden avulla. Kun sivustoa kuuntelee ruudunlukijalla, selviää, millaisia haasteita sokeilla tai heikkonäköisillä sivustoa käytettäessä on. Ruudunlukijakäyttäjän näkökulmasta on tärkeää, että sivusto on käytettävissä ainoastaan näppäimistöllä, ja siksi näppäimistökäyttökin kannattaa testata huolellisesti. (Kehitysvammaliitto, n.d.-h)

Saavutettavuus kannattaa testata yleisimmillä käyttöjärjestelmillä ja selaimilla erikseen, sillä näin voi löytyä päätelaite- ja versiokohtaisia saavutettavuusongelmia. Erityisesti vanhemmilla selaimilla eri saavutettavuusratkaisujen toiminta on paikoin puutteellista. Päätelaitetestausta suositellaan tehtäväksi ainakin Windows 7:llä ja uudemmilla Windows-käyttöjärjestelmillä, MacOS:llä sekä Android- ja iPhone-älypuhelimilla ja selaimista testaukseen kannattaa valita suosituimmat, eli Chrome, Safari, Internet Explorer, Firefox ja Edge. (Kehitysvammaliitto, n.d.-h)

Mikäli saavutettavuustestaus on tehty käyttäen jotain WCAG-tasoa, on saavutettavuuskorjauksia tehdessä havaintoon liittyvä WCAG-kriteeri tärkein työkalu. Havainnon osoittama kohta korjataan niin, että se täyttää kriteerin. Jos WCAG:ta ei ole käytetty, on korjausten tekemisessä eteneminen tapauskohtaista ja riippuu siitä, kuinka tarkasti saavutettavuushavainnot on dokumentoitu ja onko korjausehdotuksia tehty. WCAG:ta voi kuitenkin käyttää apuna, sillä se dokumentoi laajasti erilaisia saavutettavuusongelmia ja sen oheisdokumenteissa on esitelty useita korjaustapoja.

5 WCAG 2.1 -saavutettavuusohje

5.1 Yleistä WCAG 2.1 -saavutettavuusohjeesta

WCAG eli Web Content Accessibility Guidelines (suom. Verkkosisällön saavutettavuusohjeet) on erityisesti web-suunnittelijoille, kehittäjille ja testaajille suunnattu dokumentti, jonka tarkoituksena on auttaa verkkosisällön saavutettavuuden parantamisessa. Siinä yritetään huomioida mahdollisimman suuri joukko erilaisia verkon selaamiseen vaikuttavia rajoitteita, kuten näkö- ja kuulohäiriöt sekä kognitiiviset rajoitteet, tosin viimeisenä mainittujen osalta se on puutteellinen. Ohjeistus on varsin yleistasoinen eikä näin ollen ole riippuvainen tietystä teknologiasta tai käytetystä laitteesta. Teknologiaspesifiset ohjeet WCAG:n onnistumiskriteerien täyttämiseen, kuten myös niiden tarkempaan tulkitsemiseen, on tarjottu erillisissä dokumenteissa, joihin WCAG-dokumentti viittaa. (W3C, 2018)

WCAG 2:n ensimmäinen versio 2.0 julkaistiin joulukuussa 2008 ja se päivitettiin versioon 2.1 kesäkuussa 2018 (W3C, 2018). WCAG 2.1 -dokumentista on olemassa virallinen, marraskuussa 2019 julkaistu suomenkielinen käännös, joka löytyy osoitteesta <https://www.w3.org/Translations/WCAG21-fi/>. Mainittuja tarkempia ohjedokumenteja ei ole suomennettu.

5.2 WCAG 2.1:n rakenne

WCAG 2.1 -ohje on jaettu tasoihin periaatteet, ohjeet, onnistumiskriteerit sekä riittävät ja neuvoa-antavat tekniikat. Periaatteet ovat ylätasoinen ohjeistusta ja määrittelevät, millaisia saavutettavien verkkosivujen tulee olla: havaittavia, hallittavia, ymmärrettäviä ja lujatekoisia. Ohjeet ovat hieman spesifimpiä sääntöjä, kuten ”Tarjoa käyttäjille tapoja navigoida, etsiä sisältöä ja määrittää

sijaintinsa”. Niitä on yhteensä 13. Ohjeet on edelleen jaettu onnistumiskriteereihin, jotka tarkemmin määrittelevät, millaisia verkkosivuista on tehtävä. Ne on luokiteltu ohjeidenmukaisuuden tasoihin A, AA ja AAA ja ne määräytyvät seuraavalla tavalla: (W3C, 2018)

- A-tason mukainen verkkosivu toteuttaa kaikki A-tason onnistumiskriteerit (30 kriteeriä)
- AA-tason mukainen verkkosivu toteuttaa kaikki AA-tason (20 kriteeriä) ja A-tason onnistumiskriteerit
- AAA-tason mukainen verkkosivu toteuttaa kaikki AAA-tason onnistumiskriteerit (28 kriteeriä) ja lisäksi sekä A- että AA-tasojen kaikki kriteerit.

Korkein saavutettavuuden taso WCAG-standardissa on siis AAA, johon on sisällytetty eniten kriteerejä. Sen toteuttavat verkkopalvelut ovat useimpien käyttäjäryhmien saavutettavissa. On kuitenkin huomionarvoista, että edes korkeimmalla tasolla ei kyetä takaamaan saavutettavuutta tietyille erityisryhmille, joilla on esimerkiksi kognitiivisia rajoitteita. (W3C, 2018; W3C, 2016)

5.3 WCAG:n tulevaisuus

WCAG:n versio 2.2 on tarkoitus julkaista vuonna 2021. Se tulee sisältämään aiempien versioiden ohjeiden lisäksi joitain uusia ohjeita sekä olennaisena lisänä sitaatteja esimerkkihenkilöiltä, joiden tarkoitus on auttaa ohjeiden ymmärtämisessä. Merkittävimmät uudet kriteerit koskevat saavutettavaa tunnistautumista, selaimen fokuksen näkyvyyttä, aputoimintojen saavutettavuutta sekä raahattavaa sisältöä. (W3C, 2021-a)

Myös WCAG 3.0 on työn alla. Sen tarkoituksena on tehdä ohjeesta ymmärrettävämpi, yleispätevämpi ja käyttäjäkeskeisempi kuin nykyisestä versiosta, joka on varsin teknologiakeskeinen ja asiantuntijoille suunnattu. Rakenteeseen ja onnistumiskriteereihin on tulossa merkittäviä muutoksia. WCAG 3.0 on toistaiseksi varsin keskeneräinen. (W3C, 2021-b)

Seuraavissa kappaleissa käydään läpi olennaiset WCAG 2.1 -ohjeistuksen onnistumiskriteerit aihepiireittäin järjesteltynä.

5.4 Tekninen toteutus

5.4.1 Näppäimistökäyttö

On tärkeää, että kaikki verkkosivuilla oleva toiminnallisuus on käytettävissä pelkällä näppäimistöllä. Tästä hyötyvät erityisesti sokeat, heikkonäköiset, joilla on vaikeuksia erottaa kursoria näytöltä, esimerkiksi vapinan vuoksi yksinomaan näppäimistöllä verkkoa käyttävät, sekä vaihtoehtoisia, näppäimistöemulaattoreina toimivia syöttölaitteita käyttävät henkilöt. Ainoana poikkeuksena näppäimistökäytön vaatimukseen on tapaukset, joissa toiminnallisuus riippuu käyttäjän käyttämästä polusta, kuten piirto-ohjelmalla piirtäminen tai useiden pelien toiminta. (W3C, 2021-c)

Näppäimistökäytön tulee olla keskeytyksetöntä ja tapahtua loogisessa järjestyksessä. Näppäimistökäyttäjää ei tulisi sekoittaa esimerkiksi navigointijärjestyksen epäloogisuudella tai kohdistuksen siirtelyllä javascriptillä. Jokaisesta elementistä, johon näppäimistöllä voidaan navigoida, tulee päästä myös pois yksinomaan näppäimistöä käyttäen. Jos käyttäjän täytyy tehdä jotain muuta navigoidakseen eteenpäin kuin käyttää normaaleja navigoinnin näppäinkomentoja, kuten nuoli- tai tab-näppäimiä, tulee häntä neuvoa eteenpäin. Sivulla voi jäädä näppäimistöllä navigoidessa jumiin, mikäli kohdistus siirtyy esimerkiksi widgetiin tai dialogi-ikkunaan, joka renderöityy erikseen. Tällaisten elementtien käyttö sivuilla on sallittua, kunhan kohdistus siirtyy avautuneeseen dialogiin automaattisesti ja käyttäjä tietää, miten siitä pääsee pois ja kohdistuksen saa vapautettua. (W3C, 2021-d)

Näppäimistökäyttöön liittyy oleellisesti myös näppäimistökohdistuksen näkyvyys. Käytännössä tällä tarkoitetaan sitä, että käyttäjän on mahdollista nähdä näytöltä, missä näppäimistökohdistus sillä hetkellä sivulla on. Tyypillinen tapa tämän toteuttamiseksi on erillinen css-tyyli esimerkiksi tekstikenttien reunoissa tai nappuloissa, joka tulee käyttöön silloin, kun kohdistus siirtyy kyseiseen elementtiin. (W3C, 2021-e)

5.4.2 Skaalautuvuus

Heikkonäköisille on tärkeää, että sivun sisältöä pystyy suurentamaan tarpeeksi, jolloin tekstiä voi lukea ilman avustavia teknologioita. WCAG-ohjeidenmukaisuudella pyritään varmistamaan, että

verkkosivua pystyy suurentamaan 200-prosenttisesti niin, että toiminnallisuus tai luettavuus ei kärsi. Suurenus on usein selaimen ominaisuus joten kehittäjän vastuulle jää varmistaa, että suurennettaessa sisältö pysyy luettavana eli tekstin katkeamista tai häviämistä ei tapahdu. (W3C, 2021-f)

Tekstiä suurennettaessa selaimen suurennuksella myös muu sisältö suurenee samassa suhteessa. Näin ollen on tärkeää varmistaa, että sisältö pysyy käytettävänä isoillakin suurennuksilla. WCAG-ohjeidenmukainen responsiivinen sisältö siirtyy suurennettaessa yhteen sarakkeeseen, jolloin käyttäjät, joilla on heikko näkö tai vaikeuksia seurata tekstiä, voivat lukea helpommin ilman, että sivua tarvitsee vierittää useammassa suunnassa. On pidettävä huoli siitä, että kaikki sisältö pysyy edelleen käytettävissä myös skaalautuessa. (W3C, 2021-g)

5.4.3 Virhetilanteet ja ilmoitukset

Virheen tapahtuessa käyttäjälle on tiedotettava siitä mahdollisimman selkeästi ja tarkasti, jotta käyttäjä tietää, mitä on tapahtunut. On tärkeää, että virhe käy ilmi kaikille käyttäjille, myös avustaville teknologioille. Näin ollen esimerkiksi syöttövirheen kohdalla pelkkä virheellisen kentän osoittaminen ei riitä, sillä ruudunlukija tunnistaisi virheen vasta silloin, kun käyttäjä navigoi virheellisen kentän kohdalle. Käyttäjää auttaa myös se, jos palautetta annetaan myös siitä, kun toiminto on onnistunut. (W3C, 2021-h)

Käyttäjällä, jolla on kognitiivisia rajoitteita, voi olla vaikeuksia virheiden ymmärtämisessä. Siksi virheviestin tulisi pyrkiä mahdollisuuksien mukaan kertomaan virheen lisäksi, miten käyttäjä voi korjata virheen. Hyviä virheviestejä ovat esimerkiksi ”Kirjautuminen epäonnistui. Salasana puuttuu. Syötä salasana ja yritä uudelleen” tai ”Päivämäärä virheellinen. Syötä päivämäärä muodossa PP.KK.VVVV, esimerkiksi 08.11.2021”. Virheviesti ei kuitenkaan saa vaarantaa käyttäjän tietoturva. (W3C, 2021-i)

Virheviesti on muuttuvaa sisältöä, joten ARIA-rooli alert tai ARIA live-alueet ovat sopivia tekniikoita avustavien teknologioiden, kuten ruudunlukijan, ilmoittamiseksi virheestä. Toiminnaltaan role=”alert” ja aria-live=”assertive” ovat toiminnaltaan samanlaisia: kun attribuutilla merkätyn elementin sisältö muuttuu virheviestin lisäämisen yhteydessä, ruudunlukija lukee sen ääneen automaattisesti ilman, että kohdistusta tarvitsee siirtää. Virheellisen tai

puuttuvan kentän voi merkata avustaville teknologioille havaittavaksi käyttämällä aria-invalid="true" -attribuuttia. (W3C, 2021-h)

Virheviestien lisäksi verkkopalvelu voi näyttää erilaisia statusviestejä, kuten tietoja latauksen edistymisestä tai hakutulosten määrästä. Nämä eivät ole virheviestien tavoin kriittisiä, joten niiden yhteydessä ei tulisi turhaan häiritä käyttäjää alert-roolin välittömällä lukemisella ja muun toiminnan keskeyttämisellä, mutta niiden tulisi kuitenkin olla havaittavia. Tällaisia viestejä varten on omat ARIA-roolit status ja log ja live-alue aria-live="polite", joilla merkityt viestit luetaan seuraavassa sopivassa välissä, esimerkiksi kun ruudunlukija on puhunut nykyisen lauseensa loppuun. (W3C, 2021-j)

Ohessa on eräs mahdollinen tapa toteuttaa yksinkertainen virheviesti alert-roolin avulla. Kun error-tekstiä muutetaan esimerkiksi JavaScriptillä, pitäisi ruudunlukijan lukea errorRegionin sisältö välittömästi. Attribuutti aria-atomic määrittelee, luetaanko muuttunut alue kokonaan.

```
<div id="errorRegion" aria-live="assertive" aria-atomic="true">  
  <p id="error" style="color:red"></p>  
</div>
```

5.4.4 Aikarajat

Useissa verkkosovelluksissa on erilaisia aikarajoja, joista tyypillisiä esimerkkejä ovat istunnon ajoitettu vanheneminen sekä automaattisesti tietyin väliajoin päivittyvä sisältö. Aikarajoista kuitenkin seuraa saavutettavuushaasteita esimerkiksi ruudunlukijan käyttäjälle, joka saattaa tarvita reilusti aikaa sivun läpikäymiseen ja fyysisistä rajoituksista kärsiville, joilla reagointi ja kirjoittaminen voi olla hidasta. Aikarajoihin liittyvien onnistumiskriteerien tavoitteena on minimoida aikarajoista käyttäjille aiheutuva haitta antamalla riittävästi aikaa verkkosivun käyttöön, mahdollisuuksien rajoissa. (W3C, 2021-k)

Aikarajavaatimus on helpointa täyttää, kun verkkosivun käytölle ei aseteta mitään aikarajaa. Mikäli aikaraja kuitenkin halutaan, vaihtoehtoisia tapoja on esimerkiksi antaa käyttäjälle mahdollisuus säätää aikarajaa pidemmäksi tai poistaa se kokonaan käytöstä. Ohjeiden mukainen vaihtoehto on myös skripti, joka varoittaa käyttäjää aikarajan lähestymisestä ja antaa mahdollisuuden pyytää lisää aikaa, kunhan aikarajan lykkäämiseen annetaan vähintään 20 sekuntia aikaa ja käyttäjältä vaadittava toiminta on tarpeeksi yksinkertaista, kuten napin klikkaaminen. (W3C, 2021-k)

Joissain tapauksissa aikaraja on välttämätön ja onnistumiskriteerissä otetaan huomioon tiettyjä poikkeuksia. Mikäli toiminnallisuus menee rikki ilman aikarajaa, eikä sitä voi toteuttaa millään ohjeidenmukaisella tavalla, voidaan tässä kriteerissä joustaa. Aikaraja on sallittu myös reaaliaikaisissa tapahtumissa, joissa aikaraja on olennainen osa toiminnallisuutta, kuten verkkohuutokaupoissa. Lisäksi yli 20 tunnin aikarajat ovat sallittuja. (W3C, 2021-k)

Istunnon vanhenemisiin ja aikakatkaisuihin liittyen on myös tärkeää tiedottaa käyttäjää siitä, kuinka pitkä toimetttömyys aiheuttaa aikakatkaisun ja mahdollisen tietojen menetyksen. Paras tapa toteuttaa saavutettava aikakatkaisu on tallentaa käyttäjän sivulla mahdollisesti antamat tiedot määräajaksi, ohjeessa 20 tuntia, jotta uudelleentunnistautumisen jälkeen on mahdollista jatkaa samasta paikasta. Aina tämä ei kuitenkaan ole käytännöllistä tai edes mahdollista, jolloin ilmoitus aikakatkaisusta riittää. (W3C, 2021-l)

5.4.5 Tunnistautuminen

Saavutettava tunnistautuminen on WCAG 2.2 -versiossa uutena tuleva korkeimman A-tason onnistumiskriteeri. Useimmat verkkopalvelut vaativat edelleen käyttäjätunnuksen ja salasanan muistamista, mikä on tietyille kognitiivisista vaikeuksista kärsiville käyttäjäryhmille vaikeaa tai mahdotonta. Tämän onnistumiskriteerin tarkoituksena on tarjota vaihtoehtoisia tunnistautumistapoja, jotta palvelut olisivat saavutettavia myös näille käyttäjäryhmille. (W3C, 2021-m)

Tarjolla on siis oltava ainakin yksi tunnistautumistapa, joka ei testaa kognitiivisia kykyjä. Sellaisiksi luokitellaan kuuluvan esimerkiksi sivukohtaisen salasanan muistaminen, kuvion pyyhkäiseminen kosketusnäytöllä sekä CAPTCHA:n eli tiettyjen elementtien kuvasta etsimiseen perustuvien kuva-arvoitusten ratkaiseminen. Myös tunnuksen palauttamisen ja monivaiheisen tunnistautumisen on onnistuttava ilman kognitiivisten kykyjen testausta. (W3C, 2021-m)

Käytännössä tämän onnistumiskriteerin voi täyttää sallimalla käyttäjätunnus- ja salasananakenttien automaattisen täytön ja tietojen kopioimisen ja liimaamisen. Näin käyttäjä voi käyttää salasananamanageria kirjautumistietojen automaattiseen täyttöön. Vaihtoehtoisesti palvelu voi antaa käyttäjän kirjautua jollain laitteensa tarjoamalla tunnistusmetodilla, kuten sormenjälki- tai kasvojentunnistuksella, tai kolmannen osapuolen tunnistautumispalvelua käyttämällä. Myös

käyttäjän sähköpostiin toimitettu kirjautumislinkki on riittävä tapa tämän onnistumiskriteerin täyttämiseen. (W3C, 2021-m)

5.5 Rakenne, ulkoasu ja sisältö

5.5.1 Värit ja kontrasti

Näkörajoitteisten, värisokeiden ja avustavia teknologioita käyttävien käyttäjien kannalta on tärkeää, että väri ei ole ainoa keino saada sivulta tietoa selville. Värit ovat tärkeä osa web-suunnittelua ja niitä on myös saavutettavuusnäkökulmasta suositeltavaa käyttää tuomaan ylimääräistä visuaalista informaatiota tai kiinnittämään huomiota, mutta sisältö tulisi olla käytettävissä, vaikka värejä ei näkisi. Tyypillisiä tähän liittyviä virheitä on lomakkeiden virheellisten kenttien korostaminen yksinomaan esimerkiksi punaisella värillä, taulukot, joissa väri on ainut tapa asioiden erottamiseksi toisistaan tai linkit, jotka eivät erotu tekstistä muuten kuin värin perusteella. (W3C, 2021-n)

WCAG:ssa määritellään myös, että tekstin ja taustavärin välisen kontrastieron on oltava riittävä, jotta sivuston käyttö helpottuu käyttäjille, joilla on suhteellisen huono näkö ja jotka eivät käytä avustavia teknologioita. AAA-tason ohjeidenmukaisuuteen pyrkiville sivuille kontrastivaatimukset ovat tiukemmat kuin AA-tasolla. Suurikokoiselle tekstille, joka on WCAG:ssa määritelty olevan 18 pistettä tai 14 pistettä lihavoituna, kontrastivaatimukset ovat lievemmat. Kontrastivaatimuksia ei ole logoille, yksinomaan koristeena olevalle tekstille ja kuvissa oleville teksteille, jos kuvassa on muutakin sisältöä. Yksinomaan tekstiä esittäville kuville kontrastivaatimukset pätevät normaalisti. (W3C, 2021-o)

Kontrastivaatimuksia on asetettu myös käyttöliittymäkomponenteille, kuten lomakkeen kentille ja nappuloille, tosin ne ovat kevyempiä kuin tekstille. Kontrastivaatimukset koskevat aktiivisia käyttöliittymäkomponentteja ja niiden tilasta kertovia tyyliä, kuten kohdistuksen siirtyessä komponenttiin aktivoituvaa tyyliä. Epäaktiivisia käyttöliittymäkomponentteja kontrastivaatimukset eivät koske, kuten eivät myöskään värejä, jotka eivät häiritse komponentin erottamisessa. Kuvan 3 esimerkin valkoinen tekstikenttä vihreällä taustalla täyttää kontrastivaatimukset tummanharmaan reunan huonosta erottuvuudesta huolimatta, sillä se ei häiritse komponentin erottamista. (W3C, 2021-p)

Kuva 3: Kontrastikriteerit käyttävä tekstikenttä

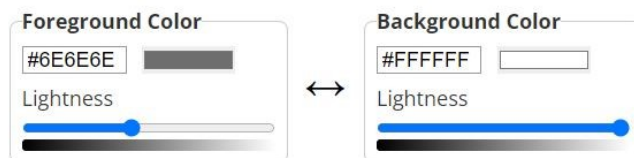


Verkossa on useita työkaluja, joilla kontrastin ohjeidenmukaisuuden voi tarkastaa. Kuvassa 4 esitelty WebAIM:in työkalu on yksi suosituimmista ja se näyttää suoraan, täyttääkö pää- ja taustavärien välinen kontrasti AA- ja AAA-tasojen vaatimukset. WebAIM:in kontrastilaskuri on käytettävissä osoitteessa <https://webaim.org/resources/contrastchecker/>. Joissain työkaluissa, kuten Acart Communicationsin kontrastilaskurissa, on tämän perustoiminnallisuuden lisäksi mahdollisuus katsoa värillistä kuvaa harmaasävyssä ja mahdollisuus valita tutkittavat väriarvot suoraan liitetystä kuvasta.

Kuva 4: WebAIM:in kontrastintarkastustyökalu

Contrast Checker

[Home](#) > [Resources](#) > Contrast Checker



Contrast Ratio
5.09:1

[permalink](#)

Normal Text

WCAG AA: **Pass**

WCAG AAA: **Fail**

The five boxing wizards jump quickly.

Large Text

WCAG AA: **Pass**

WCAG AAA: **Pass**

The five boxing wizards jump quickly.

Graphical Objects and User Interface Components

WCAG AA: **Pass**

Text Input ✓

5.5.2 Tekstivastineet

On ensiarvoisen tärkeää, että kaikella sivun ei-tekstuaalisella sisällöllä, kuten kuvilla, taulukoilla, lomakkeilla ja animaatioilla, on tekstivastine. Tällöin avustavat teknologiat, kuten ruudunlukijat, havaitsevat sisällön ja voivat sitten esimerkiksi lukea sen käyttäjälle ääneen tai kääntää pistekirjoitukseksi. Kuulorajoitteisille käyttäjille tulee tarjota tekstitys videoihin ja äänitiedostoihin. (W3C, 2021-q)

Kuva 5: Tekstivastine Ylen uutiskuvassa

Talousmahtien johtajat sitoutuivat tavoittelemaan ilmaston lämpenemisen rajoittamista 1,5 asteeseen – "vaatii merkittäviä ja tehokkaita toimia kaikilta mailta"

Pariisin ilmastositoumuksessa sovittiin tavoitteesta pitää maapallon keskilämpötilan nousu selvästi alle kahdessa asteessa suhteessa esiteolliseen aikaan ja pyrkimyksestä rajata nousu alle 1,5 asteeseen.



alt="Tehtaan piipuista nousee savua Kiinassa."

Käytännössä tekstivastineen lisäämistapa riippuu kuvailtavasta html-elementistä. Kuvien tekstivastineeksi riittää kuvaava alt-attribuuttiin lisätty teksti, kuten kuvassa 5 on tehty. Se, mitä

tekstin sisältöön tulee, riippuu kontekstissa, jossa kuvaa on käytetty. Uutis- ja kuvituskuviin riittää lyhyt kuvaus kuvan sisällöstä, mutta jos kuvaa käytetään yhteydessä, jossa sen sisältö merkitsee enemmän tai se on esimerkiksi taulukko, voidaan käyttää longdesc-attribuuttia, jolla linkitetään pitempään kuvaukseen joko samalla sivulla erillisessä elementissä tai ulkoisella sivulla. Pidemmän kuvauksen voi myös kirjoittaa sivulle leipätekstiin ja liittää sen sitten kuvaan aria-describedby-attribuutilla tai kirjoittamalla alt-attribuuttiin esimerkiksi, että tarkempi kuvaus on tekstissä taulukon jälkeen. (W3C, 2021-q)

Käyttöliittymäelementtien ja lomakkeiden tekstivastineet voi toteuttaa label-elementeillä seuraavasti:

```
<label for="etunimi">Etunimi:</label>
<input type="text" name="etunimi" id="etunimi" />
```

Vaihtoehtoinen tapa on käyttää aria-label- tai aria-describedby-attribuutteja. Aria-labelia käytetään, kun halutaan kirjoittaa kokonaan uusi tekstivastine ja aria-describedby:tä tai aria-labelledby:tä silloin, kun halutaan käyttää sivulla jo olevaa tekstiä. Aria-labelledby on tarkoitettu nimilapun liittämiseen objektiin yllä olevan esimerkin for-attribuutin tavoin, kun taas aria-describedby tarjoaa käyttäjälle hyödyllistä lisätietoa. On huomionarvoista, että aria-labelin tuki erityisesti vanhemmilla selaimilla ja avustavilla teknologioilla on paikoin puutteellista. (Mozilla, 2021-a; Mozilla, 2021-b) Attribuutit toimivat seuraavasti:

```
<button aria-label="Jatka" aria-describedby="descriptionContinue"
  onclick="myDialog.continue()">X</button>
[...]
<div id=" descriptionContinue">Painamalla jatka-painiketta vahvistat lukeneesi ja hyväksyneesi
käyttöehdot ja siirryt lataussivulle</div>
```

Ääni- ja videotiedostojen saavutettavuus kuuroille ja heikkokuuloisille paranee tekstityksen lisäämisellä. Kuulorajoitteisille tarjotussa tekstityksessä tulisi tavallisista tekstityksistä poiketen tarjota puheen lisäksi myös tieto siitä, kuka puhuu, ja sisällyttää myös muita videon ääniä sopivien ääniefektien tai kuvailun kautta. Jos tekstityksen lisää myös sivulle tekstinä, on puhtaaksi kirjoitettu äänisisältö saavutettavaa myös silloin, jos käyttäjällä on kuulovaikeuksien lisäksi myös näkörajoitteita, sillä nyt se on avustavien teknologioiden käytössä ja voidaan kääntää esimerkiksi pistekirjoitukseksi. (W3C, 2021-r)

YouTuben automaattinen tekstitys toimii suhteellisen hyvin selvästi puhutulle englannille ja se toimii myös YouTube-upotuksissa, mutta tekstitys on syytä tarkistaa ja manuaalinen tekstityksen kirjoittaminen on toistaiseksi kannattavampaa. Muille alustoille tehtäessä tekstitysten luomista varten on tarjolla useita tekstitysten generointiin tarkoitettuja työkaluja. Tekstitystiedoston voi liittää videoon ohjelmallisesti natiivi-HTML:ssä tai vaihtoehtoisesti voi käyttää ulkoista mediatoistin-pluginia, joissa on tuki yleisimmille tekstitystiedostoille.

Spämmirobotteja vastaan käytetään usein CAPTCHA-tehtäviä, jonka ratkaisemalla käyttäjä todistaa olevansa ihminen. Ne ovat pohjimmiltaan saavutettavuuden kannalta haastavia, sillä tekstivastineen tarjoaminen mahdollistaisi CAPTCHA:n ratkaisemisen ohjelmallisesti, mikä tekisi siitä hyödyttömän. On suositeltavaa tarjota usampi erityyppinen CAPTCHA tai vaihtoehtoinen tunnistautumistapa, kuten biometrinen tunnistautuminen eli esimerkiksi sormenjälki tai kasvontunnistus. Näin useimmat käyttäjät, joilla on rajoitteita, löytäisivät vaihtoehdon, jota he pystyisivät käyttämään. (W3C, 2019)

Tekstiä kuvaavia kuvia ei saavutettavuussyistä suositella, vaan tekstin muotoiluun tulisi pyrkiä käyttämään CSS:ää. Heikkonäköisillä käyttäjillä voi olla vaikeuksia tulkita tekstiä kuvaan asetetulla fontilla, fonttikoolla tai värillä. Kuitenkin mikäli tekstin muotoilu halutunlaiseksi ei syystä tai toisesta ole mahdollista tai esimerkiksi halutaan käyttää fonttia, joka ei ole yleisesti saatavilla, voidaan tekstiä kuvaavaa kuvaa käyttää. Tekstiä sisältävät logot ovat myös sallittuja poikkeuksia. Tällöin on huolehdittava siitä, että kuvilla on tekstivastineet. (W3C, 2021-s)

5.5.3 Otsikot ja otsikkotasot

WebAIM:in tekemän kyselytutkimuksen (2021) perusteella otsikoiden selaaminen on ruudunlukijoiden käyttäjien keskuudessa ylivoimaisesti suosituin tapa selata verkkosivuja. Tämän takia onkin erityisen tärkeää, että otsikot ovat kuvaavia ja ruudunlukijan havaittavissa. Käyttäjät myös pitävät asianmukaista otsikkorakennetta erittäin hyödyllisenä. Tämä tarkoittaa sitä, että sivulla käytetään eri otsikkotasoja sisällön jakamiseen.

Käytännössä tämä tarkoittaa sitä, että sivulla on käytetty <h1>- , <h2>- ja <h3>-elementtejä oikein. Ruudunlukija tunnistaa html:n otsikkoelementit automaattisesti otsikoiksi, joten niiden käyttö on suositeltavaa. Jos sivun otsikot on toteutettu jotenkin muuten, kuten esimerkiksi <div>-elementtien avulla, eikä niiden muuttaminen html:n natiiveiksi otsikkoelementeiksi syystä tai

toisesta ole mahdollista, pitää ruudunlukija saada tunnistamaan ne otsikoiksi käyttämällä ARIA-roolia heading. Sen yhteydessä käytettävä aria-level-attribuutti määrittelee, minkä tason otsikko on kyseessä <h1>- , <h2>- ja <h3>-elementtejä vastaavalla tavalla. Jos aria-level jätetään tyhjäksi, tulkitaan otsikko automaattisesti <h2>-tasoiseksi. (Mozilla, 2021-c)

```
<div role="heading" aria-level="1">Sivun pääotsikko</div>
```

5.5.4 Maamerkit ja hyppylinkit

Maamerkit ovat WAI-ARIA:n tarjoama tapa erotella sivun rakennetta ja toistuvia elementtejä navigoinnin helpottamiseksi ja nopeuttamiseksi avustavilla teknologioilla. Tällaisia ovat esimerkiksi navigointivalikko, pääsisältö ja verkkolomakkeet. Maamerkkeihin voi navigoida näppäimistöllä ja ruudunlukijan pitäisi ilmoittaa niiden olevan maamerkkejä sivua selatessa. (Mozilla, 2021-d)

Koodissa maamerkit ovat ARIA-rooleja. Maamerkit sisältyvät myös HTML5-elementteihin, esimerkiksi <header>-tagilla on automaattisesti maamerkkirooli banner, <nav>-elementillä navigation ja <main>-elementillä main-maamerkki. HTML5-elementtien oikea käyttö sivulla on suositeltu tapa saada maamerkit käyttöön, (Mozilla, 2021-d) mutta mikäli tämä ei jostain syystä ole mahdollista, maamerkkirooleja käytetään seuraavasti:

```
<div role="banner">
  
  <h1>Sivun otsikko</h1>
</div>
<div role="main">
  <h2>Otsikko</h2>
  <!-- pääsisältö -->
</div>
```

Maamerkkien käyttö sivulla navigointiin on ollut useamman vuoden ajan laskussa ruudunlukijakäyttäjien keskuudessa. Noin neljännes ruudunlukijakäyttäjistä käyttää maamerkkejä usein navigointiin, kun vuonna 2014 osuus oli vielä lähes puolet. On arvioitu, että syynä maamerkkien suosion vähenemiseen on niiden vähäinen tai virheellinen käyttö sivuilla ja muiden navigointitapojen paraneminen. (WebAIM, 2021)

Maamerkkeihin liittyy myös hyppylinkki, jonka tarkoituksena on antaa sivua näppäimistöllä ja ruudunlukijalla navigoivalle käyttäjälle suora ja kätevä tapa päästä suoraan sivun pääsisältöön ja

ohittaa sivun alussa toistuvat elementit, kuten header, navigointilinkit ja mahdolliset mainosbannerit. Suositettu tapa kriteerin täyttämiseksi on sivun alkuun piilotettu linkki, jota klikkaamalla päätyy suoraan sivun pääsisältöön. Maamerkkien avulla navigoivalle tämä täyttyy käyttämällä main-maamerkkiä asianmukaisessa paikassa. (W3C, 2021-t)

6 Yhteenveto

Saavutettavuusasiat voivat olla yllättävän hankalia hahmottaa käyttäjälle tai kehittäjälle, jolla ei ole verkon käyttöä haittaavia rajoitteita. Asiat, joihin ei tule edes kiinnittäneeksi huomiota, voivat olla joillekin käyttäjille sivuston käytön estäviä ongelmia. WCAG-saavutettavuuskriteerit käsittelevät hyvin laajasti erilaisia saavutettavuusongelmia ja niitä noudattamalla mahdollistetaan itsenäinen verkkopalveluiden käyttö mahdollisimman monelle käyttäjälle.

Yllättävän monet WCAG:n saavutettavuuskriteerit täyttyvät jo HTML:n oikeaoppisella käytöllä. Avustavat teknologiat tunnistavat html-elementit valtaosin oikein ja jo valmiiksi hyvin koodattujen, selkeiden ja toimintavarmojen sivujen muokkaaminen erityisesti matalempien WCAG-tasojen mukaisiksi ei usein vaadi suurta työtä. On myös syytä muistaa, että saavutettavuuden huomiointi verkkosivuilla parantaa käytettävyyttä laajemminkin. Saavutettavista verkkopalveluista hyötyvät kaikki käyttäjät.

Lähteet

Aluehallintovirasto. (n.d.-a). *Yleistä saavutettavuudesta*.

<https://www.saavutettavuusvaatimukset.fi/yleista-saavutettavuudesta/>

Aluehallintovirasto. (n.d.-b). *Kenelle saavutettavuus on tärkeää?*

<https://www.saavutettavuusvaatimukset.fi/yleista-saavutettavuudesta/kenelle-saavutettavuus-on-tarkeaa/>

Aluehallintovirasto. (n.d.-c). *WCAG 2.1: lain vaatimukset*.

<https://www.saavutettavuusvaatimukset.fi/digipalvelulain-vaatimukset/wcag-2-1/>

Kehitysvammaliitto. (n.d.-a). *Kuka hyötyy saavutettavuudesta?* Papunet Saavutettavuus.

<https://papunet.net/saavutettavuus/kuka-hyotyy-saavutettavuudesta>

Kehitysvammaliitto. (n.d.-b). *Näköön liittyvät rajoitteet*. Papunet Saavutettavuus.

<https://papunet.net/saavutettavuus/nakoon-liittyvat-rajoitteet>

Kehitysvammaliitto. (n.d.-c). *Kuuloon liittyvät rajoitteet*. Papunet Saavutettavuus.

<https://papunet.net/saavutettavuus/kuuloon-liittyvat-rajoitteet>

Kehitysvammaliitto. (n.d.-d). *Fyysiset ja motoriset rajoitteet*. Papunet Saavutettavuus.

<https://papunet.net/saavutettavuus/fyysiset-ja-motoriset-rajoitteet>

Kehitysvammaliitto. (n.d.-e). *Kognitiiviset ja kielelliset vaikeudet*. Papunet Saavutettavuus.

<https://papunet.net/saavutettavuus/kognitiiviset-ja-kielelliset-vaikeudet>

Kehitysvammaliitto. (n.d.-f). *Ruudunlukuohjelmat*. Papunet Saavutettavuus.

<https://papunet.net/saavutettavuus/ruudunlukuohjelmat>

Kehitysvammaliitto. (n.d.-g). *Saavutettavuuden arvioinnin suunnitelma*. Papunet Saavutettavuus.

<https://papunet.net/saavutettavuus/saavutettavuuden-arvioinnin-suunnitelma>

Kehitysvammaliitto. (n.d.-h). *Saavutettavuuden testaaminen itse*. Papunet Saavutettavuus.

<https://papunet.net/saavutettavuus/saavutettavuuden-testaaminen-itse>

Laki digitaalisten palvelujen tarjoamisesta 306/2019.

<https://www.finlex.fi/fi/laki/alkup/2019/20190306#Pidm45237815904656>

Mozilla. (6.10.2021-a). *Using the aria-labelledby attribute*. MDN Web Docs.

https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/ARIA_Techniques/Using_the_aria-labelledby_attribute

Mozilla. (3.10.2021-b). *Using the aria-describedby attribute*. MDN Web Docs.

https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/ARIA_Techniques/Using_the_aria-describedby_attribute

Mozilla. (7.11.2021-c). *ARIA: heading role*. MDN Web Docs. [https://developer.mozilla.org/en-](https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Roles/heading_role)

[US/docs/Web/Accessibility/ARIA/Roles/heading_role](https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Roles/heading_role)

Mozilla. (7.11.2021-d). *ARIA: landmark role*. MDN Web Docs. [https://developer.mozilla.org/en-](https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Roles/landmark_role)

[US/docs/Web/Accessibility/ARIA/Roles/landmark_role](https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Roles/landmark_role)

W3C. (7.10.2016). *Understanding Conformance*. [https://www.w3.org/TR/UNDERSTANDING-](https://www.w3.org/TR/UNDERSTANDING-WCAG20/conformance.html)

[WCAG20/conformance.html](https://www.w3.org/TR/UNDERSTANDING-WCAG20/conformance.html)

W3C. (14.12.2017). *Accessible Rich Internet Applications (WAI-ARIA) 1.1*.

<https://www.w3.org/TR/wai-aria-1.1/>

W3C. (5.6.2018). *Web Content Accessibility Guidelines (WCAG) 2.1*.

<https://www.w3.org/TR/WCAG21/>

W3C. (9.12.2019). *Inaccessibility of CAPTCHA*. <https://www.w3.org/TR/turingtest/>

W3C. (4.8.2021-a). *What's New in WCAG 2.2 Working Draft*. Haettu 23.11.2021 osoitteesta

<https://www.w3.org/WAI/standards-guidelines/wcag/new-in-22/>

W3C. (8.6.2021-b). *WCAG 3 Introduction*. Haettu 23.11.2021 osoitteesta

<https://www.w3.org/WAI/standards-guidelines/wcag/wcag3-intro/>

W3C. (27.7.2021-c). *Understanding Success Criterion 2.1.1: Keyboard*.

<https://www.w3.org/WAI/WCAG21/Understanding/keyboard.html>

W3C. (27.7.2021-d). *Understanding Success Criterion 2.1.2: No Keyboard Trap*.

<https://www.w3.org/WAI/WCAG21/Understanding/no-keyboard-trap.html>

- W3C. (27.7.2021-e). *Understanding Success Criterion 2.4.3: Focus Order*.
<https://www.w3.org/WAI/WCAG21/Understanding/focus-order.html>
- W3C. (27.7.2021-f). *Understanding Success Criterion 1.4.4: Resize text*.
<https://www.w3.org/WAI/WCAG21/Understanding/resize-text.html>
- W3C. (27.7.2021-g). *Understanding Success Criterion 1.4.10: Reflow*.
<https://www.w3.org/WAI/WCAG21/Understanding/reflow.html>
- W3C. (27.7.2021-h). *Understanding Success Criterion 3.3.1: Error Identification*.
<https://www.w3.org/WAI/WCAG21/Understanding/error-identification.html>
- W3C. (27.7.2021-i). *Understanding Success Criterion 3.3.3: Error Suggestion*.
<https://www.w3.org/WAI/WCAG21/Understanding/error-suggestion.html>
- W3C. (27.7.2021-j). *Understanding Success Criterion 4.1.3: Status Messages*.
<https://www.w3.org/WAI/WCAG21/Understanding/status-messages.html>
- W3C. (27.7.2021-k). *Understanding Success Criterion 2.2.1: Timing Adjustable*.
<https://www.w3.org/WAI/WCAG21/Understanding/timing-adjustable.html>
- W3C. (27.7.2021-l). *Understanding Success Criterion 2.2.6: Timeouts*.
<https://www.w3.org/WAI/WCAG21/Understanding/timeouts.html>
- W3C. (15.11.2021-m). *Understanding Success Criterion 3.3.7: Accessible Authentication*. Haettu
23.11.2021 osoitteesta <https://www.w3.org/WAI/WCAG22/Understanding/accessible-authentication>
- W3C. (27.7.2021-n). *Understanding Success Criterion 1.4.1: Use of Color*.
<https://www.w3.org/WAI/WCAG21/Understanding/use-of-color.html>
- W3C. (27.7.2021-o). *Understanding Success Criterion 1.4.3: Contrast (Minimum)*.
<https://www.w3.org/WAI/WCAG21/Understanding/contrast-minimum.html>
- W3C. (27.7.2021-p). *Understanding Success Criterion 1.4.11: Non-text Contrast*.
<https://www.w3.org/WAI/WCAG21/Understanding/non-text-contrast.html>

W3C. (27.7.2021-q). *Understanding Success Criterion 1.1.1: Non-text Content.*

<https://www.w3.org/WAI/WCAG21/Understanding/non-text-content.html>

W3C. (27.7.2021-r). *Understanding Success Criterion 1.2.2: Captions (Prerecorded).*

<https://www.w3.org/WAI/WCAG21/Understanding/captions-prerecorded.html>

W3C. (27.7.2021-s). *Understanding Success Criterion 1.4.5: Images of Text.*

<https://www.w3.org/WAI/WCAG21/Understanding/images-of-text.html>

W3C. (27.7.2021-t). *Understanding Success Criterion 2.4.1: Bypass Blocks.*

<https://www.w3.org/WAI/WCAG21/Understanding/bypass-blocks.html>

WebAim. (30.6.2021). *Screen Reader User Survey #9 Results.*

<https://webaim.org/projects/screenreadersurvey9/>