

| Fonctionnalité : Recherche de recettes + tags | Fonctionnalité n°1 |
|---|--------------------|
| <p>Problématique : Permettre à l'utilisateur de trouver rapidement une recette grâce à la fonctionnalité de 'recherche'. La recherche pouvant s'effectuer en recherchant par ingrédients, par noms, par description, par ustensiles ou par appareils.</p> <p>Architecture : Permet de trier les recettes disponibles grâce à un champ de recherche ainsi que des filtres (tags) les ingrédients, appareils et ustensiles. La recherche renverra la(les) recette(s) dont le nom, la description ou les ingrédients contiennent la recherche. Pour réaliser cela nous allons développer deux algorithmes avec des méthodes de tableaux et avec des boucles natives.</p> | |
| <p>Nombres de champs : 1 champ de recherche (optionnelle)</p> <p>Nombres de sélecteurs : 3 sélecteurs (ingrédients, appareils, ustensiles) (optionnelle)</p> <p>Nombres de champs minimum : 0 champ et 0 sélecteurs (renvoie la totalité des recettes).</p> | |

| | |
|--|--|
| <p>Algorithme n°1 : Boucle avancée (map... filter ect..)</p> <p>Dans cette option, le code est composé de boucle et de méthodes de tableau avancées. Afin de traiter de la manière la plus rapide nos recettes, ainsi l'utilisateur obtient un résultat lié à ses recherches.</p> | |
| <p>Problématique : Accéder rapidement à une recette correspond au besoin de l'utilisateur grâce à la fonctionnalité de 'recherche'. Il peut rechercher soit par ingrédients, par noms de recettes, par appareils, par ustensils ou grâce à la description.</p> | |
| <p>Avantages :</p> <ul style="list-style-type: none"> - écriture du code plus simple. - Utilisation et pratique de méthodes de tableau avancées. - Permet une meilleure compréhension du code. | <p>Inconvénients :</p> <ul style="list-style-type: none"> - Avoir connaissance des méthodes et de comment elles fonctionnent ainsi de ce qu'elles retournent pour construire son algorithme. |

| | |
|--|---|
| <p>Algorithme n°2 : Boucle native (for..while)</p> <p>Dans cette option, le code est composé de boucle et de méthodes de tableau natives de JavaScript. Afin de traiter de la manière la plus rapide possible nos recettes, ainsi l'utilisateur obtient un résultat lié à ses recherches.</p> | |
| <p>Problématique : Accéder rapidement à une recette correspond au besoin de l'utilisateur grâce à la fonctionnalité de 'recherche'. Il peut rechercher soit par ingrédients, par noms de recettes, par appareils, par ustensils ou grâce à la description.</p> | |
| <p>Avantages :</p> <ul style="list-style-type: none"> - Utiliser les connaissances de bases de JS pour réussir à créer notre algorithme (travail en équipe facilité) - Décomposer / comprendre le fonctionnement des méthodes avancées de tableau. | <p>Inconvénients :</p> <ul style="list-style-type: none"> - Rend la compréhension et l'écriture du code un peu plus complexe. - réduit l'efficacité que peuvent avoir certaines méthodes avancées. |

| |
|--|
| <p>Solution retenue : Le second algorithme puisqu'il nous permet d'effectuer des recherches plus efficaces bien qu'étant moins maintenable.</p> |
|--|

