

Fiche d'investigation de fonctionnalité

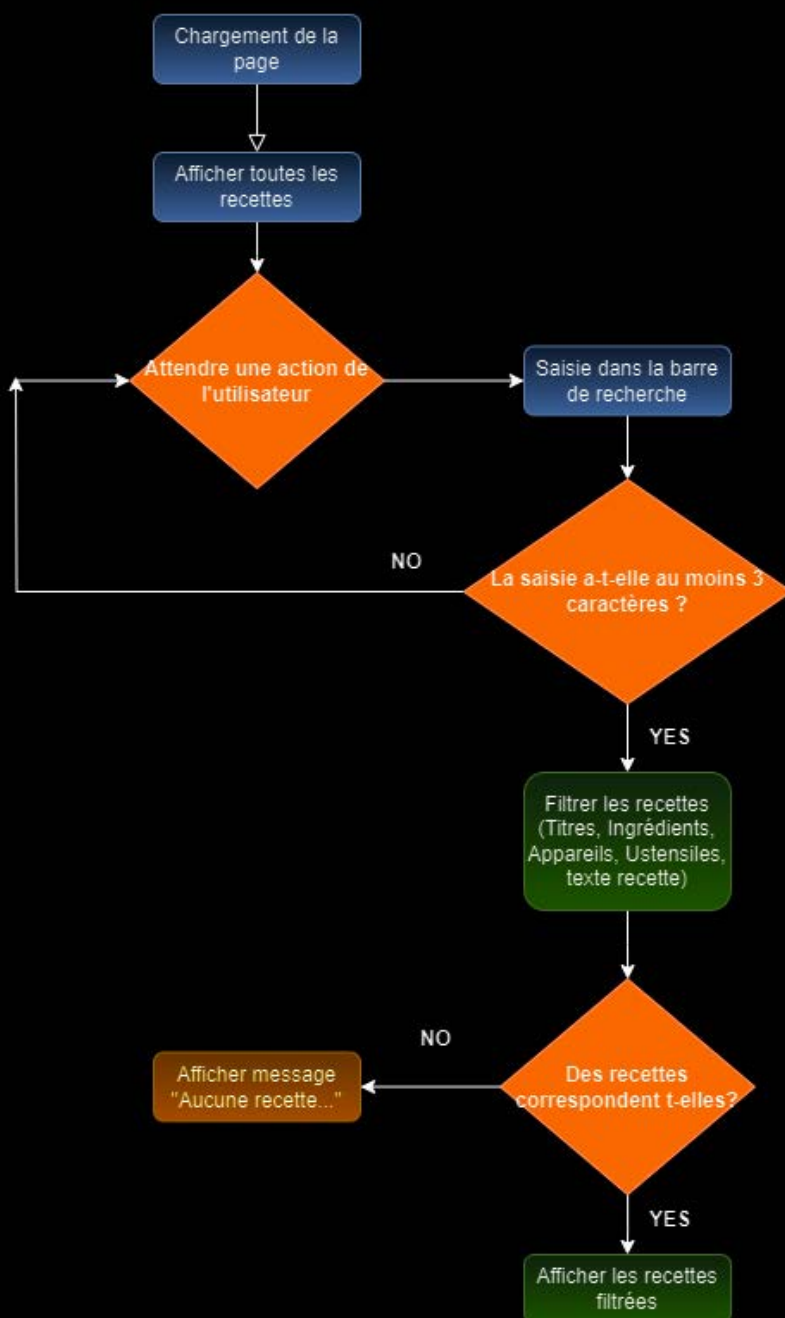
Fonctionnalité : Recherche principale	Fonctionnalité #1
Problématique : Nous cherchons à implémenter une fonction de recherche principale efficace et performante pour filtrer les recettes en fonction des entrées de l'utilisateur. L'objectif est d'optimiser la performance et la lisibilité du code.	

Option 1 : Implémentation avec boucle FOR (branche main) Cette option utilise des boucles `for` traditionnelles pour parcourir et filtrer les recettes.	
Avantages <ul style="list-style-type: none">⊕ Performance potentiellement meilleure pour de très grands ensembles de données⊕ Contrôle précis sur l'itération⊕ Possibilité d'interrompre la boucle prématurément si nécessaire	Inconvénients <ul style="list-style-type: none">⊖ Code plus verbeux et potentiellement moins lisible⊖ Plus susceptible aux erreurs de logique (par exemple, erreurs d'indexation)⊖ Moins expressif en termes d'intention du code
Nombre de caractère minimum dans la barre de recherche : 3	

Option 2 : Implémentation avec méthodes d'array (filter) (branche feature/functional-search) Cette option utilise des méthodes d'array fonctionnelles comme `filter` pour parcourir et filtrer les recettes.	
Avantages <ul style="list-style-type: none">⊕ Code plus concis et plus lisible⊕ Approche déclarative, exprimant clairement l'intention du code⊕ Réduction des risques d'erreurs liées à l'indexation⊕ Facilité de maintenance et de modification	Inconvénients <ul style="list-style-type: none">⊖ Peut être légèrement moins performant pour de très grands ensembles de données⊖ Potentiellement moins flexible pour des logiques de filtrage très complexes⊖ Courbe d'apprentissage pour les développeurs moins familiers avec la programmation fonctionnelle
Nombre de caractère minimum dans la barre de recherche : 3	

Solution retenue : Option 2 (méthodes d'array)
Justification : <ol style="list-style-type: none">1. Code plus lisible et maintenable (Code plus concis et expressif, facilite la compréhension et maintenance à long terme).2. Réduction des erreurs d'indexation3. Performance adaptée à l'application4. Alignement avec les pratiques JavaScript modernes5. Facilité d'extension et de modification future

Annexes



Algorithme fonctionnalités barre de recherche principale