

Fonctionnalité

Search

Problématique

Afin d'offrir aux utilisateurs une expérience de recherche optimale, nous souhaitons identifier la meilleure approche pour implémenter la recherche par saisie utilisateur et la recherche par filtres dans les menus déroulants.

Option 1: Boucle "For"

Cette option consiste à utiliser une boucle for pour parcourir la liste de recettes et identifier celles qui correspondent aux critères de recherche de l'utilisateur.

- Approche simple et facile à comprendre
- Bonne performance pour la recherche par saisie utilisateur
- Moins performant pour la recherche par filtres dans les menus déroulants

Option 2: Méthode "Filter"

Cette option consiste à utiliser la méthode filter() de JavaScript pour filtrer la liste de recettes en fonction des critères de recherche de l'utilisateur.

- Bonne performance pour la recherche par filtres dans les menus déroulants
- Plus concis et élégant que l'utilisation d'une boucle for
- Moins performant pour la recherche par saisie utilisateur

Solution retenue

Sur la base des résultats des tests de performance, nous avons décidé d'utiliser les approches suivantes pour la recherche par saisie utilisateur et la recherche par filtres dans les menus déroulants :

- Recherche par saisie utilisateur: Boucle "For"
- Recherche par filtres dans les menus déroulants: Méthode "Filter"

Ce choix nous permet de bénéficier des meilleures performances pour les deux types de recherche.

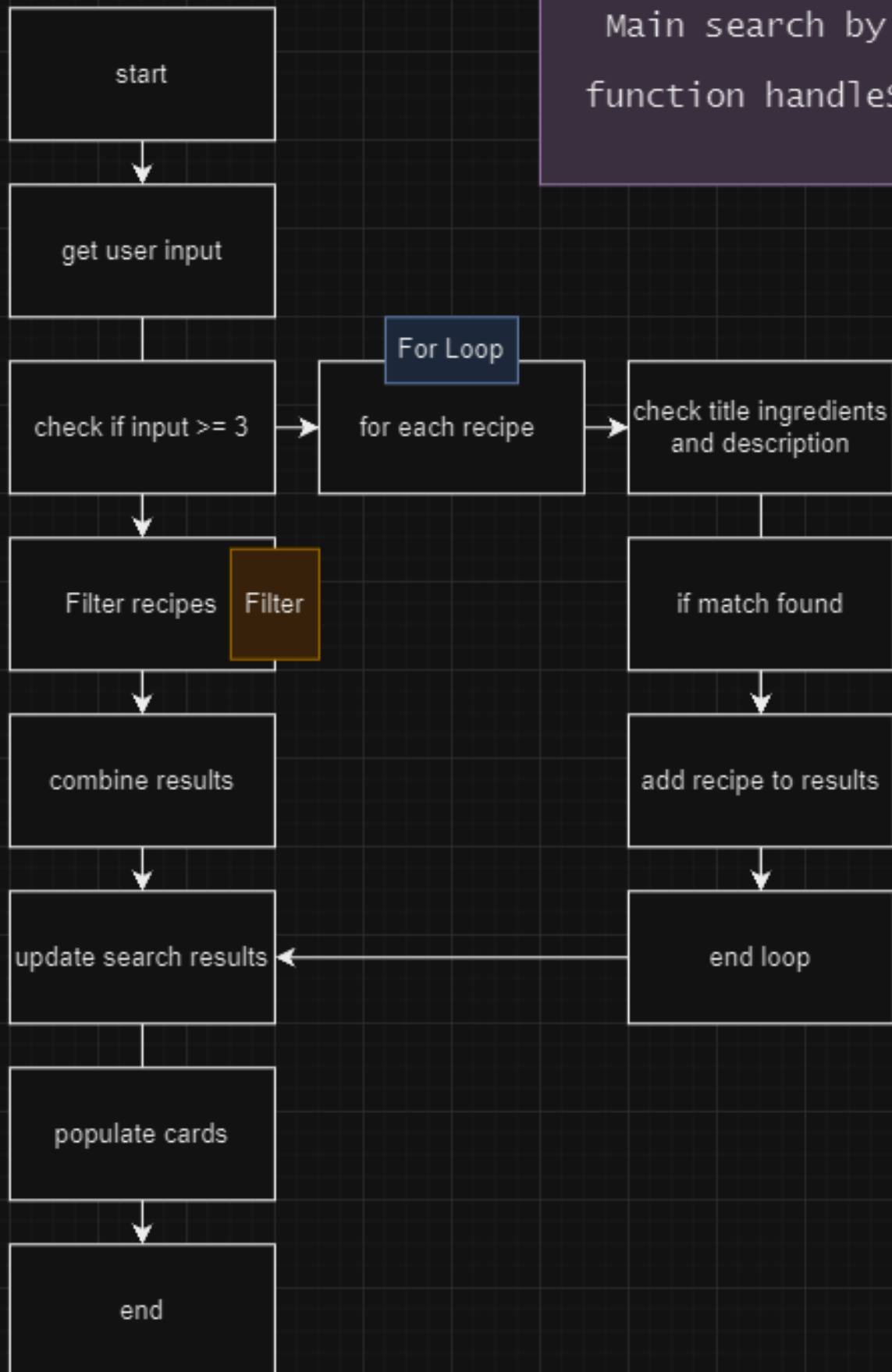
Recommandations

- Surveiller régulièrement les performances de la recherche afin de s'assurer qu'elles restent satisfaisantes.
- Envisager l'utilisation d'une bibliothèque de recherche spécialisée si les performances de la recherche deviennent insuffisantes.

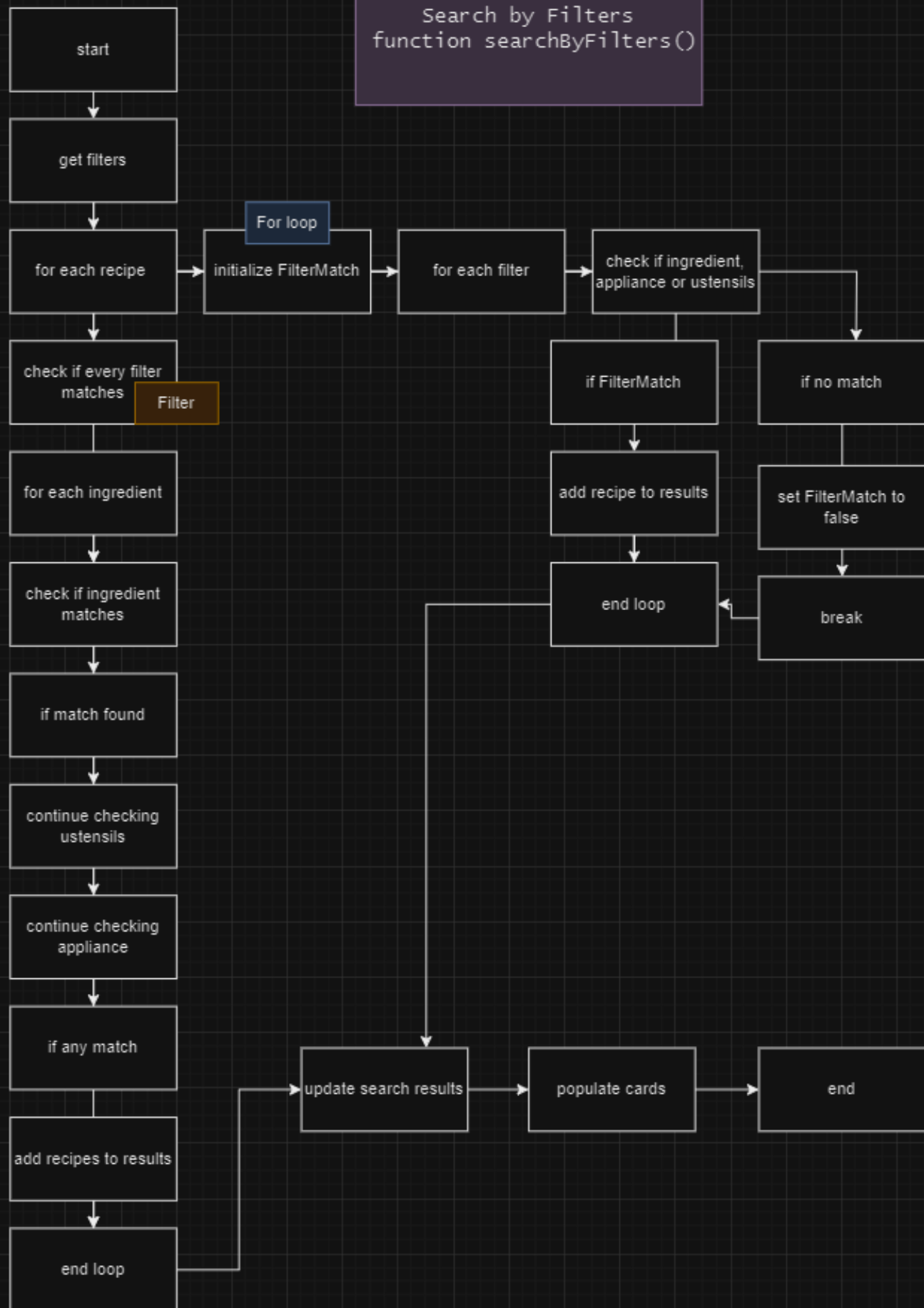
Conclusion

Nous avons identifié les meilleures approches pour implémenter la recherche par saisie utilisateur et la recherche par filtres dans les menus déroulants. Ces approches nous permettent de bénéficier des meilleures performances pour les deux types de recherche.

Main search by input
function handleSearch()



Search by Filters function searchByFilters()



Setup block (useful for function initialization. it will be run before every test, and is not part of the benchmark.)

boilerplate block (code will executed before every block and is part of the benchmark. use it for data initializing.)

```
1 const recipes = [
2   {
3     "id": 1,
4     "image": "Recette01.jpg",
5     "name" : "Limonade de Coco",
6     "servings" : 1,
7     "ingredients": [
8       {
9         "ingredient" : "Lait de coco",
10        "quantity" : 400,
11        "unit" : "ml"
12      }
13    ]
14  }
15 ]
```

code block 1

```
1 function handleSearch() {
2   const userInput = searchInput.value.toLowerCase();
3   if (userInput.length >= 3) {
4     selectedFilters = [];
5     results = recipes.filter(recipe => {
6       const titleMatch = recipe.name.toLowerCase().includes(userInput);
7       const ingredientsMatch = recipe.ingredients.some(ingredient => ingredient.ingredient.toLowerCase().includes(userInput));
8       const descriptionMatch = recipe.description.toLowerCase().includes(userInput);
9       return titleMatch || ingredientsMatch || descriptionMatch;
10    });
11    updateSearchResults(results);
12    populateCards(results);
13  }
```

result











code block 2 (125689) 🏆



code block 1 (124258)



If you like to donate (Thank you!):

-  Ethereum (ETH)
-  Chia (XCH)
-  Cardano (ADA)
-  Ravencoin (RVN)
-  Bitcoin (BTC)
-  Ripple (XRP)
-  Litecoin (LTC)
-  Monero (XMR)
-  Dogecoin (DOGE)
-  SOLANA (SOL)

Setup block (useful for function initialization. it will be run before every test, and is not part of the benchmark.)

boilerplate block (code will executed before every block and is part of the benchmark. use it for data initializing.)

```
1 const recipes = [
2   {
3     "id": 1,
4     "image": "Recette01.jpg",
5     "name" : "Limonade de Coco",
6     "servings" : 1,
7     "ingredients": [
8       {
9         "ingredient" : "Lait de coco",
10        "quantity" : 400,
11        "unit" : "ml"
12      }
13    ]
14  }
15 ]
```

code block 1

```
1 function searchByFilters(selectedFilters) {
2   results = recipes.filter(recipe => {
3     return selectedFilters.every(filter => {
4       if (recipe.ingredients.some(ingredient => ingredient.ingredient.toLowerCase().includes(filter.ingredient.toLowerCase())) {
5         return true;
6       } else if (recipe.appliance.toLowerCase().includes(filter.appliance.toLowerCase())) {
7         return true;
8       } else if (recipe.ustensils.some(ustensil => ustensil.toLowerCase().includes(filter.ustensil.toLowerCase())) {
9         return true;
10      } else {
11        return false;
12      }
13    })
14  })
15 }
```

result


code block 1 (126622)



code block 2 (125811)



If you like to donate (Thank you!):

-  Ethereum (ETH)
-  Chia (XCH)
-  Cardano (ADA)
-  Ravencoin (RVN)
-  Bitcoin (BTC)
-  Ripple (XRP)
-  Litecoin (LTC)
-  Monero (XMR)
-  Dogecoin (DOGE)
-  SOLANA (SOL)