

La validation de formulaire avec Bootstrap 5



La validation des formulaires est l'un des facteurs les plus critiques lorsqu'il s'agit de collecter des informations et de prévenir l'utilisation abusive des formulaires Web.

En plus de vous protéger des cyber-attaques, la validation des formulaires protège également votre site web contre les saisies erronées.

Lors de la soumission du formulaire avec validation, l'utilisateur aura la possibilité de modifier les champs contenant des données invalides avant la soumission.

En général, vous avez besoin de la validation de formulaire comme mesure de sécurité, car les données de formulaire invalidées sont l'une des *principales sources de vulnérabilités de sécurité* des sites Web.

Principe général

Bootstrap 5 utilise les fonctionnalités de validation native du HTML5, enrichies par des styles CSS et des classes utilitaires pour fournir des retours visuels clairs aux utilisateurs lors de la saisie de formulaires. La validation repose principalement sur les pseudo-classes CSS `:valid` et `:invalid`, appliquées aux champs de formulaire (`<input>`, `<select>`, `<textarea>`) selon leur état de validité.

Mise en place de la validation

1. Structure de base

- Ajoutez la classe `.needs-validation` ou `.was-validated` à votre balise ``.
- Utilisez l'attribut HTML5 `required` sur les champs à valider.
- Ajoutez des éléments `div` et `div` pour afficher les messages de validation.

Exemple de formulaire :

```
<form class="needs-validation" novalidate>
  <div class="mb-3">
    <label for="uname" class="form-label">Username:</label>
    <input type="text" class="form-control" id="uname" required>
    <div class="valid-feedback">Valid.</div>
    <div class="invalid-feedback">Please fill out this field.</div>
  </div>
  <button type="submit" class="btn btn-primary">Submit</button>
</form>
```

- L'attribut `novalidate` désactive la validation native du navigateur, permettant à Bootstrap de gérer l'affichage des messages.

2. Activation de la validation avec JavaScript

Pour déclencher l'affichage des styles de validation après soumission, utilisez ce script recommandé par Bootstrap:

```
// Ce bloc crée une fonction auto-exécutée (IIFE) qui isole le code pour éviter les conflits de
((() => {
  'use strict';
  const forms = document.querySelectorAll('.needs-validation');

  Array.from(forms).forEach(form => {

    form.addEventListener('submit', event => {

      // La méthode checkValidity() est une méthode
      // native JavaScript disponible sur les éléments de formulaire
      if (!form.checkValidity()) {

        // Elle empêche le comportement par défaut de l'événement.
        // Dans le cas d'un formulaire, cela empêche l'envoi du formulaire
        // et donc le rechargement de la page ou la navigation.
        event.preventDefault();

        // Elle empêche la propagation de l'événement dans la phase de bouillonnement (bubbling)
        // Cela signifie que l'événement ne sera pas transmis aux
        // éléments parents dans le DOM.
        event.stopPropagation();

      }

      form.classList.add('was-validated');

    }, false);
  });
})();
```

- Ce script empêche la soumission si le formulaire n'est pas valide et ajoute la classe `.was-validated` pour activer les styles visuels Bootstrap.

Classes et messages de feedback

- `.is-valid` et `.is-invalid` : peuvent être appliquées manuellement pour la validation côté serveur.

- `.valid-feedback` : message affiché si le champ est valide.
- `.invalid-feedback` : message affiché si le champ est invalide.

Bonnes pratiques

- Utilisez `novalidate` pour personnaliser les messages et styles.
- Placez les messages de feedback juste après le champ à valider.
- Pour réinitialiser l'état du formulaire (ex: après un envoi AJAX), retirez la classe `.was-validated` du `<form>`.

Résumé:

Bootstrap 5 facilite la validation des formulaires en combinant la validation HTML5, des classes CSS dédiées et un petit script JavaScript pour gérer l'affichage des retours utilisateurs.

Vous pouvez ainsi offrir une expérience utilisateur claire et professionnelle, tout en gardant le contrôle sur les messages et l'apparence du formulaire.