



## Les petits plats

### Fiche d'investigation de fonctionnalité#2

<b>Fonctionnalité:</b> recherche de recettes	<b>Fonctionnalité #2</b>
<b>Problématique:</b> Implémenter une recherche rapide presque instantanée	

#### Option 1: Les boucles natives

Une boucle `for` itère l'objet tableau "Recipes" pour vérifier si la propriété "name" ou la propriété "Description" ou la propriété "Ingrédients" de l'élément contiennent l'objet string "recherche" ; en cas de test positif, l'élément est ajouté au tableau des résultats de la recherche.

##### Avantages

- ⊕ code accepté par tous les navigateurs même anciens

##### Inconvénients

- ⊕ code moins lisible
- ⊕ code plus volumineux
- ⊕ code moins performant

##### JSBench

709289.6 ops/s +/- 0.58%

19.92% slower

#### Option 2: La méthode `filter`

`filter()` appelle la fonction callback fournie pour chacun des éléments d'un tableau, et construit un nouveau tableau contenant tous les éléments pour lesquels l'appel de callback retourne `true` ou une valeur équivalente à `true` dans un contexte booléen. La fonction callback n'est utilisée que pour les éléments du tableau ayant une valeur assignée — les index supprimés ou pour lesquels il n'y a jamais eu de valeur ne sont pas pris en compte. Les éléments du tableau qui ne passent pas le test effectué par la fonction callback sont ignorés, ils ne sont pas inclus dans le nouveau tableau.

##### Avantages

- ⊕ code plus lisible
- ⊕ code plus compact
- ⊕ code plus performant

##### Inconvénients

- ⊕ algorithme optimisé
- ⊕ besoin de polyfill pour les navigateurs anciens ne supportant pas ES5

##### JSBench

885744.21 ops/s +/- 0.7%

fastest



### Solution retenue :

Nous avons donc retenu l'**option 2**, la méthode ES5 filter(), car plus performante.

**La rapidité est un prérequis du client.**

### Annexes

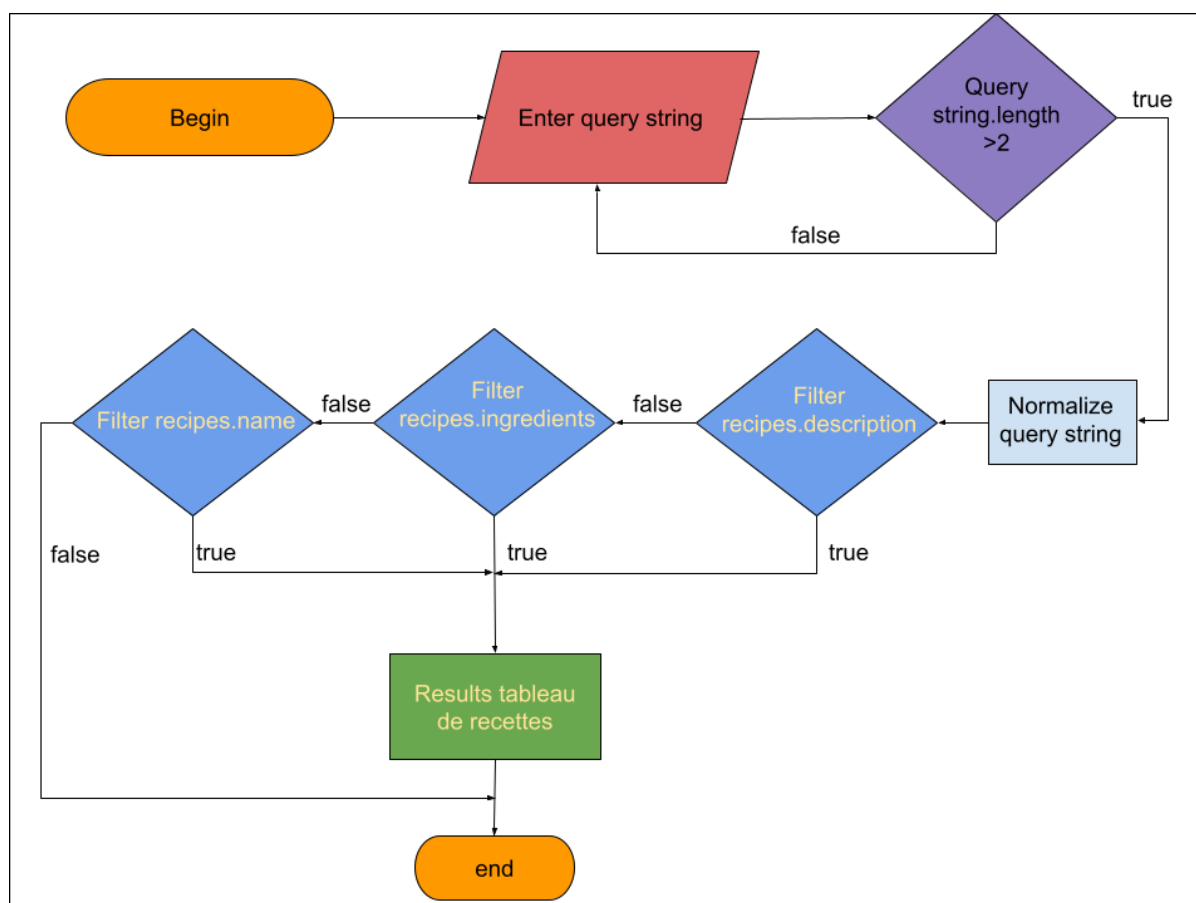


Figure 1 : logigramme de la version de recherche par la méthode ES5 filter()

[OBJ]



## Les petits plats fonctionnalité#2

[OBJ] Fiche d'investigation de

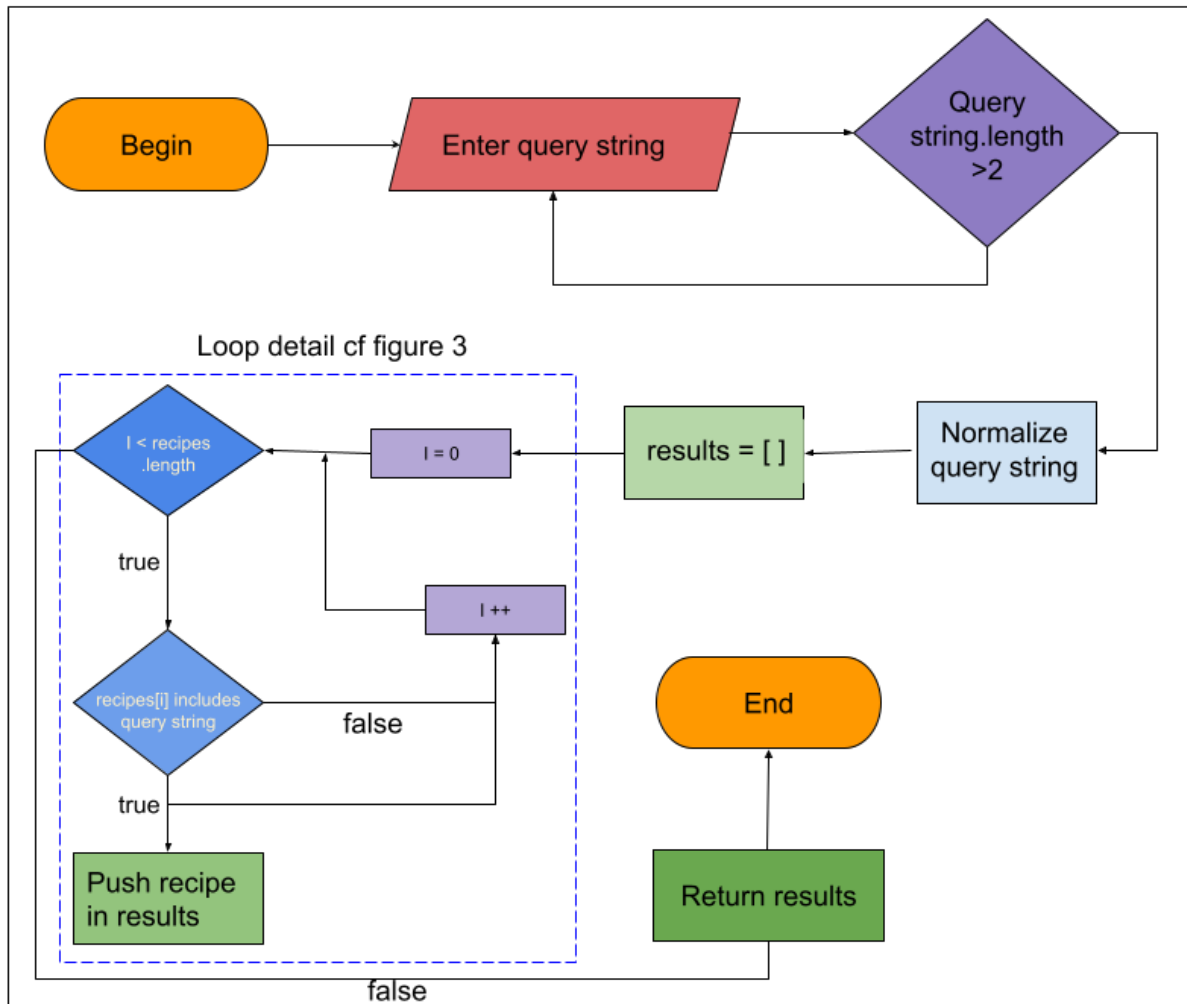


Figure 2 : logigramme de la version de recherche par boucle native for()

[OBJ]

[OBJ]



## Les petits plats fonctionnalité#2

[OBJ] Fiche d'investigation de

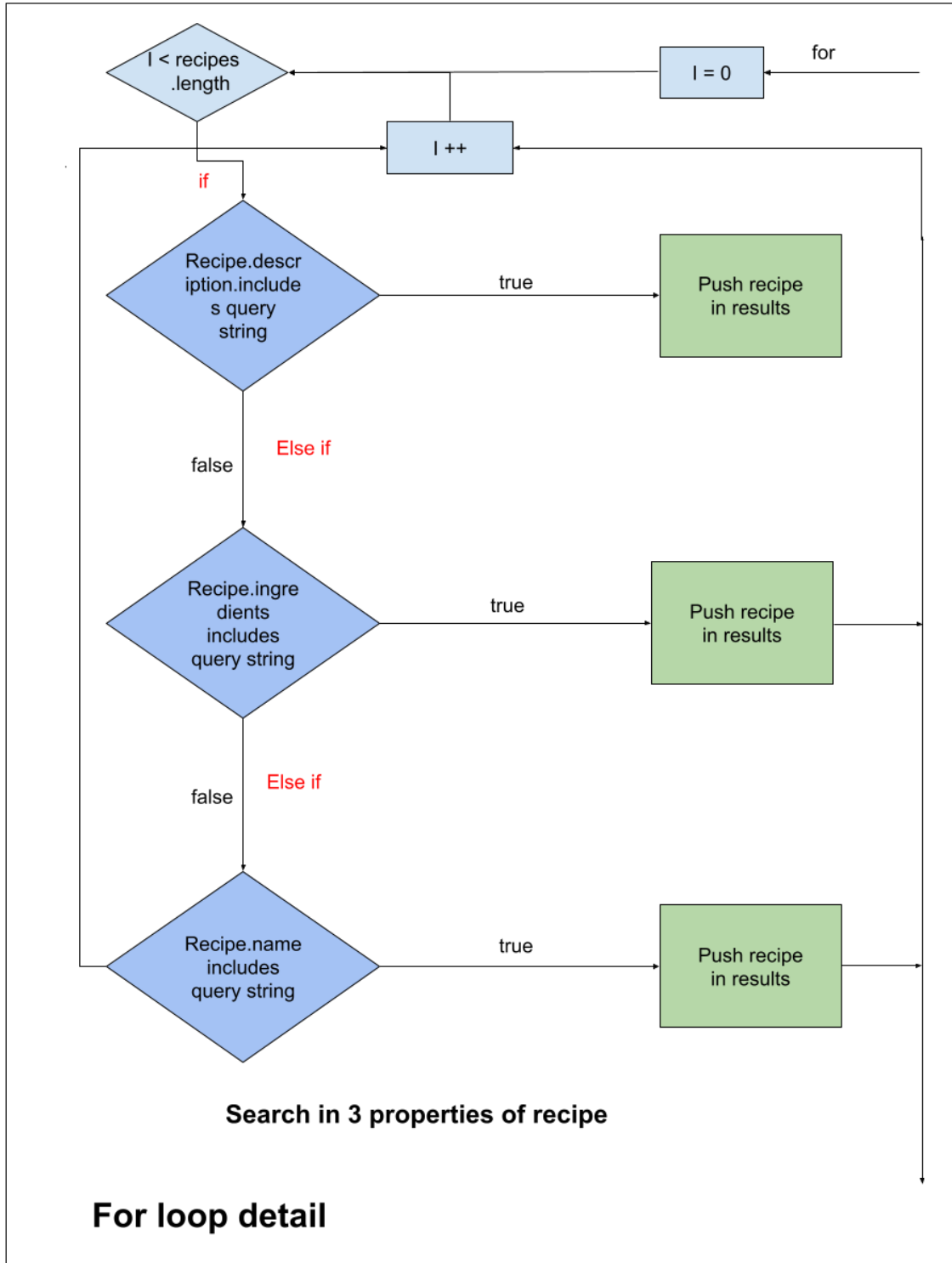


figure 3 : logigramme de l'enchaînement 2 "OU logique" ou de if avec 2 else if