

# Fiche d'investigation de fonctionnalité

Fonctionnalité : research engine	Fonctionnalité #2

Option 1 : Boucles natives	
<b>Avantage :</b>  ⊕ Simplicité de l'architecture	<b>Inconvénients :</b>  ⊗ Lisibilité diminuée ⊗ Répétition parfois excessive ⊗ Maintenance + complexe
Nombre minimum de caractères à remplir : 3 caractères Rechercher le titre de la recette, la liste des ingrédients, la description de la recette.	

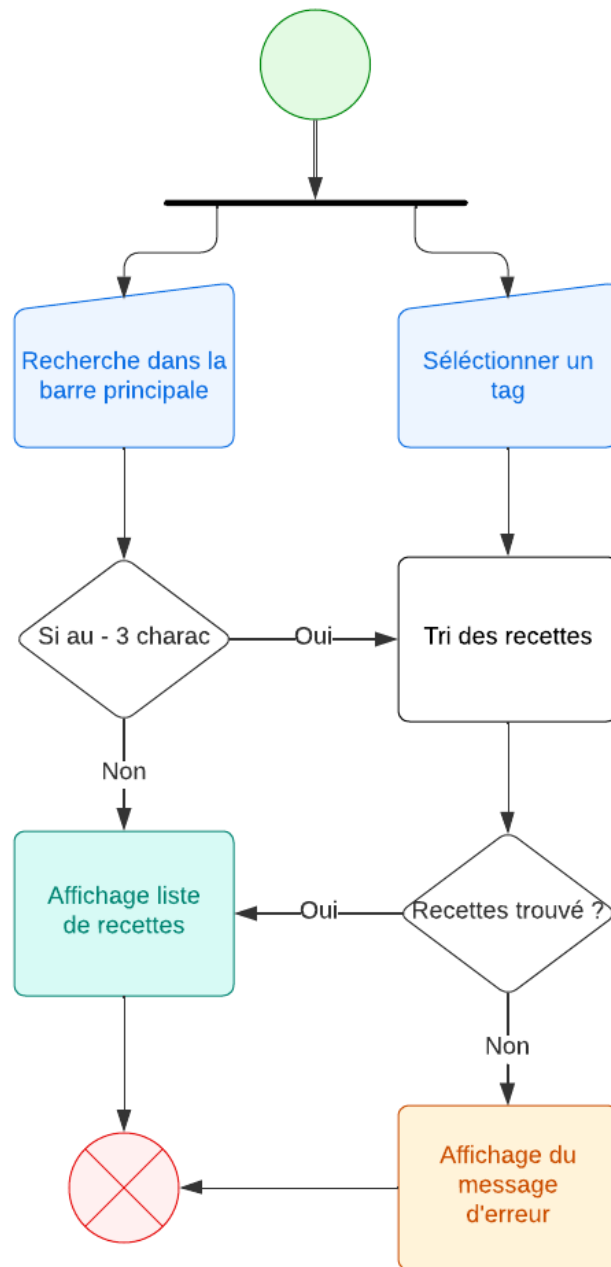
Option 2 : Programmation fonctionnel (méthode de l'objet <i>array</i> )	
<b>Avantage :</b>  ⊕ Code plus expressif ⊕ Lisibilité ⊕ Maintenable ⊕ Réutilisabilité	<b>Inconvénients :</b>  ⊗ RAS
Nombre minimum de caractères à remplir : 3 caractères Rechercher le titre de la recette, la liste des ingrédients, la description de la recette.	

<b>Solution retenue :</b> Le JS Benchmark notifie 510 ops/s quant à l'algorithmie en programmation fonctionnel, et 259 ops/s quant aux boucles natives. La solution de programmation fonctionnel étant plus rapide (~ 96% + rapide) nous retenons cette dernière.
--

# Annexes

## PRJ 7 OC

LAUER Jonathan | September 19, 2023



# Annexes

JSBench.Me

jonathan

Sponsor

Settings

Run

Save

Setup JavaScript

```
{
  "id": 50,
  "image": "recette50.jpg",
  "name": "Fragipane",
  "servings": 2,
  "ingredients": [
    {
      "ingredient": " pâte feuilletée",
      "quantity": 400,
      "unit": "grammes"
    }
  ]
}
```

array v

finished

510 ops/s ± 2.93%

Fastest

natif v

finished

259 ops/s ± 13.78%

49.14 % slower

```
/* ***** Import utils class & functions ***** */
class DisplayData {
  constructor(recipes) {
    this.recipes = recipes;
    this.recipesContainer = document.querySelector("#recipesContainer");
    this.nbrRecipesContainer = document.querySelector("#nbrRecipes");
  }
  cardTemplate() {
    this.recipesContainer.innerHTML = "";
    let nbrRecipesContent = "";
  }
}
/* ***** Import utils class & functions ***** */
class DisplayData {
  constructor(recipes) {
    this.recipes = recipes;
    this.recipesContainer = document.querySelector("#recipesContainer");
    this.nbrRecipesContainer = document.querySelector("#nbrRecipes");
  }
  cardTemplate() {
    this.recipesContainer.innerHTML = "";
    let nbrRecipesContent = "";
    this.recipes.length > 1
  }
}
```

DEFER

DEFER