

Fiche d'investigations de fonctionnalité

Fonctionnalité : Recherche principal
Problématique : on a besoin d'une recherche performante pour que nos utilisateur fassent une recherche rapide presque instantanée.

Option 1 : utilisation de filter et some	
Avantages: <ul style="list-style-type: none">- code plus court- code plus claire- moins de code répétitif	Inconvénients <ul style="list-style-type: none">- génère un nouveau tableau temporaire- appelle une fonction de rappel à chaque itération
Utiliser Filter quand il n'y a pas beaucoup de données à gérer	D'après l'outils de comparaison de performance Jsben Test 1: 139 789 execution 99% performance Test 2: 140 671 execution 95.6% performance Test 3: 139 830 execution 96.24% performance

Option 2 : utilisation de for	
Avantages: <ul style="list-style-type: none">- pas de creation de tableau temporaire- pas d'appelle de fonction de rappel à chaque itération	Inconvénients <ul style="list-style-type: none">- code plus long- code moins visible- plus de code répétitif
Utiliser For quand il y a beaucoup de données à gérer	D'après l'outils de comparaison de performance Jsben Test 1: 141 202 execution 100% performance Test 2: 147 032 execution 100% performance Test 3: 145 288 execution 100% performance

Solution retenue :

Nous avons donc choisi d'adopter l'approche utilisant For, car à terme, il y aura un volume de données beaucoup plus important à traiter.

Annexes

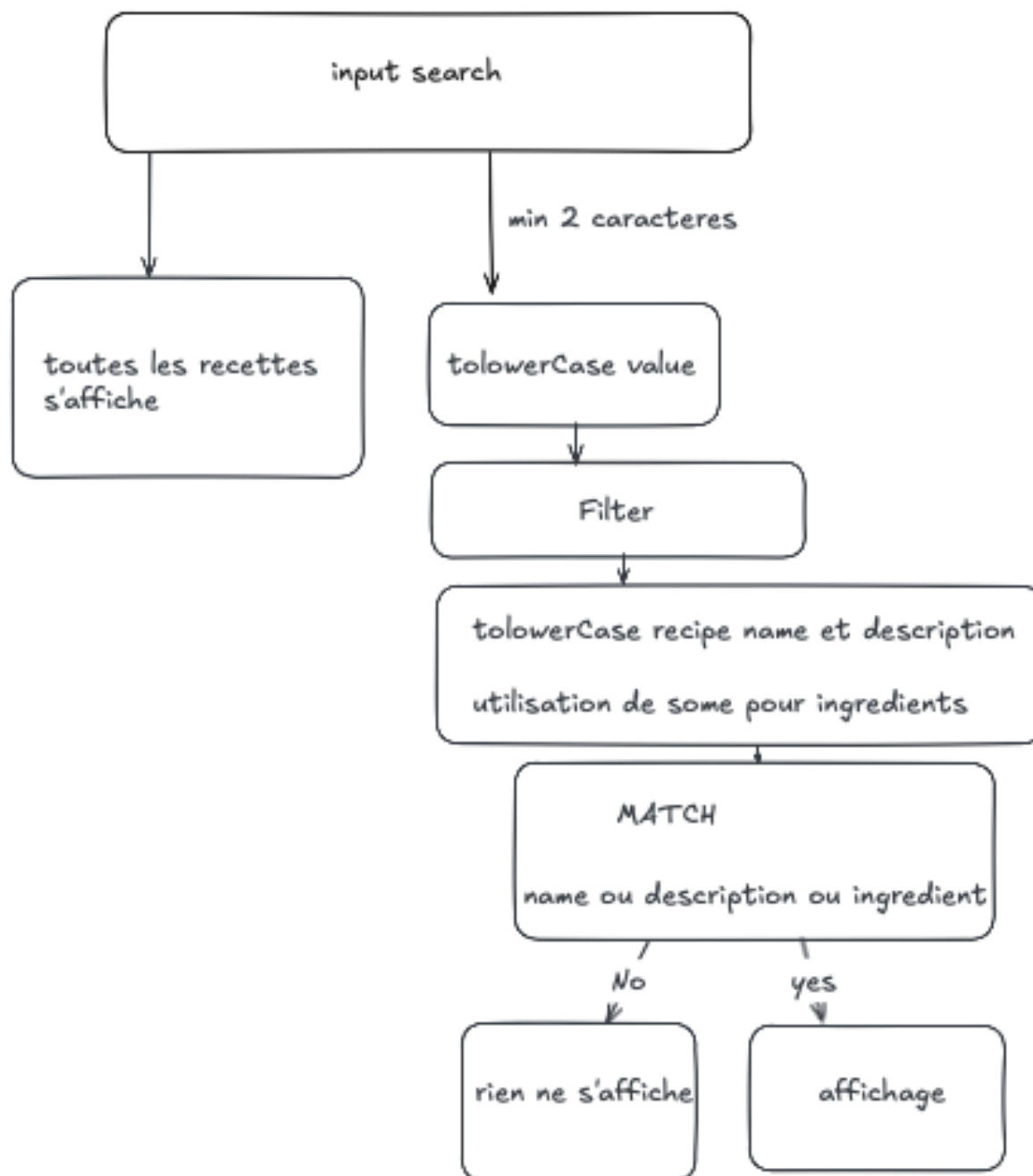


Figure 1 - schéma recherche avec filter et some

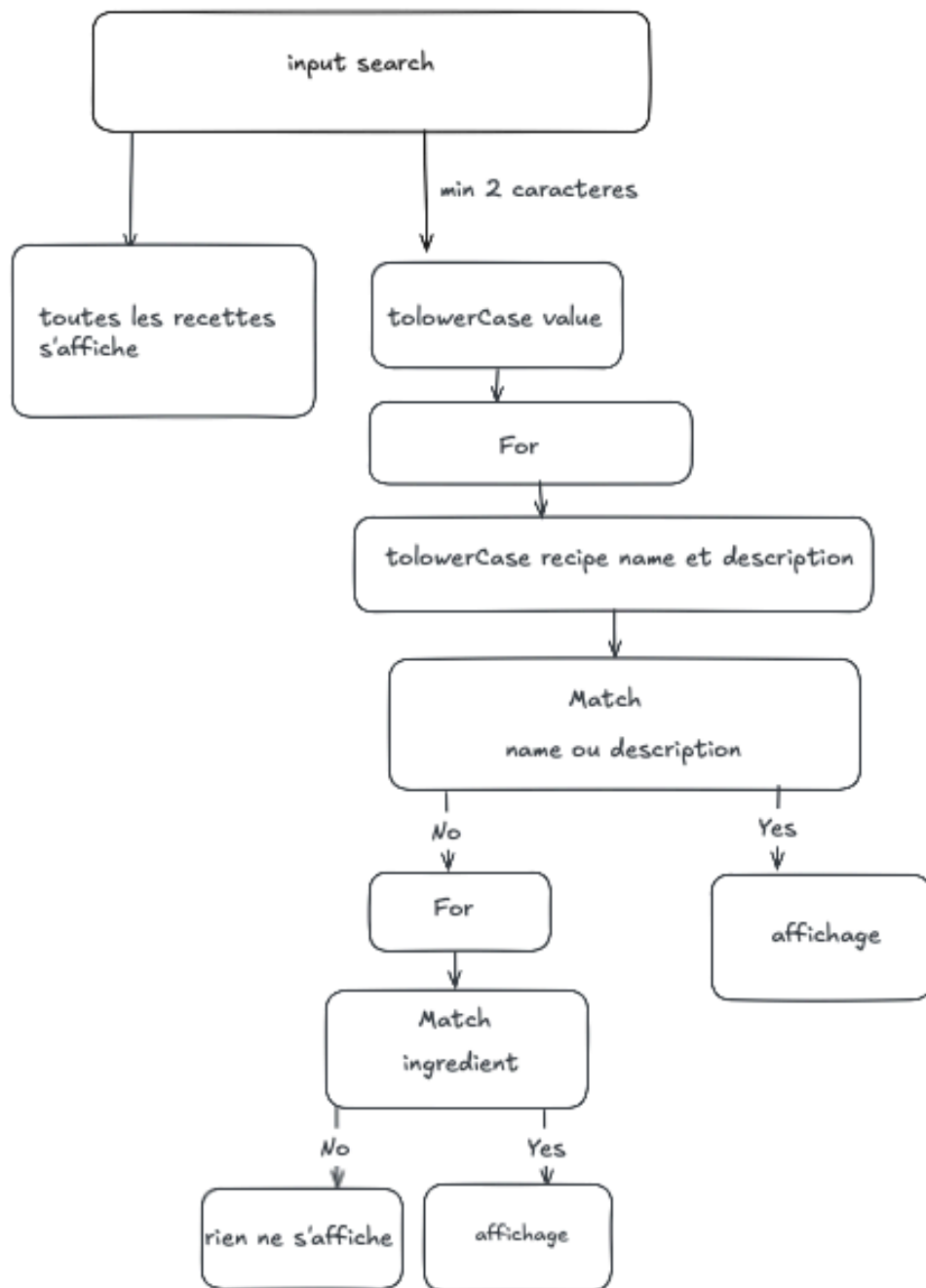


Figure 2 - schéma recherche avec for