

Stratégie pour Ajouter/Modifier/Supprimer un élément depuis le front-end

Pour ajouter, modifier ou supprimer un élément depuis le front-end, on peut utiliser une requête HTTP de type **POST**, **PUT** ou **DELETE** pour envoyer les modifications au serveur.

Créer un élément

Pour créer un élément, on peut utiliser une requête HTTP de type **POST** pour envoyer les nouvelles valeurs au serveur. La requête doit contenir les valeurs à ajouter.

La stratégie de création est la suivante :

- On affiche un formulaire pour ajouter un élément.
 - On prévalide les champs du formulaire avec les attributs **required**, **pattern**, **minLength**, **maxLength**, **min**, **max** et **type** des balises **input**.
- On écoute les changements dans le formulaire et on met à jour les valeurs dans le state.
- Récupérer le jeton d'authentification dans le local storage et l'ajouter à la section **authorization** du header de la requête.
- On envoie les nouvelles valeurs au serveur avec une requête asynchrone **POST** au format json.
- On reçoit la réponse du serveur et on met à jour l'interface au besoin.
- On affiche un message de succès ou d'erreur.
- On vide le formulaire en réinitialisant les valeurs dans le state.

Modifier un élément

Pour modifier un élément, on peut utiliser une requête HTTP de type **PUT** pour envoyer les modifications au serveur. La requête doit contenir l'identifiant de l'élément à modifier et les nouvelles valeurs.

La stratégie de modification est la suivante :

- On récupère les infos de l'élément à modifier par son id au chargement de la page. Ex: `fetch(http://localhost:3000/api/films/${id})` et on remplace les valeurs dans le state.
- On affiche un formulaire pour modifier l'élément avec les valeurs actuelles.
 - On prévalide les champs du formulaire avec les attributs **required**, **pattern**, **minLength**, **maxLength**, **min**, **max** et **type** des balises **input**.
- On écoute les changements dans le formulaire et on met à jour les valeurs dans le state.
- Récupérer le jeton d'authentification dans le local storage et l'ajouter à la section **authorization** du header de la requête.
- On envoie les nouvelles valeurs au serveur avec une requête **PUT** au format Json.
- On reçoit la réponse du serveur et on met à jour l'élément dans le state.
- On affiche un message de succès ou d'erreur.

Supprimer un élément

Pour supprimer un élément, on peut utiliser une requête HTTP de type **DELETE** pour envoyer l'identifiant de l'élément à supprimer au serveur. Généralement, on clique sur un bouton pour confirmer la suppression.

La stratégie de suppression est la suivante :

- On affiche un bouton pour supprimer l'élément.
- On récupère le id de la ressource à supprimer.
- On écoute le clic sur le bouton.
- Récupérer le jeton d'authentification dans le local storage et l'ajouter à la section **authorization** du header de la requête.
- On envoie l'identifiant de l'élément à supprimer au serveur avec une requête **DELETE**. Idéalement, on confirme avec une boîte de dialogue.
- On reçoit la réponse du serveur et on met à jour l'interface.
- On affiche un message de succès ou d'erreur.
- On retire l'élément supprimé de la liste des éléments ou on redirige l'utilisateur vers une autre page.