

Fiche d'investigation de fonctionnalité

Fonctionnalité : Affichage recettes filtrées via un champ	Fonctionnalité #1
Problématique : Optimiser la rapidité à laquelle les recettes sont affichées, en fonction de la valeur du champ de recherche principal.	

Option 1 : Utilisation de la méthode .filter() Dans cette option, nous utilisons la méthode de tableaux .filter() de JavaScript pour filtrer les recettes. Cette méthode est simple à utiliser et maintient le code concis et lisible.	
Avantages <ul style="list-style-type: none">⊕ Syntaxe concise⊕ Intégré aux prototypes de tableaux en JavaScript	Inconvénients <ul style="list-style-type: none">⊖ Performance potentiellement moins optimale pour de très grands tableaux (pas notre cas)

⊖ Moins de contrôle sur l'itération

Option 2 : Utilisation de la boucle native "for" Dans cette option, nous utilisons une boucle for native pour parcourir et filtrer les recettes. Cette méthode peut offrir une performance légèrement meilleure, surtout pour de très grands tableaux, en raison du contrôle granulaire sur l'itération	
Avantages <ul style="list-style-type: none">⊕ Contrôle total sur le processus d'itération (possibilité de commencer, stopper à l'index voulu)	Inconvénients <ul style="list-style-type: none">⊖ Code plus verbeux et potentiellement moins lisible⊖ Peut être plus difficile à maintenir

Solution retenue : Nous avons retenu l'utilisation de la méthode .filter(), car pour la plupart des cas d'utilisation, cette méthode offre un bon équilibre entre simplicité de code et performance. Cependant, il faudrait utiliser la boucle "for" si notre tableau de recettes était plus conséquent.
--