

## Fiche d'investigation de fonctionnalité

<b>Fonctionnalité :</b> Filtre des recettes	<b>Fonctionnalité #1</b>
<b>Problématique :</b> Les sites offrant des recettes de cuisine étant nombreux, l'un des éléments qui peuvent faire la différence sur notre site est la fluidité du moteur de recherche.	

### Option 1 : `.forEach()`

**Définition de MDN :** La méthode `.forEach()` permet d'exécuter une fonction donnée sur chaque élément du tableau.

Ex :

```
let newArrays = [];  
arrays.forEach((array) => { if (array.includes(value)) { newArrays.push(array) } })
```

Dans cette option, on vérifie si la valeur recherchée est incluse dans le nom, l'appareil, la description, les ustensiles et les ingrédients de chaque recette. Si la méthode `.includes()` renvoie « true » alors la recette est ajoutée à un nouveau tableau à initialiser au préalable.

#### Avantages :

- Possibilité d'imbriquer d'autres boucles pour les sous-tableau
- Possibilité de regrouper les actions

#### Inconvénients :

- Nouveau tableau à initialiser au préalable (variable modifiable : source d'erreur)
- Rédaction longue (90 caractères pour l'ex)

### Option 2 : `.filter()`

**Définition MDN :** La méthode `.filter()` crée et retourne un nouveau tableau contenant tous les éléments du tableau d'origine qui remplissent une condition déterminée par la fonction callback.

Ex : `const newArrays = arrays.filter(array => array.includes(value));`

Dans cette option, on vérifie si la valeur recherchée est incluse dans le nom, l'appareil et la description de chaque recette (éléments de 1er niveau). Si la méthode `.includes()` renvoie « true » alors la recette est poussée dans un nouveau tableau créé par la méthode.

#### Avantages :

- Facilité de rédaction du code (58 caractères)
- Meilleure performance benchmark
- Le nouveau tableau est une constante (code robuste)

#### Inconvénients :

- Nécessité de maintenir une boucle for sur les éléments de sous tableau (ustensils et ingrédients)
- Résultat n'est qu'un tableau (pas de regroupement d'action)
- Attention à l'emplacement de `.filter()` dans le code

### Solution retenue :

Compte tenu des performances étudiées sur jsben.ch l'option 2 - `.filter()` est plus adaptée car plus rapide. Dans le cas d'un plus grand nombre de recettes, cet argument pèsera sur l'expérience utilisateur.



