



Lista kontrolna WCAG 2.1

dla poziomu A oraz AA

Poniższa lista kontrolna ma służyć podsumowaniu oraz ułatwić pracę twórcom stron oraz redaktorom treści, których wspólnym celem jest poprawa dostępności stron internetowych.

Lista kontrolna

Postrzegalność

1.1.1 Treść nietekstowa (A)	Redaktor treści	<input type="checkbox"/>
<p>Treści, które nie są tekstem elektronicznym, takie jak m.in. ilustracje, diagramy, fotografie muszą posiadać alternatywny opis w postaci atrybutu ALT obrazka.</p>		
1.2.1 Tylko audio lub tylko wideo (nagranie) (A)	Redaktor treści	<input type="checkbox"/>
<p>Samo audio (np. podcast, wywiad radiowy, itp.) – wymagana jest transkrypcja nagrania; Wideo (bez dźwięku) lub pokaz slajdów – wymagana jest alternatywa w postaci tekstu lub audiodeskrypcja opisująca co przedstawia wideo.</p>		
1.2.2 Napisy rozszerzone (nagranie) (A)	Redaktor treści	<input type="checkbox"/>
<p>Zapewnij napisy do nagrań nietransmitowanych na żywo. Powinny uwzględniać zarówno dialogi, jak i istotne informacje dźwiękowe. Dobrą praktyką jest defaultowe włączenie napisów, z możliwością ukrycia ich.</p>		
1.2.3 Audiodeskrypcja lub alternatywa tekstowa dla mediów (nagranie) (A)	Redaktor treści	<input type="checkbox"/>
<p>Przygotuj audiodeskrypcję, czyli nagranie audio opisujące treści wizualne. Drugim rozwiązaniem jest opracowanie alternatywy dla mediów opartych na czasie w postaci tekstu. Treść powinna zawierać zarówno transkrypcję dialogów jak i opis wszystkich istotnych elementów wizualnych i dźwiękowych nagrania.</p>		
1.2.4 Napisy rozszerzone (na żywo) (AA)	Redaktor treści	<input type="checkbox"/>

<p>Zapewnij napisy do nagrań transmitowanych na żywo. Powinny uwzględniać zarówno dialogi, jak i istotne informacje dźwiękowe. Dobrą praktyką jest defaultowe włączenie napisów, z możliwością ukrycia ich.</p>	
<p>1.2.5 Audiodeskrypcja (nagranie) (AA) Redaktor treści</p> <p>Opracuj audiodeskrypcję i dodaj ją do głównej ścieżki dźwiękowej. W skrócie audiodeskrypcją możemy nazwać nagranie audio opisujące treści wizualne. To kryterium służy osobom z dysfunkcjami narządu wzroku zrozumieć treści wideo.</p>	<input type="checkbox"/>
<p>1.3.1 Informacje i relacje (A) Programista</p> <p>Wykorzystuj znaczniki semantyczne zgodnie z ich przeznaczeniem (między innymi po to, by czytnik ekranu właściwie i po kolei odczytywał treści znajdujące się na stronie).</p>	<input type="checkbox"/>
<p>1.3.2 Zrozumiała kolejność (A) Programista</p> <p>Zapewnij zrozumiałą kolejność elementów na stronie również po wyłączeniu stylu CSS. Użytkownicy korzystający z czytnika ekranu lub korzystający jedynie z klawiatury powinni móc odbierać treści w logicznej kolejności.</p>	<input type="checkbox"/>
<p>1.3.3 Właściwości zmysłowe (A) Designer</p> <p>Elementy nawigacji nie mogą opierać się tylko na wrażeniach zmysłowych, takich jak kolor, rozmiar, kształt elementu, jego położenie na stronie. Informacje nie powinny być przekazywane tylko i wyłącznie w postaci ikon lub symboli.</p>	<input type="checkbox"/>
<p>1.3.4 Orientacja (AA) Programista & Designer & Tester</p> <p>To kryterium dotyczy wyświetlania się strony na urządzeniach mobilnych. Strona powinna wyświetlać się prawidłowo zarówno w pionie jak i w poziomie.</p>	<input type="checkbox"/>

<p>Są jednak wyjątki, do których możemy zaliczyć aplikację do gry na pianinie. W tym przypadku, układ klawiszy wymusza poziomą orientację urządzenia.</p>	
<p>1.3.5 Określenie pożądanej wartości (AA) Programista</p> <p>Pola wprowadzania danych użytkownika powinny mieć dodany atrybut "autocomplete".</p>	<input type="checkbox"/>
<p>1.4.1 Użycie koloru (A) Designer</p> <p>Kolor nie powinien być jedynym nośnikiem informacji. I tak w przypadku błędnie wypełnionego pola formularza, oprócz podświetlenia w kolorze czerwonym, trzeba również umieścić tekstową informację o błędzie. Kolejnym przykładem są linki. Dobrą praktyką jest oznaczanie hiperłączy poprzez ich podkreślenie, dzięki temu będą wyraźnie widoczne pośród reszty tekstu.</p>	<input type="checkbox"/>
<p>1.4.2 Kontrola odtwarzania dźwięku (A) Designer & Programista</p> <p>Nagrania audio umieszczone na stronie internetowej nie powinny uruchamiać się automatycznie. Jeżeli dźwięk jest odtwarzany przez więcej niż 3 sekundy trzeba umożliwić użytkownikowi wstrzymanie/wyłączenie dźwięku lub regulację głośności. Kontrolka wyciszania dźwięku powinna być łatwo dostępna, aby osoby korzystające z czytnika ekranu lub używające klawiatury z łatwością mogły wyłączyć dźwięk.</p>	<input type="checkbox"/>
<p>1.4.3 Kontrast (minimum) (AA) Designer</p> <p>Tekst i grafiki tekstowe (w postaci plików .svg, .png, itd.) powinny kontrastować z tłem w stosunku 4.5:1. W przypadku użycia dużego tekstu (na przykład 18px) zapewnij minimalny kontrast elementów w stosunku 3:1. Zasada nie obejmuje dekoracyjnych elementów, logotypów, nieaktywnych elementów interfejsu.</p>	<input type="checkbox"/>
<p>1.4.4 Zmiana rozmiaru tekstu (AA) Programista & Designer</p>	<input type="checkbox"/>

<p>Strona internetowa powinna być czytelna i funkcjonalna po podwojeniu rozmiaru tekstu (200%).</p>	
<p>1.4.5 Obrazy tekstu (AA)</p> <p style="text-align: right;">Redaktor treści</p> <p>Jeżeli to możliwe unikaj grafik przedstawiających tekst, a używaj tekstu elektronicznego. Wyjątkiem są logotypy oraz elementy identyfikacji wizualnej. Zapewnij im tekst alternatywny.</p>	<input type="checkbox"/>
<p>1.4.10 Dopasowanie do ekranu (AA)</p> <p style="text-align: right;">Programista & Designer & Tester</p> <p>Stwórz responsywną stronę, której treść będzie wyświetlana również w szerokości 320 px bez utraty funkcjonalności i informacji oraz bez użycia horyzontalnego scrolla. W określonych przypadkach można zastosować przewijanie w poziomie (rozbudowane tabele, karuzela obrazów, mapy itd.)</p>	<input type="checkbox"/>
<p>1.4.11 Kontrast elementów nietekstowych (AA)</p> <p style="text-align: right;">Designer</p> <p>Współczynnik kontrastu nietekstowych elementów interfejsu użytkownika (takich jak ikony, inputy, buttony) do tła powinien wynosić co najmniej 3:1.</p>	<input type="checkbox"/>
<p>1.4.12 Odstępy w tekście (AA)</p> <p style="text-align: right;">Designer</p> <p>Zastosuj odstępy w tekście według poniższych reguł:</p> <ul style="list-style-type: none"> • wysokość linii – co najmniej 1.5 razy większa od rozmiaru fontu, • odstęp między akapitami – co najmniej 2 razy większy od rozmiaru fontu, • odstęp pomiędzy literami – co najmniej 0,12-krotność rozmiaru fontu, • odstęp pomiędzy wyrazami – co najmniej 0,16-krotność rozmiaru fontu. 	<input type="checkbox"/>
<p>1.4.13 Treść spod kursora lub fokusu (AA)</p> <p style="text-align: right;">Programista & Designer & Tester</p>	<input type="checkbox"/>

Kryterium sukcesu określa zachowania elementów interaktywnych aktywowanych kursorem lub fokusem. Hover aktywuje się zarówno po najechaniu kursorem jak i przy nawigacji klawiaturą (klawiszem TAB). Treść powinna zostać odrzucona po wciśnięciu klawisza ESC. Hover elementu znika, kiedy najedziemy kursorem na inny element.

Funkcjonalność

2.1.1 Klawiatura (A)	Programista & Designer & Tester	<input type="checkbox"/>
Wszystkie funkcjonalności strony powinny być dostępne przy użyciu jedynie klawiatury lub czytników emulujących klawiaturę. Użytkownik musi mieć możliwość wejścia w link za pomocą klawisza SPACJI lub ENTER. Pamiętaj, aby cel linków i buttonów był jasno sprecyzowany. Unikaj frazy "Kliknij tutaj".		
2.1.2 Bez pułapki na klawiaturę (A)	Programista & Tester	<input type="checkbox"/>
Upewnij się, że można przenosić fokus klawiatury na elementy interfejsu oraz usuwać fokus elementu również za pomocą klawiatury. Przykładem może być pop-up, który możemy zamknąć za pomocą klawisza ESC.		
2.1.4 Jednoznakowe skróty klawiaturowe (A)	Programista	<input type="checkbox"/>
Jeżeli w treści strony zaimplementowany jest skrót klawiszowy, powinien być aktywny tylko po otrzymaniu fokusu. Zapewnij możliwość zmiany skrótu lub mechanizm wyłączenia go przez użytkownika. Czytniki ekranu również mają swoje skróty klawiszowe, dlatego ważne jest unikanie sytuacji, w której te same skróty będą ze sobą kolidować.		
2.2.1 Dostosowanie czasu (A)	Programista & Designer	<input type="checkbox"/>
Unikaj nakładania limitów czasowych. Jeśli ograniczenie czasowe jest konieczne zapewnij użytkownikowi możliwość zapauzowania, wyłączenia lub wydłużenia limitu. Od tej zasady również są wyjątki. Nie ma potrzeby		

<p>wydłużania limitu jeśli czas na ukończenie zadania wynosi co najmniej 20 godzin. Innym wyjątkiem jest wydarzenie w czasie rzeczywistym, na przykład aukcja internetowa.</p>	
<p>2.2.2 Pauza, zatrzymanie, ukrycie (A) Programista & Designer</p> <p>Elementy na stronie, które poruszają się, animują i migotają dłużej niż 5s i są aktywowane automatycznie, powinno dać się wyłączyć, zapauzować lub ukryć. Jeżeli nie jest to konieczne, unikaj automatycznego odświeżania strony. Zapewnij użytkownikowi komfort korzystania z Twojej strony. :)</p>	<input type="checkbox"/>
<p>2.3.1 Trzy błyski lub wartości poniżej progu (A) Redaktor treści & Designer</p> <p>Kryterium sukcesu skupia się na osobach z epilepsją światłoczułą. Upewnij się, że Twoja strona nie zawiera w treści niczego, co błyska częściej niż trzy razy w ciągu jednej sekundy lub też błysk nie przekracza wartości granicznych dla błysków ogólnych i czerwonych. Jeżeli taki materiał multimedialny koniecznie ma się znaleźć na stronie dodaj wyraźne ostrzeżenie, na przykład: "Materiał zawiera migające obrazy."</p>	<input type="checkbox"/>
<p>2.4.1 Możliwość pominięcia bloków (A) Programista</p> <p>Ułatwiał użytkownikom szybkie dotarcie do treści, jakich szukają. Nawigacja za pomocą klawiatury wymaga przejścia przez każdy interaktywny element, co wydłuża drogę użytkownika chcącego przejść do konkretnego elementu strony. Kryterium jest spełnione jeżeli na stronie istnieje mechanizm, który umożliwia pominięcie powtarzających się bloków treści (na przykład listy linków). Dobrą praktyką jest dodanie na początku strony linku "Przeskocz do głównej treści", który jest widoczny zaraz po wejściu na stronę i wciśnięciu klawisza TAB.</p>	<input type="checkbox"/>
<p>2.4.2 Tytuł strony (A) Redaktor treści & Designer & Programista</p> <p>Każda podstrona serwisu powinna mieć unikalny tytuł, opisujący jej cel lub przedstawiający jej temat.</p>	<input type="checkbox"/>

<p>2.4.3 Kolejność fokusu (A) Programista</p> <p>Przy nawigacji za pomocą klawiatury, fokus aktywuje elementy interfejsu w logicznej kolejności, dzięki której zachowany jest sens treści.</p>	<input type="checkbox"/>
<p>2.4.4 Cel łącza (w kontekście) (A) Programista & Designer</p> <p>Cel łącza (linków, buttonów) powinien wynikać jasno z kontekstu w jakim się znalazło. Unikaj buttonów o treści "Kliknij tutaj". Jeżeli strona otwiera się w nowym oknie przeglądarki koniecznie poinformuj o tym użytkownika (np. w treści odnośnika za pomocą klasy "sr-only").</p>	<input type="checkbox"/>
<p>2.4.5 Wiele dróg (AA) Designer</p> <p>Strona internetowa powinna zapewniać przynajmniej dwa różne sposoby dotarcia do podstron serwisu. Przykłady takich rozwiązań:</p> <ul style="list-style-type: none"> • główne menu w headerze/footerze, • mapa serwisu, • wyszukiwarka, • lista powiązanych stron, • spis treści. <p>Wyjątkiem są na przykład quizy online, w których jest możliwy tylko jeden sposób nawigacji – odpowiedź na zadane pytanie warunkuje przejście do kolejnego ekranu.</p>	<input type="checkbox"/>
<p>2.4.6 Nagłówki i etykiety (AA) Programista & Designer</p> <p>Porządkuj treści za pomocą nagłówków i etykiet. Zarówno jedno jak i drugie powinny być znacznikami semantycznymi i określać temat lub cel treści. Stosuj etykiety do opisu inputów formularzy, zrezygnuj ze stosowania placeholderów w polach formularzy. Pamiętaj, że nagłówki są bardzo istotne z punktu widzenia nawigacji.</p>	<input type="checkbox"/>

<p>2.4.7 Widoczny fokus (A) Programista & Designer</p> <p>Kryterium jest spełnione, gdy każdy interaktywny element zyskuje widoczny fokus przy nawigacji za pomocą klawiatury. Nie polegaj jedynie na domyślnych stylach fokusa przeglądarek, ponieważ część z nich nie spełnia kryteriów dostępności.</p>	<input type="checkbox"/>
<p>2.5.1 Gesty dotykowe (A) Programista & Designer</p> <p>Zapewnij uproszczoną alternatywę (dotyk jednopunktowy) dla opcji wymagających gestów wielopunktowych lub opartych na ścieżkach (takich jak: “szczypanie” ekranu, przeciąganie). Wyjątkiem od reguły jest na przykład składanie elektronicznego podpisu. Przykład rozwiązania: mapka na urządzeniu mobilnym może być powiększona zarówno “szczypaniem” ekranu (gest wielopunktowy), jak i przyciskiem plusa (dotyk jednopunktowy).</p>	<input type="checkbox"/>
<p>2.5.2 Rezygnacja ze wskazania (A) Programista</p> <p>Kryterium odnosi się do klikalnych elementów interfejsu, aktywowanych za pomocą dotyku jednopunktowego. Poniżej opiszę schemat działania na przykładzie przycisku:</p> <ul style="list-style-type: none"> • naciskanie buttona (onMouseDown) nie powoduje przejścia do podstrony, • zjechanie kursorem z elementu (onMouseOut) lub zwolnienie nacisku powoduje przerwanie czynności i przywraca stan sprzed zdarzenia, • kliknięcie buttona (onClick) powoduje przejście do podstrony. 	<input type="checkbox"/>
<p>2.5.3 Etykieta w nazwie (A) Programista & Designer</p> <p>Dostępna nazwa elementu interfejsu – czyli taka, jaką usłyszy użytkownik korzystający z czytnika ekranu powinna być zbieżna z graficznym interfejsem (może być rozwinięciem nazwy buttonu).</p>	<input type="checkbox"/>
<p>2.5.4 Aktywowanie ruchem (A) Programista & Designer</p>	<input type="checkbox"/>

Oprócz opcji aktywowania zdarzenia ruchem urządzenia (np. potrząśnięciem, pochylaniem telefonu) powinien być też możliwy inny sposób aktywacji, na przykład button. Zapewnij użytkownika, że może wyłączyć opcję reagowania na ruch bez utraty funkcjonalności strony.

Zrozumiałość

3.1.1 Język strony (A)

Programista & Redaktor treści

Treść strony powinna być zrozumiała zarówno dla ludzi jak i technologii. Nadaj atrybut lang ze znacznikiem odpowiedniego języka w elemencie HTML (`html lang="pl"`), aby technologie asystujące mogły odczytać treść strony we właściwym języku.

3.1.2 Język części (AA)

Programista & Redaktor treści

Jeżeli na stronie znajdują się treści w innym języku, niż główny język strony, odpowiednio je oznacz. Czytnik ekranu przeczyta je z charakterystyczną dla danego kraju wymową. Jeżeli w tekście pojawia się skrót lub żargon warto na końcu tekstu dać przypis z wyjaśnieniem co on oznacza. Dobrą praktyką jest również dodanie w nawiasie wymowy fonetycznej obco brzmiącego słowa.

3.2.1 Po otrzymaniu fokusu (A)

Programista & Tester

Przyjęcie fokusu przez dowolny komponent interfejsu użytkownika nie powoduje nieoczekiwanej zmiany kontekstu, co oznacza, że sam fokus nie powoduje akcji (dopiero po kliknięciu SPACJI lub ENTER).

3.2.2 Podczas wprowadzania danych (A)

Programista & Designer

Proces wprowadzania danych powinien być przewidywalny. Wypełnianie formularza lub zaznaczenie pola wyboru nie powinno powodować zmiany kontekstu czy automatycznego przesyłania danych, aż do momentu kliknięcia

<p>przez użytkownika w przycisk. Jeżeli funkcjonalność strony wymaga inaczej, użytkownik powinien być z wyprzedzeniem poinformowany o takim działaniu.</p>	
<p>3.2.3 Spójna nawigacja (AA) Programista & Designer</p> <p>Zachowaj spójną kolejność elementów, które powtarzają się na podstronach serwisu. Najlepiej przyzwyczaić użytkownika do pewnego schematu, aby poruszając się po stronie z łatwością mógł odszukać interesujący go element. Wyjątkiem jest sytuacja, w której użytkownik samodzielnie wprowadzi zmiany w układzie treści.</p>	<input type="checkbox"/>
<p>3.2.4 Spójna identyfikacja (AA) Programista</p> <p>Oznaczaj w podobny sposób komponenty, które mają tę samą funkcjonalność. Stosuj konsekwentnie etykiety, nazwy i alternatywy tekstowe.</p>	<input type="checkbox"/>
<p>3.3.1 Identyfikacja błędu (A) Programista & Designer</p> <p>Wyraźnie oznaczaj wymagane pola formularza. Przy identyfikacji błędu, popełnionego przy wprowadzaniu danych, poinformuj użytkownika, który element wymaga poprawy. Identyfikuj błędy za pomocą aria invalid.</p>	<input type="checkbox"/>
<p>3.3.2 Etykiety lub instrukcje (A) Programista & Designer</p> <p>Dobrym rozwiązaniem jest dodanie krótkiej instrukcji na temat wymaganego formatu danych, jakie mają być wprowadzone (na przykład formatu daty). Nie stosuj placeholderów w inpucie, ale dodawaj etykiety do pól formularzy.</p>	<input type="checkbox"/>
<p>3.3.3 Sugestie korekty błędów (AA) Programista & Designer</p> <p>Jeżeli strona wykryje błąd w formacie danych wprowadzonych przez użytkownika, na przykład w postaci braku znaku "@" w polu e-mail, zasugeruj korektę. Wyjątek stanowią sytuacje, w których sugerowanie podpowiedzi zagrażałoby bezpieczeństwu danych (np. w bankowości internetowej) lub zmieniałoby cel treści.</p>	<input type="checkbox"/>

<p>3.3.4 Zapobieganie błędom (prawnym, finansowym, w danych) (AA)</p> <p>Strony, na których użytkownik dokonuje transakcji finansowych, zawiera umowy prawne, modyfikuje dane lub wykonuje testy powinny mieć zapewnione mechanizmy sprawdzania, potwierdzania, poprawiania i cofania działań przez użytkownika.</p>	<p>Designer & Programista & Tester</p>	<p><input type="checkbox"/></p>
---	--	---------------------------------

Solidność

<p>4.1.1 Poprawność kodu (A)</p> <p>Zadbaj o poprawny semantycznie kod HTML i CSS. Sprawdź czy elementy mają znaczniki otwierające i zamykające, czy są zagnieżdżane zgodnie ze specyfikacją i czy nie mają zduplikowanych atrybutów. Przekonaj się czy wszystkie ID są unikalne.</p>	<p>Programista</p>	<p><input type="checkbox"/></p>
<p>4.1.2 Nazwa, rola, wartość (A)</p> <p>Zapewnij elementom HTML odpowiednią nazwę. Jeżeli tworzysz nowe, niestandardowe elementy interfejsu – zdefiniuj ich rolę, nazwę i status.</p>	<p>Programista</p>	<p><input type="checkbox"/></p>
<p>4.1.3 Komunikaty o stanie (AA)</p> <p>Powiadomienia i komunikaty o stanie powinny być oznaczone odpowiednią rolą lub właściwością za pomocą znaczników ARIA. Dzięki temu użytkownik korzystający z czytnika ekranu dostanie komunikat o stanie: błędzie, sukcesie, czy postępie niezależnie od tego gdzie znajduje się na stronie.</p>	<p>Programista</p>	<p><input type="checkbox"/></p>

Skontaktuj się z nami



Marcin Stawowiak
Co-founder, Executive Manager

—

tel. 791 991 797

marcin@smultron.pl