

Fiche d'investigation de fonctionnalité

Fonctionnalité : Recherche de recettes par la barre principale	Fonctionnalité #1 (Fiche Cas d'utilisation #3)
<p>Le filtrage se fait par la recherche d'une correspondance entre la valeur entrée par l'utilisateur dans la barre principale (lorsque supérieure ou égale à 3 lettres) et des mots ou groupes de lettres dans le titre, les ingrédients ou la description des recettes. La recherche s'actualise pour chaque nouveau caractère entré.</p>	
<p>Problématique : Nous cherchons à avoir un filtrage des recettes le plus rapide possible. La recherche principale affiche les premiers résultats le plus rapidement possible.</p> <p>Dans les deux cas, utilisation d'un proxy, le tableau des recettes triées est donc conservé pour effectuer la nouvelle recherche lorsqu'un nouveau caractère est entré. Par contre à la suppression d'un caractère, le filtrage se fait de nouveau sur le tableau de toutes les recettes.</p>	
<p>Perte de temps supplémentaire possible en plus de l'optimisation de la fonction elle-même :</p> <ul style="list-style-type: none"> - Biais dus à la casse ou aux fautes d'orthographe de l'utilisateur lors de la recherche, Solution => utilisation d'une fonction de correction d'orthographe et de casse - Affichage des recettes triées par ordre alphabétique. Solution => tri des recettes directement avant toute action de l'utilisateur 	

<p>Option 1 : Filtrage par l'utilisation de boucles natives de JavaScript (boucle for)</p> <p>Dans cette option, on crée dans un premier temps un objet tableau qui va recevoir toutes les recettes filtrées répondant au test. On boucle successivement sur chaque recette en vérifiant sur chacune de leurs propriétés "name", "description" et "ingrédients" les conditions de test. Dès qu'une propriété répond true au test alors la boucle s'arrête et la recette est copiée dans le tableau de résultat.</p>	
<p>Avantages</p> <ul style="list-style-type: none"> ⊕ Méthode native de base de JavaScript 	<p>Inconvénients</p> <ul style="list-style-type: none"> ⊖ code moins compacte. ⊖ nécessité de créer un nouveau tableau de results ⊖ nécessité de mettre la recette passant le test dans le tableau de results
<p>- Test jsben.ch</p> <p>- test 1000 recettes : sur le passage de co à coc puis coco et co => 229 opérations par seconde 97.45%</p> <p>- test 1000 recettes : sur le passage de far à fari puis far => 112 puis 116 soit 98.25% puis 99.15%</p>	

<p>Option 2 : Filtrage par l'utilisation des méthodes de l'objet array de JavaScript.</p> <p>Dans cette option, nous effectuons le filtrage des recettes par l'utilisation de la méthode filter. La méthode filter() crée et retourne un nouveau tableau contenant tous les éléments du tableau d'origine qui remplissent une condition déterminée par la fonction callback. L'avantage du système est la création d'un nouveau tableau. Et en utilisant une fonction fléchée, nous avons une fonction agréable et propre pour créer le même tableau..</p>	
<p>Avantages</p> <ul style="list-style-type: none"> ⊕ Création direct d'un nouveau tableau de results ⊕ Sur un tableau d'objet, le processus de filtrage reste relativement inchangé. (S'apparentant à une itération) ⊕ Code simplifié, méthode ayant fait ses preuves 	<p>Inconvénients</p> <ul style="list-style-type: none"> ⊖ Utilisation d'une fonction callback
<p>- Test jsben.ch</p> <p>- test 1000 recettes : sur le passage de 'co' à 'coc' puis 'coco' et 'co' => 235 opérations par seconde 100%</p> <p>- test 1000 recettes : sur le passage de 'far' à 'fari' puis 'far' => 114 puis 117 soit 100% puis 100%</p>	

<p>Solution retenue : Code relativement similaire. Les méthodes d'array sont suffisamment récentes pour être parfaitement supportées. La méthodologie de filtrage retenue est celle qui utilise la méthode filter car légèrement plus rapide lorsque la recherche se fait sur une grande quantité de recettes (test sur 1000 recettes).</p> <p>Toutefois la rapidité de la recherche passe principalement non par le choix de l'implémentation entre boucle for et méthode d'array mais plus par la préparation en amont (tel que le tri préalable des recettes utilisées par ordre alphabétique), par la correction orthographique pour la réponse la plus adéquate rendue lors du filtrage, par les raccourcis des contrôles effectués (grâce à if et else if), ainsi que par la réutilisation du tableau des</p>



Annexes

Diagramme d'activité Filtrer les recettes dans l'interface utilisateur

