

Fiche d'investigation de fonctionnalité

Fonctionnalité : Rechercher une recette	Fonctionnalité #1
Problématique : Les performances en vitesse de réponse de l'algorithme de recherche	

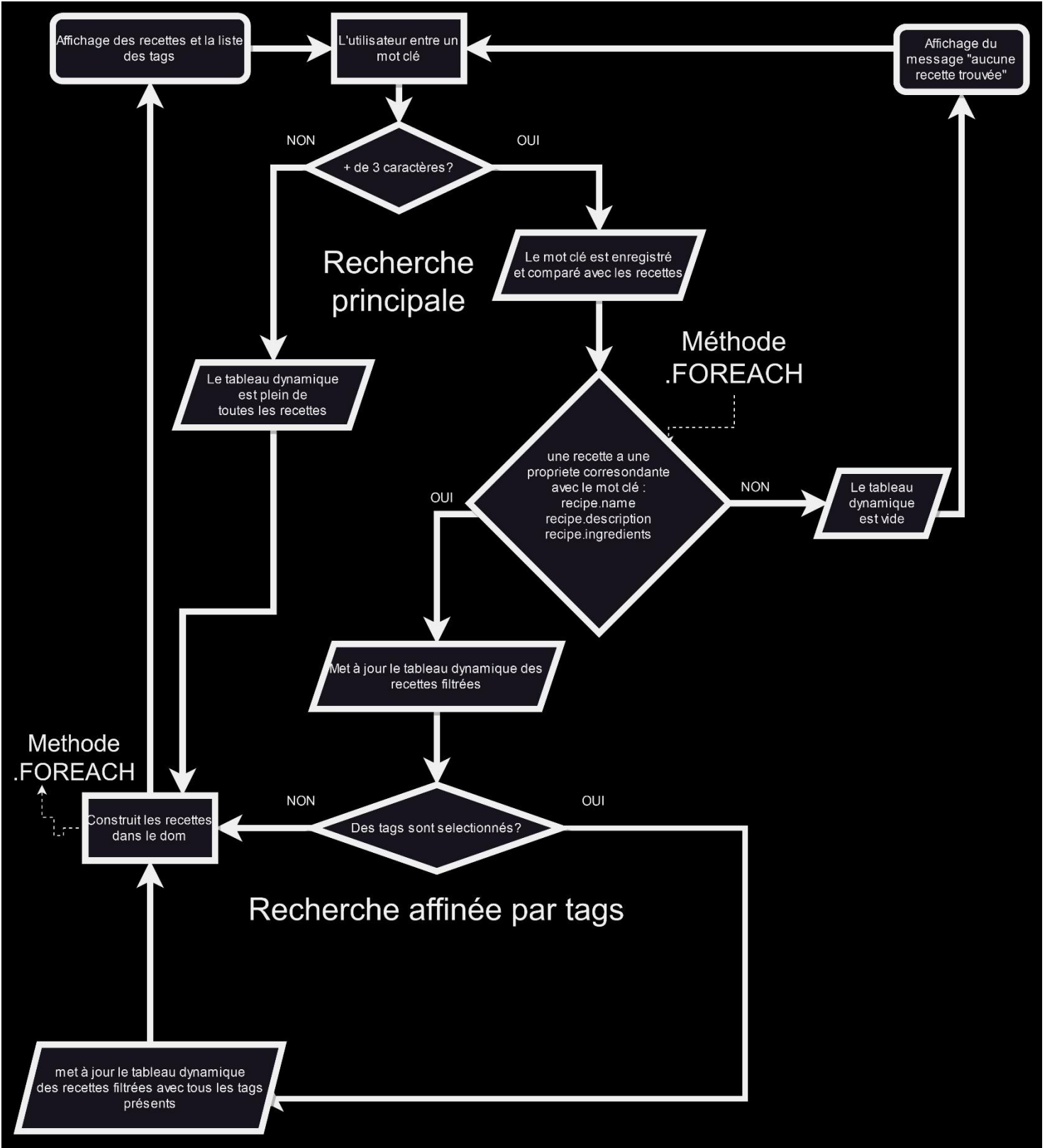
Option 1 : Les méthodes de l'objet Array de Javascript La méthode .forEach est utilisé pour les boucles parmi les recettes. Pour forEach, 3 callbacks sont retournés à chaque élément parcouru. Ce qui a tendance à ralentir le travail de l'algorithme à mesure que les données s'enrichissent. Sortir de la boucle avant d'avoir parcouru tous les éléments du tableau n'est pas aussi évident que pour la boucle For.	
Avantages <ul style="list-style-type: none">⊕ Lisibilité⊕ Ecriture rapide et agréable du code⊕ réponse rapide sur des données légères	Inconvénients <ul style="list-style-type: none">⊖ au vue de données conséquentes, elle est moins performante que l'usage de FOR
Nombre d'opération par seconde visible sur jsben.ch https://jsben.ch/L84hx	

Option 2 : Usage de la boucle FOR native de Javascript Dans la version 2, la boucle FOR est utilisée pour parcourir les recettes et ajoute les recettes répondant positives aux conditions fixées par les mots clés dans un nouveau tableau. Les conditions des returns et leur nature sont définie selon les situations attendues et permet une sortie de boucle à tout moment.	
Avantages <ul style="list-style-type: none">⊕ Plus rapide	Inconvénients <ul style="list-style-type: none">⊖ écriture plus lourde que les précédentes méthodes
Nombre d'opération par seconde visible sur jsben.ch : https://jsben.ch/L84hx	

Solution retenue : Sur des données légères, les méthodes des objets Array sont beaucoup plus pertinentes a utilisé que la classique boucle For. Néanmoins, pour ce projet, nous devons anticiper les besoins en performance qu'aura le site lors du traitement de données de plus grande envergure. Pour cela, la boucle For y répond avec beaucoup plus de force.
--

Annexes

Algorithme : .FOREACH



Algorithme : BOUCLE FOR

