



FEEDBACK



Introduction:

Ce projet est une application web de recettes développée en HTML/CSS/Javascript. Il permet aux utilisateurs de rechercher des recettes de cuisine, de filtrer les résultats par différents critères, et d'afficher les ingrédients et étapes de préparation. L'objectif est de créer une expérience intuitive et agréable pour trouver et découvrir de nouvelles recettes.

Problème N°1:

Description: Nous devons charger les recettes depuis un fichier JSON et les afficher dynamiquement sur le site en respectant un format de carte spécifique.

Solution: Nous avons créé un script JavaScript qui récupère le JSON qui utilise lors du chargement de la page et qui utilise une fonction `chargerRecettes` pour générer les cartes de recettes. Chaque carte est construite avec les éléments de la recette (titre, image, description, ingrédients) et insérée dans le conteneur HTML des recettes. On effectue la lecture grâce au `fetch`.

Problème N°2:

Description: Nous devons permettre aux utilisateurs de filtrer les recettes en sélectionnant des ingrédients, appareils et ustensiles. Le défi était de gérer des filtres multiples qui devaient être appliqués simultanément.

Solution: Nous avons créé un système de sélection avec des `select` HTML pour chaque type de filtre (ingrédients, appareils, ustensiles). Ensuite, nous avons utilisé une logique de filtrage dans la fonction `filterRecettes`, qui vérifie pour chaque recette si elle correspond aux critères de tous les filtres sélectionnés. Cela permet d'afficher uniquement les recettes qui utilisent les ingrédients, appareils et ustensiles choisis. La fonction permet que pour chaque ingrédients, appareils, ustensiles présents dans

Problème N°3:

Description: Pour une meilleure expérience utilisateur, les options de chaque filtre devaient être mises à jour en fonction des recettes restantes après filtrage. Par exemple, si seuls certains ingrédients sont présents dans les recettes filtrées, seuls ceux-ci doivent apparaître dans la liste des ingrédients.

Solution: Nous avons implémenté une fonction `afficherRecettes()` qui parcourt les recettes filtrées pour collecter tous les ingrédients, appareils et ustensiles uniques. La fonction `filterRecette` met à jour les options de chaque `select` de manière

dynamique pour refléter les éléments réellement disponibles dans le sous-ensemble de recettes filtrées.

Problème N°4

Description: Nous devons implémenter une barre de recherche permettant aux utilisateurs de taper un mot-clé pour filtrer les recettes, avec une recherche déclenchée après la saisie de trois caractères.

Solution: Nous avons ajouté un écouteur `input` à la barre de recherche, qui stocke le texte recherché dans une variable `searchQuery`. Lorsque la longueur de cette variable atteint trois caractères ou plus, nous utilisons cette valeur dans la fonction `filterRecettes` pour filtrer les recettes dont le titre correspond au mot-clé recherché.

Problème N°5

Description: Pour une interface plus intuitive, les filtres choisis devaient être affichés sous forme de tags. Chaque tag devait pouvoir être supprimé individuellement pour ajuster les filtres sans recharger la page.

Solution: Nous avons créé une fonction `updateTags` qui génère des tags pour chaque filtre sélectionné. Pour chaque tag, un bouton de suppression est inclus, permettant de retirer le filtre correspondant et de mettre à jour les recettes affichées. Cela améliore la flexibilité de l'interface de filtrage.

Des fonctions tel que `addFilter()` ou `removeFilter()` permette de gérer les filtres et les tags des filtres.

Problème N°6

Description: Implémentation d'un test unitaire afin de tester les fonctions de nos script Lors du **npm run test**, erreur de fonction.

Problème N°7

Description: Erreur de déploiement sur github. Lors du deployment sur github, les scripts ne veulent pas se lire.

Étude de l'Algorithme:

