Les petits plats

Fiche d'investigation fonctionnalitée #2

Fonctionnalité: Recherche Principale	Fonctionnalité #2
Problématique: Afin de proposer la meilleure expérience aux utilisateurs, nous cherchons à avoir la recherche la plus fluide et rapide possible.	

Version A: Recherche utilisant la programmation fonctionnelle et les methodes de tableaux (Ecma Script version 6, Juin 2015)
Dans cette option, nous utilisons les methodes de l'objet tabeau pré-conçus (tel que spécifié dans la norme ES6). L'avantage de cette version, c'est la réutilisation d'opérations communes sur les tableaux, stable et sans développement supplémentaires, ni maintenance à apporter.

Avantages:	**Inconvénients:**
• Syntaxe plus condensé et lisible. • Utilise les méthodes de tableaux fiables. • Moins de dette technique (dans le futur).	• La solution doit être transpilé en ES5 pour une compatibilité maximale avec les navigateurs plus anciens. • Doit être transpiler et mise en production à chaque nouvelles modifications. • Utilisation de différents outils, Npm, Webpack et Babel.

Doit respecter la description du cas d'utilisation de recherche.
L'utilisateur rentre au moins 3 charactères dans le champ de recherche.
Cette version A doit être plus rapide que la version B.

Version B: Recherche utilisant des boucles natives (Ecma Script version 5, Décembre 2009)
Dans cette option, nous développons nous-mêmes les fonctionnalitésde recherche sur les tableaux, en utilisant des boucles for {...} et des opérations de comparaisons. L'avantage de cette méhode, c'est une plus grande compatibilité avec les anciens navigateurs et moins de dépendance à installer pour le développement et la mise en production de la solution.

Avantages:	**Inconvénients:**
• Compatibilité avec les anciens navigateurs. • Moins de dépendances à installer pour développer la solution.	• Peut entraîner une dette technique. • Peu être source d'erreurs humaines (dans le futur).

Doit respecter la description du cas d'utilisation de recherche.
L'utilisateur rentre au moins 3 charactères dans le champ de recherche.
Cette version B doit être plus rapide que la version A.

Solution retenue:
Nous retiendrons la version A, en effet celle-ci est plus rapide (environ 70% selon les tests) que la version B (Page 4, Fig.3 et le résultat et lien du Benchmark comparatifs des deux solutions). Le résultat du Benchmark indique, que les methodes de tableaux ES6 sont interpretées plus rapidement par les navigateurs Web actuels.

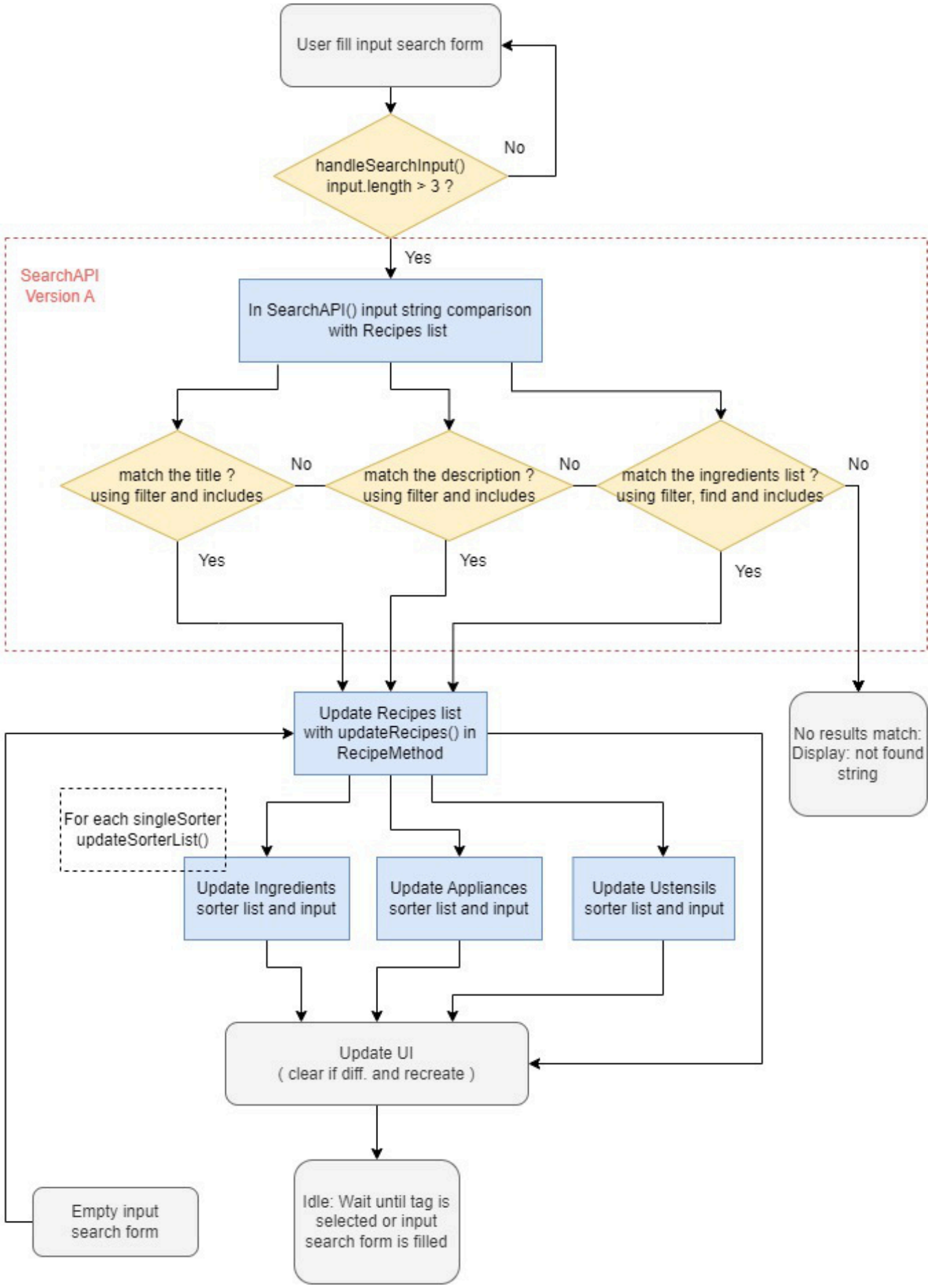

Figure 1 - Version A: Diagramme de la fonctionnalité en ES6 (Built-in Array method)

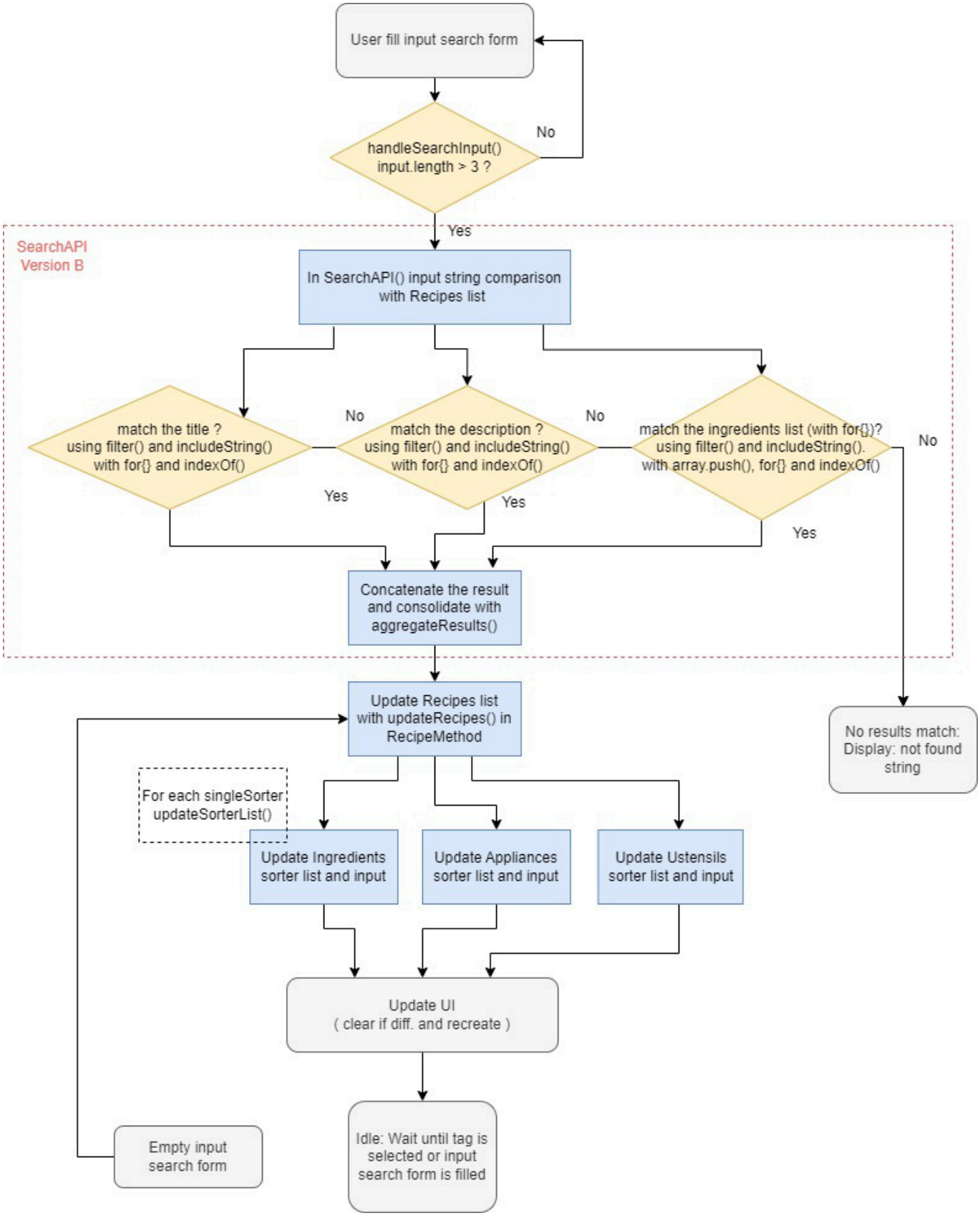

Figure 2 - Version : Diagramme de la fonctionnalité en Vanilla Javascript (ES5)

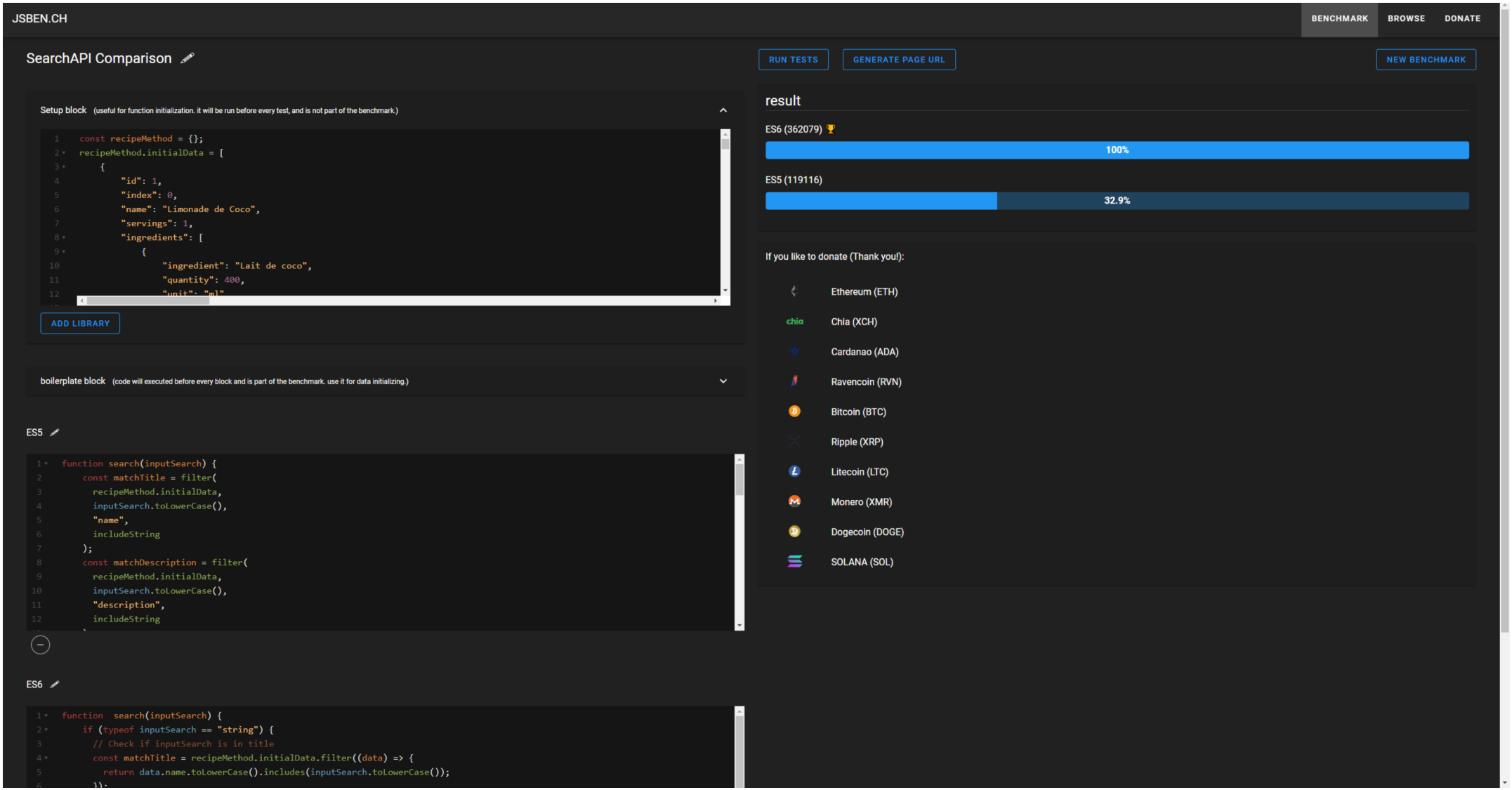

Figure 3 - Capture d'écran du Benchmark des deux solutions proposées.
https://jsben.ch/evkR9