

Fiche d'investigation fonctionnalité #1

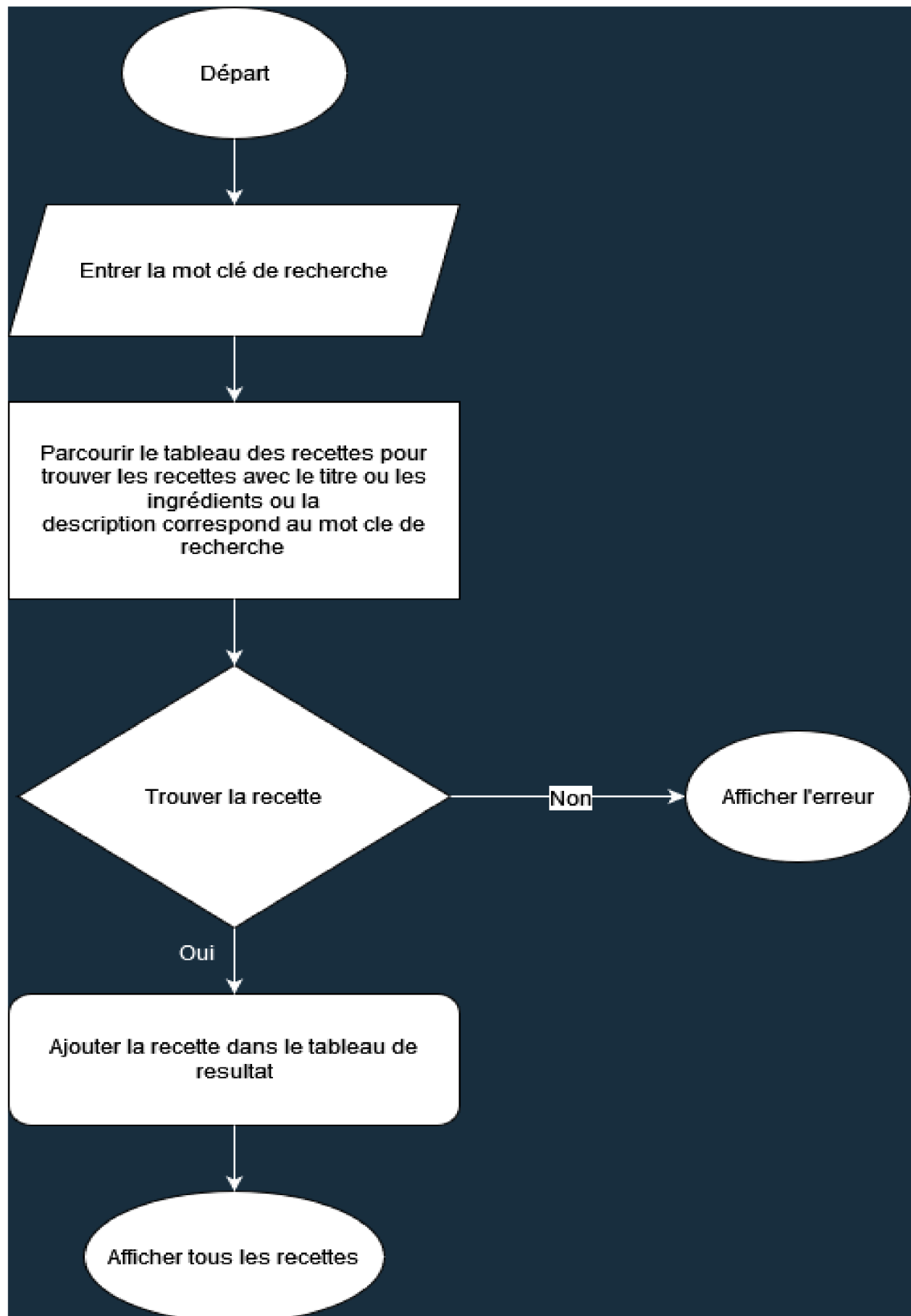
Fiche d'investigation de fonctionnalité

Fonctionnalité : Moteur de recherche	Fonctionnalité #2
Problématique : Permettre aux utilisateurs de trouver la recette souhaitée	

Option 1: Utiliser la boucle "for" native	
Avantages: La performance de la boucle native est plus important	Inconvénients: Le code est moins lisible c
La performance calculee avec jsben.ch: 100 %	

Option 2: Utiliser des méthodes array ("filter", "some", "forEach")sur les tableaux	
Avantages: La méthode de tableau facilite le travail avec eux et garde également votre code propre	Inconvénients: le mappage d'un tableau créera une copie de chaque valeur afin de ne pas modifier le tableau d'origine.
La performance calculee avec jsben.ch: 86,79%	

Solution retenue : La différence de performances est perceptible lorsque vous avez une grande quantité de données. Après avoir comparé les 2 méthodes, les méthodes de tableau sont beaucoup plus lentes que pour la boucle. Donc, l'algorithme qui utilise la boucle for native a été choisi pour l'implémentation finale.
--



no title (put title and/or keywords here, which describes your test)

RUN TESTS GENERATE PAGE URL NEW BENCHMARK

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

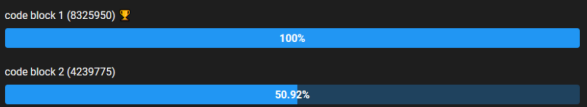
code block 1

```
1+ const searchOnRecipes = (searchingList, searchKey) => {  
2+   const filteredRecipes = [];  
3+  
4+   searchKey = searchKey.toLowerCase();  
5+  
6+   for (let i = 0; i < searchingList.length; i++) {  
7+     const recipe = searchingList[i];  
8+     if (isSearchWordExist(recipe.name, searchKey)) {  
9+       filteredRecipes.push(recipe);  
10+    } else if (isSearchWordExist(recipe.description, searchKey)) {  
11+      filteredRecipes.push(recipe);  
12+    } else {
```

code block 2

```
1+ const searchOnRecipes = (searchingList, searchKey) => {  
2+   let filteredRecipes = []  
3+   searchKey = searchKey.toLowerCase();  
4+  
5+   const res = searchingList.filter( item => isSearchWordValid(item.name, searchKey) || isSearchWordValid(item.description, searchKey) )  
6+  
7+  
8+   filteredRecipes = [...res]  
9+   return filteredRecipes;  
10+ };
```

result



If you like to donate (Thank you!):

- Ethereum (ETH)
- Chia (XCH)
- Cardano (ADA)
- Ravencoin (RVN)
- Bitcoin (BTC)
- Ripple (XRP)