



Walidacja stron internetowych

Na czym polega i jak przebiega walidacja?



Standardy W3C

W3C czyli **Word Wide Web Consortium** to organizacja założona 1 października 1994 roku, która zajmuje się ustanawianiem **standardów tworzenia stron WWW**. Konsorcjum powstało aby powstrzymać proces niepoprawnie tworzonych stron, które źle się wyświetlały na różnych przeglądarkach. Organizacja ta zajmuje się również ocenianiem zawartości stron pod kątem przemocy i pornografii, dbaniem o prawa autorskie oraz bezpieczeństwem w sieci. W3C oferuje bezpłatne walidatory HTML i CSS, dzięki którym możemy sprawdzić czy dana strona jest zgodna ze standardami.

Walidacja formularzy po stronie klienta



Walidacja formularza po stronie klienta

Przed przesłaniem danych do serwera ważne jest upewnienie się, że wszystkie wymagane kontrolki formularza są wypełnione w prawidłowym formacie. Nazywa się to **walidacją formularza po stronie klienta** i pomaga upewnić się, że przesłane dane są zgodne z wymaganiami określonymi w różnych kontrolkach formularza.

Walidacja po stronie klienta to **wstępna kontrola** i ważna cecha dobrego doświadczenia użytkownika; wyłapując nieprawidłowe dane po stronie klienta, użytkownik może je od razu naprawić. Jeśli dane dotrą do serwera, a następnie zostaną odrzucone, zauważalne jest opóźnienie spowodowane podróżą w obie strony do serwera, a następnie z powrotem do strony klienta, aby poinformować użytkownika o konieczności poprawienia danych.



Walidacja formularza po stronie serwera

Jednak walidacja po stronie klienta nie powinna być uważana za wyczerpujący środek bezpieczeństwa! Twoje aplikacje powinny zawsze wykonywać kontrole bezpieczeństwa danych przesłanych przez formularz zarówno **po stronie serwera**, jak i klienta, ponieważ walidacja po stronie klienta jest zbyt łatwa do obejścia, więc złośliwi użytkownicy nadal mogą łatwo wysyłać złe dane do twojego serwera.



Czym jest walidacja formularza?

Gdy wprowadzasz dane, przeglądarka i/lub serwer WWW sprawdzają, czy dane mają poprawny format i mieszczą się w ograniczeniach ustawionych przez aplikację.

Przykładowe komunikaty:

- „To pole jest wymagane” (Nie możesz pozostawić tego pola pustego).
- „Proszę podać numer telefonu w formacie xxx-xxx-xxx” (Aby numer został uznany za prawidłowy, wymagany jest określony format danych).
- „Wprowadź prawidłowy adres e-mail” (wprowadzone dane nie mają prawidłowego formatu).
- „Twoje hasło musi mieć od 8 do 30 znaków i zawierać jedną wielką literę, jeden symbol i cyfrę”. (Twoje dane muszą mieć bardzo konkretny format danych).



Rodzaje walidacji po stronie klienta

- **Wbudowana walidacja formularza** wykorzystuje funkcje walidacji formularza HTML. Ta walidacja zazwyczaj nie wymaga dużo JavaScript. Wbudowana walidacja formularza ma lepszą wydajność niż JavaScript, ale nie jest tak konfigurowalna jak walidacja JavaScript.
- **Walidacja JavaScript** jest kodowana przy użyciu JavaScript. Ta walidacja jest całkowicie konfigurowalna, ale trzeba ją stworzyć (lub użyć biblioteki).



Wbudowana walidacja formularzy

Jedną z najważniejszych cech [nowoczesnych kontrolek formularzy](#) jest możliwość walidacji większości danych użytkownika bez polegania na JavaScript. Odbywa się to za pomocą atrybutów walidacji w elementach formularza.

- **required:** pole musi zostać wypełnione przed wysłaniem formularza,
- **minlength** i **maxlength:** minimalna i maksymalna długość danych tekstowych,
- **min** i **max:** wartości minimalne i maksymalne typów danych wejściowych,
- **type:** wstępnie ustalony typ: liczba, adres e-mail itp.
- **pattern:** wyrażenie regularne definiujące wzorzec, według którego muszą być przestrzegane wprowadzane dane.

Atrybuty związane z walidacją



Dane prawidłowe

Jeśli dane wprowadzone do pola formularza spełniają wszystkie reguły określone przez powyższe atrybuty, uznaje się je za **prawidłowe**. Jeśli nie, uznaje się je za **nieprawidłowe**.

Jeśli element jest **prawidłowy**, prawdziwe są następujące warunki:

- Element pasuje do pseudoklasy **:valid**, która umożliwia zastosowanie określonego stylu do prawidłowych elementów.
- Jeśli użytkownik spróbuje wysłać dane, przeglądarka wyśle formularz, o ile nic jej tego nie uniemożliwi (np. JavaScript).



Dane nieprawidłowe

Jeśli element jest nieprawidłowy, spełnione są następujące warunki:

- Element pasuje do pseudoklasy **:invalid**, a czasem do innych pseudoklas interfejsu użytkownika (np. :out-of-range), w zależności od błędu, co umożliwia zastosowanie określonego stylu do nieprawidłowych elementów.
- Jeśli użytkownik spróbuje wysłać dane, przeglądarka zablokuje formularz i wyświetli komunikat o błędzie.

Przykłady wbudowanej walidacji formularza



Walidacja względem wyrażenia regularnego

Inną przydatną funkcją walidacji jest atrybut **pattern**, który oczekuje wyrażenia regularnego jako swojej wartości. Wyrażenie regularne (regexp) to wzorzec, którego można użyć do dopasowania kombinacji znaków w ciągach tekstowych, więc regexp są idealne do walidacji formularzy i służą do wielu innych zastosowań w JavaScript.

Kilka przykładów:

- `a`— dopasowuje jeden znak, którym jest `a` (nie `b`, nie `aa`, itd.)
- `ab?c`— dopasowuje `a`, opcjonalnie z pojedynczym `b`, po którym następuje `c`
- `ab*c`— pasuje do `a`, opcjonalnie po którym następuje dowolna liczba `b`, po których następuje `c`. (`ac`, `abc`, `abbbbbbc` itd.)
- `a|b`— dopasowuje jeden znak, którym jest `a` lub `b`.

Walidacja HTML i CSS



Jakie elementy są sprawdzane podczas walidacji?

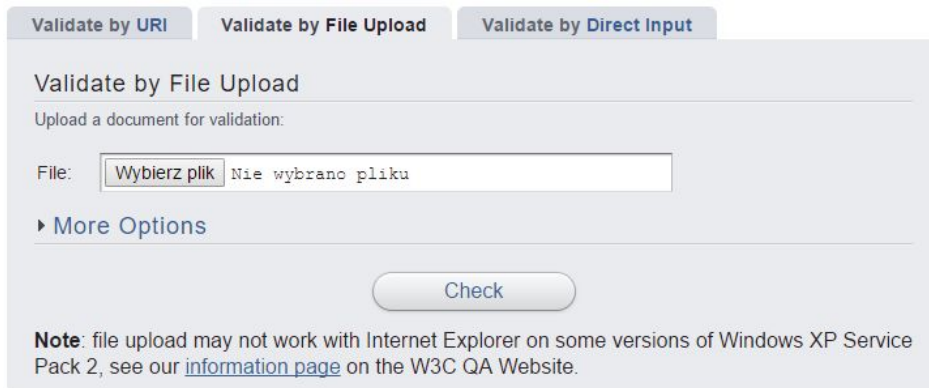
- znaczniki HTML – czy istnieją, czy są odpowiednio otwarte oraz zamknięte,
- obecność kodu niezbędnego na stronie, takiego jak metadane, title, body, head,
- obecność atrybutów alt przy zdjęciach,
- obecność przestarzałych znaczników,
- prawidłowe zastosowanie atrybutów HTML – ID i class,
- struktura kaskadowych arkuszy stylów,
- błędy w plikach .css takie jak występowanie przestarzałych lub niepoprawnych właściwości,



Przykładowe podstawowe walidatory

- <https://jigsaw.w3.org/css-validator/> – do walidacji CSS,
- <https://validator.w3.org/> – do walidacji HTML5 i innych formatów bazujących na znacznikach
- rozszerzenie Visual Studio Code

Proces walidacji



The screenshot shows the W3C Validation Service interface. At the top, there are three tabs: 'Validate by URI', 'Validate by File Upload' (which is selected), and 'Validate by Direct Input'. Below the tabs, the 'Validate by File Upload' section is active. It contains the text 'Upload a document for validation:' followed by a file selection area. The file selection area includes a 'File:' label, a button labeled 'Wybierz plik' (Choose file), and a text box that says 'Nie wybrano pliku' (No file selected). Below this, there is a link 'More Options'. At the bottom of the section is a 'Check' button. A note at the bottom states: 'Note: file upload may not work with Internet Explorer on some versions of Windows XP Service Pack 2, see our [information page](#) on the W3C QA Website.'

Walidacja na kilka sposobów:

- poprzez przesłanie pliku z rozszerzeniem HTML,
- poprzez podanie adresu URL do naszej witryny z możliwością wyboru typu dokumentu oraz kodowania znaków,
- poprzez wklejenie kodu w pole tekstowe

Błędy walidacji

index.html

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title></title>
```

```
  </head>
```

```
  <body>
```

```
    <h1>Przykładowa strona</h1>
```

```
  </body>
```

```
</html>
```

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for uploaded file index.html

Checker Input

Show ☐ source ☐ outline ☐ image report

Check by Nie wybrano pliku

Uploaded files with .xhtml or .xht extensions are parsed using the XML parser.

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

1. **Error** The character encoding was not declared. Proceeding using `windows-1252`.

2. **Error** Element `title` must not be empty.

From line 4, column 10; to line 4, column 17

`<title></title></he`

Document checking completed.

Used the HTML parser.

Total execution time 2 milliseconds.



Cele walidacji stron internetowych

- **Ranking Google** — problem z poprawnym odczytywaniem zawartości strony i indeksowaniem w wyszukiwarce
- **Zadowolenie użytkowników** — zminimalizowanie wystąpienia błędów
- **Szybkość ładowania** — błędna składnia utrudnia budować strony z dokumentu HTML przeglądarce, która próbuje odnaleźć błąd i go naprawić
- **Poprawność wyświetlania** — jeśli kod strony napisany jest zgodnie ze standardem to wyświetli się ona w ten sam sposób w każdej przeglądarce



Cele walidacji stron internetowych cd.

- **Poprawne działanie na różnych urządzeniach** – strony internetowe są odwiedzane na różnych urządzeniach
- **Zapobieganie błędom w przyszłości** – nowe przeglądarki będą musiały pracować zgodnie ze standardami W3C
- **Uproszczony rozwój i utrzymanie** – zmiany na stronach z poprawnym kodem są szybkie w realizacji
- **Szukanie błędów** – kiedy strona przestała wyświetlać się poprawnie należy zweryfikować poprawności kodu