

Validation des données de formulaires

Ce n'est pas tout d'envoyer des données — il faut aussi s'assurer que les données mises dans un formulaire par un utilisateur sont dans un format correct pour pouvoir être traitées correctement et qu'elles ne vont pas casser nos applications

Qu'est-ce qu'une validation de formulaire?

- **obtenir de bonnes données dans un bon format** — les applications ne tourneront pas correctement si les données utilisateur sont stockées dans un format fantaisiste, ou si les bonnes informations ne sont pas aux bons endroits ou totalement omises.
- **protéger nos utilisateurs** — s'ils entrent un mot de passe facile à deviner ou aucun, des utilisateurs malveillants peuvent aisément accéder à leurs comptes et voler leurs données.
- **nous protéger nous-mêmes** — il existe de nombreuses façons dont les utilisateurs malveillants peuvent utiliser les formulaires non protégés pour endommager l'application dans laquelle ils se trouvent (voir [Sécurité du site Web](#))

Les divers types de validation de formulaire

Vous rencontrerez différents types de validation de formulaires sur le Web :

- La validation côté client est la validation qui est effectuée dans le navigateur, avant que les données n'aient été soumises au serveur. Cette méthode est plus conviviale que la validation côté serveur car elle donne une réponse instantanée. Il est possible de la subdiviser encore avec :
 - la validation JavaScript, codée en JavaScript, entièrement personnalisable.
 - la validation de formulaire intégrée avec les fonctions de validation de formulaire HTML5. Elle ne nécessite généralement pas de JavaScript,

a de meilleures performances, mais elle n'est pas aussi personnalisable.

- La validation côté serveur est la validation opérée sur le serveur, après que les données ont été soumises — le code côté serveur est utilisé pour valider les données avant de les mettre dans la base de données. Si elles sont erronées, une réponse est envoyée au client pour dire à l'utilisateur ce qui a mal tourné. La validation côté serveur n'est pas aussi conviviale que la validation côté client, car elle nécessite un aller-retour vers le serveur, mais c'est la dernière ligne de défense de votre application contre les mauvaises données (c'est-à-dire les données incorrectes, voire malveillantes). Tous les modèles de canevas de vérification courants côté serveur ont des fonctions de validation et de nettoyage des données (ce qui les sécurise).

Dans le monde réel, les développeurs ont tendance à utiliser une combinaison de validations côté client et côté serveur, pour être du côté sûr.