



université  
PARIS-SACLAY

FACULTÉ  
DES SCIENCES  
D'ORSAY

# Le carnet des Saveurs

*“Where every ingredient speaks your language.”*

**Site Web de recette**

*Programmation Web*

**Fatoumata OULARE, Line HAJJAR**

*Licence 3 Informatiqu – S6 – G2*

*Année Universitaire : 2024-2025*

*Responsable de TD : Aquilina Al Khoury - [aquilina.al-khoury@universite-paris-saclay.fr](mailto:aquilina.al-khoury@universite-paris-saclay.fr)*

# **Sommaire**

## **1. Introduction**

## **2. Fonctionnalités**

## **3. Appel AJAX**

## **4. Difficultés + Solutions**

## **5. Conclusion**

## **Introduction**

Ce projet a pour objectif de développer un site de cuisine interactif. Chaque utilisateur peut avoir un ou plusieurs rôles parmi Chef, Traducteur et Cuisinier, chacun donnant accès à des fonctionnalités spécifiques.

Par défaut, tout utilisateur est Cuisinier. Il peut consulter des recettes, les aimer, laisser des commentaires, modifier les informations de son profil et soumettre une demande pour se voir attribué un ou plusieurs rôles supplémentaires (Chef et/ou Traducteur).

Le Traducteur dispose des fonctionnalités du Cuisinier, et peut en plus traduire les recettes incomplètes dans l'une des deux langues disponibles : français ou anglais.

Le Chef a la possibilité d'ajouter de nouvelles recettes ainsi que modifier complètement celles qu'il a créées.

Enfin, l'Administrateur dispose des mêmes droits que le Chef et le Traducteur, et a également la possibilité d'accepter ou de refuser les demandes de changement de rôle des utilisateurs.

---

## Fonctionnalités

### 1. Inscription d'un nouvel utilisateur

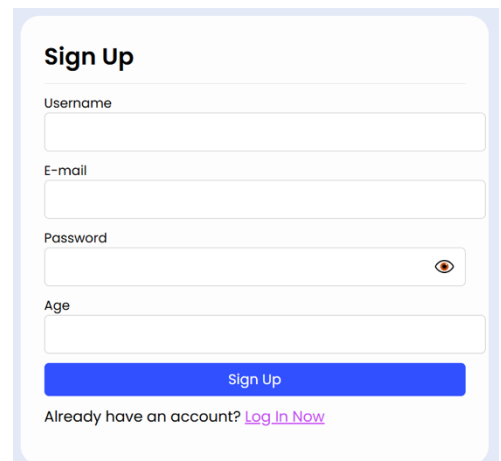
- Permet à un utilisateur de créer un compte en renseignant un nom d'utilisateur, un email, un mot de passe et un âge.

- Des vérifications sont effectuées :

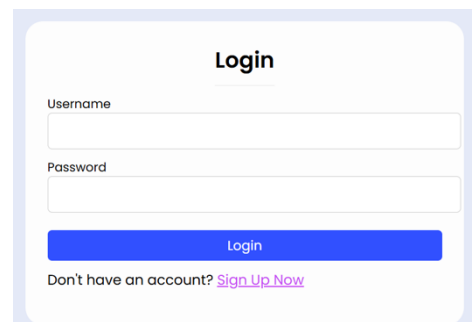
- Nom d'utilisateur et email doivent être uniques
- L'email doit être valide
- Le mot de passe doit contenir au minimum 6 caractères, une majuscule et un chiffre
- L'âge doit être un chiffre
- En cas de non-respect des restrictions, un message d'erreur spécifique est affiché.

### 2. Connexion au compte

- Permet aux utilisateurs déjà inscrits de se connecter avec leur nom d'utilisateur et mot de passe.
- En cas d'erreur d'identifiants, un message d'erreur est affiché.



The 'Sign Up' form is a light blue rounded rectangle. It contains the title 'Sign Up' in bold. Below it are four input fields: 'Username', 'E-mail', 'Password' (with a toggle eye icon), and 'Age'. A blue 'Sign Up' button is at the bottom. Below the button is a link: 'Already have an account? [Log In Now](#)'.



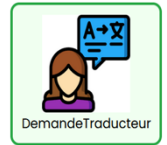
The 'Login' form is a light blue rounded rectangle. It contains the title 'Login' in bold. Below it are two input fields: 'Username' and 'Password'. A blue 'Login' button is at the bottom. Below the button is a link: 'Don't have an account? [Sign Up Now](#)'.

### 3. Choix du rôle après inscription

- Une fois inscrit, l'utilisateur est invité à choisir s'il souhaite devenir Chef, Traducteur, ou les deux.
- S'il ne sélectionne aucun rôle, il est automatiquement affecté au rôle de Cuisinier.



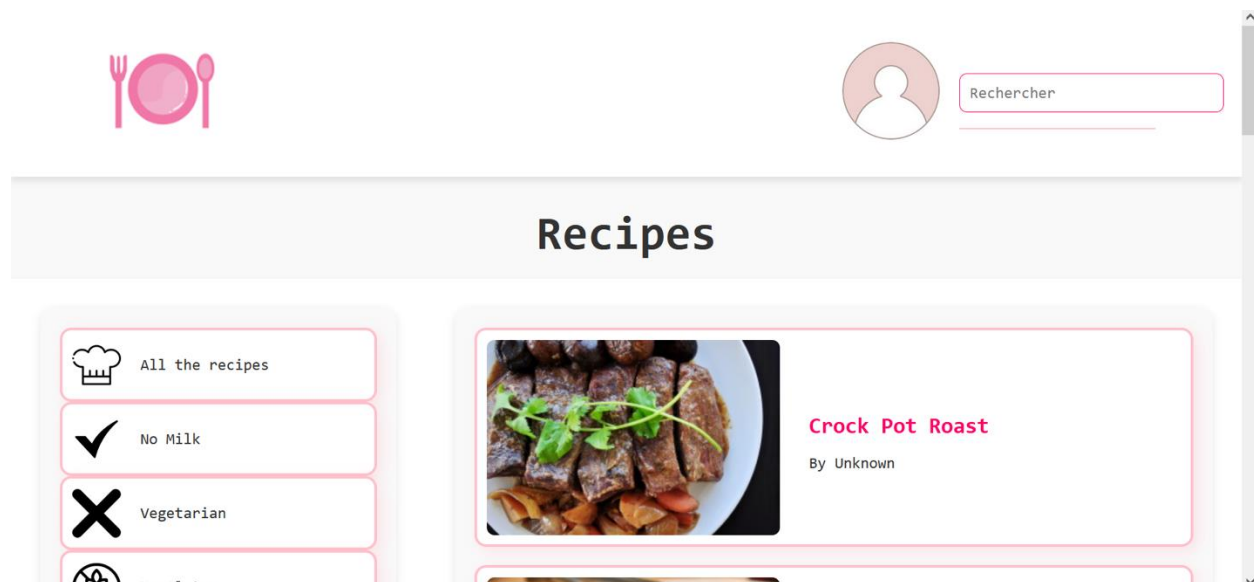
Choisir un rôle



Valider

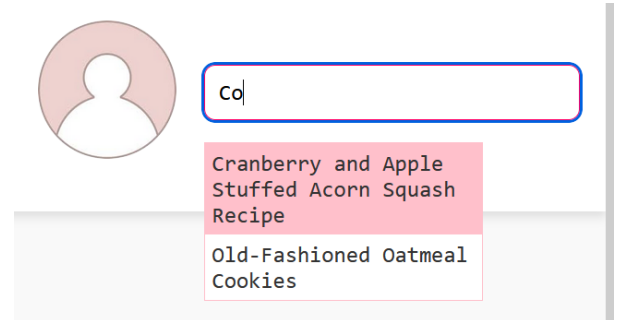
### 4. Accueil du site / Home

- La page d'accueil affiche tous les boutons de navigation : Home, Profil, Recherche.
- Selon le rôle de l'utilisateur, des boutons supplémentaires apparaissent : "Chef" pour les Chefs, "Admin" pour l'Administrateur.
- La page affiche également la liste de toutes les recettes disponibles.



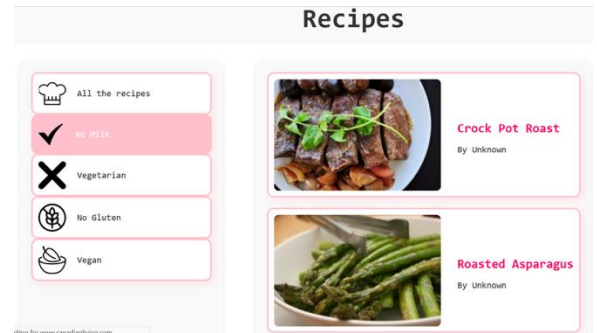
## 5. Recherche instantanée de recettes

- Une barre de recherche permet de trouver des recettes par titre.
- À chaque frappe de caractère, les recettes contenant le caractère dans leur titre sont affichées instantanément, sans rechargement de la page.
- En cliquant sur un résultat, l'utilisateur est redirigé vers la page de la recette.



## 6. Navigation par filtres de recettes

- Des boutons de filtres permettent d'afficher uniquement les recettes correspondant à un critère particulier.



## 7. Affichage dynamique au survol

- Tous les boutons du site changent de couleur lorsqu'on passe la souris dessus (hover), améliorant l'interactivité.



## 8. Consultation des recettes

- Chaque recette possède sa propre page détaillée contenant toutes ses informations, le nombre de likes et les commentaires.



Rechercher



## Crock Pot Roast

By Unknown

Without: NoGluten NoMilk

Time: 7 hours 3 minutes



Unlike

### Ingredients

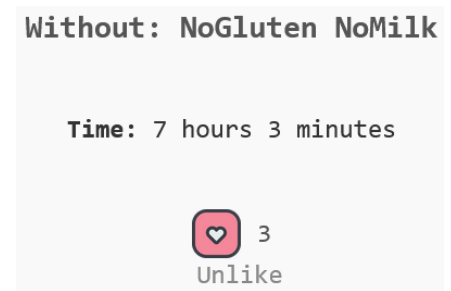
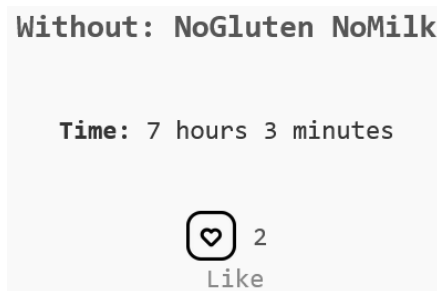
1 of beef roast  
1 package of brown gravy mix  
1 package of dried Italian salad dressing mix  
1 package of dry ranch dressing mix  
1/2 cup of water  
of

### Steps

1. Place beef roast in crock pot.
2. Mix the dried mixes together in a bowl and sprinkle over the roast.
3. Pour the water around the roast.
4. Cook on low for 7-9 hours.

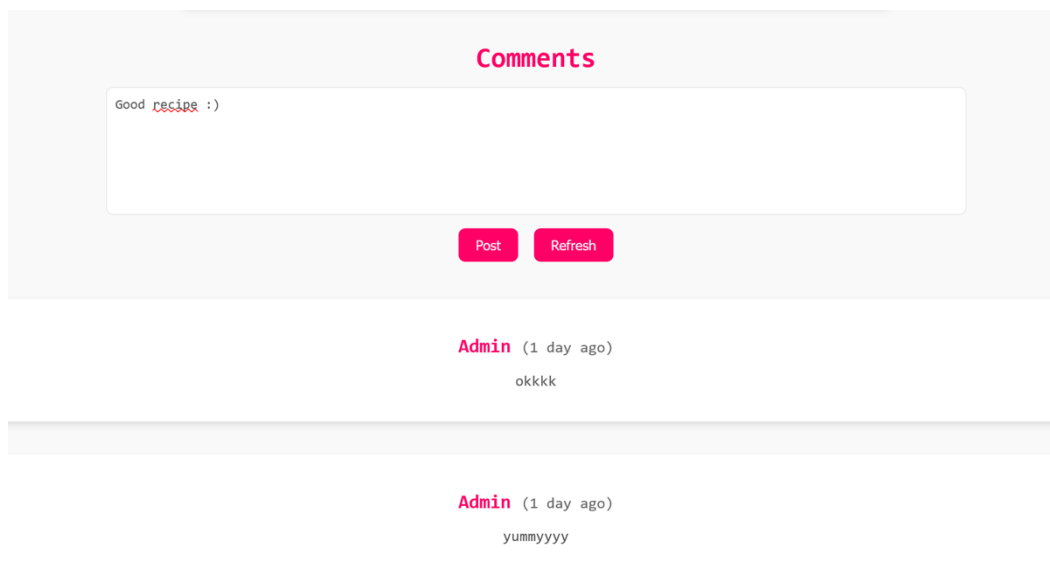
## 9. Aimer une recette

- Les utilisateurs peuvent **aimer** une recette (le nombre de likes est mis à jour).



## 10. Publication de commentaires

- Les utilisateurs peuvent **commenter** une recette.
- Chaque commentaire affiche **le nom de l'utilisateur**, le **message** et le **temps écoulé** depuis la publication. OI



## 11. Modification de recettes (Chef et Admin)

- Les utilisateurs ayant le rôle de **Chef** peuvent accéder à une page de modification complète de leurs propres recettes à partir de la page de la recette concernée.
- L'Admin peut accéder à une page de modification complète de toutes les recettes à partir de la page de la recette concernée.

English Version	Version Française
<b>Recipe Name:</b> <input type="text" value="Crock Pot Roast"/>	<b>Nom de la recette:</b> <input type="text" value="Rôti à la cocotte"/>
<input type="checkbox"/> Vegan <input type="checkbox"/> Vegetarian <input checked="" type="checkbox"/> NoMilk <input checked="" type="checkbox"/> NoGluten	
<b>Ingredients:</b> <ul style="list-style-type: none"> <li>1 Meat beef roast</li> <li>1 package Baking brown gravy mix</li> <li>1 package Condiments dried Italian salad dress</li> <li>1 package Condiments dry ranch dressing mix</li> <li>1/2 cup Drinks water</li> </ul>	<b>Ingrédients:</b> <ul style="list-style-type: none"> <li>1 Viande rôti de bœuf</li> <li>1 sachet Aide culinaire mélange sauce ou vin</li> <li>1 package Condiments</li> <li>1 package Condiments</li> <li>1/2 verre Boisson eau</li> </ul>
<b>Steps:</b> <ol style="list-style-type: none"> <li>Place beef roast in crock pot.</li> <li>Mix the dried mixes together in a bowl and sprink</li> <li>Pour the water around the roast.</li> <li>Cook on low for 7-9 hours.</li> </ol>	<b>Étapes:</b> <ol style="list-style-type: none"> <li>Placer le rôti dans la cocotte en fonte</li> <li>Mélanger les sachets de sauces en poudre dans</li> <li>Arroser le rôti</li> <li>Cuire à feu doux entre 7 et 9 heures.</li> </ol>

## 12. Traduction de recettes (Traducteurs et Admin)

- Les utilisateurs ayant le rôle de Traducteur ou Admin peuvent traduire les champs incomplets d'une recette (dans l'autre langue disponible : français ou anglais) à partir de la page de la recette concernée.

English Version	Version Française
<b>Recipe Name:</b> <input type="text" value="Roasted Asparagus"/>	<b>Nom de la recette:</b> <input type="text" value="Asperges rôties"/>
<b>Ingredients:</b> <ul style="list-style-type: none"> <li>Quantity: 1 lb Type: Produce Name: asparagus</li> <li>Quantity: 1 1/2 tbsp Type: Condiments Name: olive oil</li> <li>Quantity: 1/2 tsp Type: Baking Name: kosher salt</li> </ul>	<b>Ingrédients:</b> <ul style="list-style-type: none"> <li>Quantité Type Nom</li> <li>Quantité Type Nom</li> <li>Quantité Type Nom</li> </ul>
<b>Steps:</b> <ol style="list-style-type: none"> <li>Preheat oven to 425°F.</li> <li>Cut off the woody bottom part of the asparagus spears and discard.</li> <li>With a vegetable peeler, peel off the skin on the bottom 2-3 inches of the spears (this keeps the asparagus from being "slimy," and if you eat asparagus you know what I mean by that).</li> <li>Place asparagus on foil-lined baking sheet and drizzle with olive oil.</li> <li>Sprinkle with salt.</li> <li>With your hands, roll the asparagus around until they are evenly coated with oil and salt.</li> <li>Roast for 10-15 minutes, depending on the thickness of your stalks and how tender you like them.</li> <li>They should be tender when pierced with the tip of a knife.</li> <li>The tips of the spears will get very brown but watch them to prevent burning.</li> <li>They are great plain, but sometimes I serve them with a light vinaigrette if we need something acidic to balance out our meal.</li> </ol>	<b>Étapes:</b> <ol style="list-style-type: none"> <li>Traduction de l'étape</li> <li>Traduction de l'étape</li> <li>Traduction de l'étape</li> <li>Traduction de l'étape</li> <li>Traduction de l'étape</li> <li>Traduction de l'étape</li> <li>Traduction de l'étape</li> <li>Traduction de l'étape</li> <li>Traduction de l'étape</li> <li>Traduction de l'étape</li> </ol>

## 13. Gestion du profil utilisateur

- Depuis la page d'accueil, les utilisateurs peuvent accéder à leur profil personnel.
- Ils peuvent modifier leurs informations personnelles et soumettre une demande de changement de rôle (Chef et/ou Traducteur).
- Un bouton permet également de se déconnecter du site.



**Modifier le profil**

**Nom d'utilisateur:**

**E-mail:**

**Âge:**

**Rôle:**  
☒ DemandeChef  
☐ DemandeTraducteur

**Profile**

**User:** LineH**E-mail:** line.hajjar@universite-paris-saclay.fr**Age:** 20**Role:** Cuisinier, DemandeChef, DemandeTraducteur

## 14. Espace Chef

- Les Chefs disposent d'un espace dédié affichant la liste de leurs propres recettes.
- Ils peuvent cliquer sur une recette pour la consulter ou accéder à un formulaire pour créer une nouvelle recette.

**Chef**

**Crock Pot Roast**  
By LineH

## 15. Création de nouvelles recettes

- Depuis l'espace Chef, un formulaire permet de créer et soumettre une nouvelle recette.

### English Version

Recipe Name:

**Without:**

☐ Vegan

☐ Vegetarian

☐ NoMilk

☐ NoGluten

**Ingredients:**

Quantity

Name

Type

**Steps:**

**Timers:**

### Version Française

Nom de la recette:

**Sans:**

☐ Vegan

☐ Vegetarian

☐ NoMilk

☐ NoGluten

**Ingrédients:**

Quantité

Nom

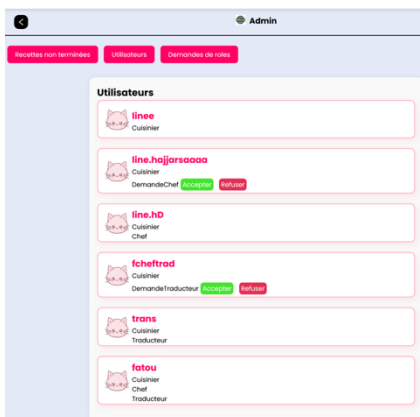
Type

**Etapas:**

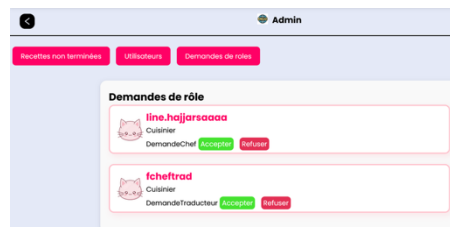
**Temps:**

## 16. Espace Admin

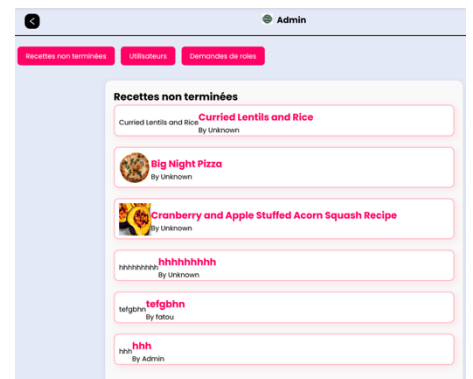
- Les Administrateurs disposent d'une page spéciale comportant plusieurs fonctionnalités de gestion :
  - Liste des utilisateurs** : affiche tous les utilisateurs avec leur nom et leurs rôles.
  - Gestion des demandes de rôle** : permet d'accepter ou de refuser les demandes pour devenir Chef ou Traducteur.
  - Liste des recettes incomplètes** : affiche les recettes ayant des champs manquants ; permet de consulter ou modifier ces recettes.



1



2



3

---

## Appel AJAX

### 1. Gérer l'inscription de l'utilisateur :

La fonction *Connexion ()* du fichier *signup.php* est appelée lorsque l'utilisateur souhaite se connecter en cliquant sur le bouton "Sign Up" de la page *Sign Up*.

- Un appel AJAX en méthode **POST** est envoyé au fichier **signup.php**.
- L'appel ajax récupère les champs *username*, *email*, *password* et *age* des formulaires.
- Ce fichier PHP lit le fichier *users.json* contenant les informations sur les utilisateurs et vérifie si le *username* et l'*email* sont unique (s'ont pas été déjà pris par un autre utilisateur). Il stocke ces informations dans une session pour l'utiliser dans d'autre page plus-tard.
- Si la requête réussit, l'utilisateur est redirigé vers la page *home.php*.
- En cas d'échec, un message d'erreur est affiché dans la balise div de classe "**Error: network problem**" sans recharger la page.

### 2. Gérer la connexion de l'utilisateur :

La fonction *connexion ()* du fichier *signup.php* est appelée lorsque l'utilisateur souhaite se connecter en cliquant sur le bouton "Login" de la page *Sign Up*.

- Un appel AJAX en méthode **POST** est envoyé au fichier **login.php**.
- L'appel ajax récupère les champs *username*, *password* des formulaires.
- Ce fichier PHP lit le fichier *users.json* contenant les informations sur les utilisateurs et vérifie si le *username* et le *password* récupéré correspondent dans le *json*. Il stocke ces informations dans une session pour l'utiliser dans d'autre page plus-tard.
- Si la requête réussit, l'utilisateur est redirigé vers la page *home.php*.
- En cas d'échec, un message d'erreur est affiché dans la balise div de classe "**Erreur : Informations mal chargés**" sans recharger la page.

### 3. Affichage de la liste des utilisateurs dans la page Admin

La fonction **getUsers()** du fichier **Admin.php** est appelée lorsque l'admin souhaite afficher la liste de tous les utilisateurs en cliquant sur le bouton "Utilisateurs" de la page Admin ou que la celle-ci est rafraîchie.

- Un appel **AJAX** en méthode **GET** est envoyé au fichier **users.php**.
- Ce fichier PHP lit le fichier *users.json* contenant les informations sur les utilisateurs, et renvoie du contenu HTML qui représente tous les utilisateurs.

- Si la requête réussit, la réponse HTML est insérée dans l'élément ayant l'id `#liste_users` de la page, sans recharger la page.
- En cas d'échec, un message d'erreur est affiché dans `#liste_users`.

#### 4. Gestion des demandes de changement de rôle

La fonction **`demande(role, statut, user)`** du fichier **`Admin.php`** est appelée lorsque l'admin souhaite accepter ou refuser une demande de changement de rôle en cliquant sur les boutons *Accepter* ou *Refuser* de chaque utilisateur de la page *Admin*.

- Un appel **AJAX** en méthode **GET** est envoyé au fichier **`users.php`** avec 3 paramètres :
  - **Rôle** : DemandeChef ou DemandeTraducteur
  - **Statut** : accepter ou refuser
  - **User** : nom de l'utilisateur concerné
- Ce fichier PHP modifie le fichier **`users.json`** contenant les informations sur les utilisateurs. Si la demande est acceptée, le rôle demandé est remplacé par Chef ou Traducteur et si elle est refusée, ce rôle est supprimé de la liste de rôle de l'utilisateur. **`users.php`** met à jour automatiquement la liste des utilisateurs et la renvoie en contenu HTML.
- Les réponses sont affichées comme pour **`getUsers()`**.

#### 5. Affichage de la liste des recettes non terminées

La fonction **`getTermine()`** du fichier **`Admin.php`** est appelée lorsque l'admin souhaite afficher la liste de toutes les recettes dont certaines informations sont manquantes en cliquant sur le bouton "Recettes non terminées" de la page *Admin*.

- Un appel AJAX en méthode **GET** est envoyé au fichier **`termine.php`** sans passer de paramètre particulier.
- Ce fichier PHP lit le fichier **`recipes.json`** contenant les informations sur les recettes. Pour chaque recette, il vérifie si certains champs importants sont absents. Si au moins un des champs est manquant, la recette est considérée comme "non terminée" et la liste est générée en revoyant un contenu HTML. Chaque recette est cliquable et redirige vers la page de cette recette.
- Si la requête réussit, la réponse HTML est insérée dans l'emplacement prévu dans la page Admin, sans recharger la page.
- En cas d'échec, un message d'erreur est affiché à la place.

#### 6. Affichage de la liste des demandes de changement de rôle

La fonction **getDemandes(demande)** du fichier **Admin.php** est appelée lorsque l'admin souhaite afficher uniquement la liste des utilisateurs ayant soumis une demande pour devenir **Chef** ou **Traducteur** en cliquant sur le bouton "Demandes de rôle" de la page **Admin**.

- Un appel **AJAX** en méthode **GET** est envoyé au fichier **users.php** avec un paramètre demande qui permet de sélectionner que les utilisateurs ayant fait une demande de changement de rôle.
- Ce fichier PHP lit le fichier **users.json** contenant les informations sur les utilisateurs et filtre les utilisateurs ayant pour rôle DemandeChef et/ou DemandeTraducteur, puis génère la liste en revoyant un contenu HTML.
- Les résultats sont affichés de la même manière que pour **getUsers()**, sans rechargement de la page.

## 7. Affichage de la liste des recettes personnelles

La fonction **myRecipes(author)** du fichier **chef.php** est appelée lorsque le chef ou l'admin souhaite afficher la liste de toutes les recettes qu'il a créé chargeant la page chef.

- Un appel **AJAX** en méthode **GET** est envoyé au fichier **recipes.php** avec comme paramètre le nom de l'utilisateur dont on veut voir les recettes.
- Ce fichier PHP lit le fichier **recipes.json** contenant les informations sur les recettes. Il filtre les recettes en affichant uniquement celles dont le champ **Author** correspond exactement au paramètre reçu, puis génère la liste en revoyant un contenu HTML. Chaque recette est cliquable et redirige vers la page de cette recette.
- Si la requête réussit, la réponse HTML est insérée dans l'emplacement prévu dans la page **Chef**, sans recharger la page.
- En cas d'échec, un message d'erreur est affiché à la place.

## 8. Filtrage des recettes

La fonction **getRecipes(nomfiltre)** du fichier **home.php** est appelée lorsque l'utilisateur souhaite afficher la liste des recettes par catégorie en sélectionnant un filtre sur la page d'accueil.

- Un appel **AJAX** en méthode **GET** est envoyé au fichier **recipes.php** avec comme paramètre nomfiltre, qui est le nom du filtre sélectionné.
- Ce fichier PHP lit le fichier **recipes.json** contenant les informations sur les recettes. Il filtre les recettes en affichant uniquement celles dont le champ Without correspond exactement au paramètre reçu, ou affiche toutes les recettes si nomfiltre est "tout". Il génère la liste en revoyant un contenu HTML. Chaque recette est cliquable et redirige vers la page de cette recette.
- Si la requête réussit, la réponse HTML est insérée dans l'emplacement prévu dans la page **Home**, sans recharger la page.
- En cas d'échec, un message d'erreur est affiché à la place.

## Difficultés

L'une de nos premières difficultés a été de réussir à poster un commentaire et l'afficher sur la page de la recette, car à l'envoi du formulaire, on perdait l'**id** nécessaire pour afficher la bonne recette. Ce problème a finalement été résolu en utilisant un champ input de type hidden dans lequel on stocke l'**id** avant l'envoi du formulaire.

On souhaitait également ajouter dans le formulaire d'ajout de recette un bouton radio pour choisir le type de chaque ingrédient. Cependant, nous avons rencontré des difficultés : lorsqu'il y avait plusieurs ingrédients, il n'était possible de choisir qu'un seul type pour tous les ingrédients. Face à ce problème, nous avons finalement décidé d'utiliser un **textarea** à la place.

Au départ, l'initialisation du rôle utilisateur posait problème car nous avons supposé que le rôle était une **valeur unique**, à l'image de l'**username**, que l'on stockait simplement dans la session via une ligne comme : `$_SESSION['username'] = $_POST['username'];`

Cependant, dans notre cas, un utilisateur peut se voir attribuer **plusieurs rôles simultanément** (par exemple : *Chef*, *Traducteur*, etc.). Pour gérer cela correctement, nous avons remplacé le champ de rôle unique par une **liste de cases à cocher (checkbox)** dans le formulaire. Cela permet à l'utilisateur de sélectionner plusieurs rôles à la fois. Les rôles sélectionnés sont ensuite récupérés sous forme de **tableau (array)** et stockés dans la session via :

```
$_SESSION['role'] = $_POST['selected_role'];
```

Actuellement, les URL d'images invalides ne sont pas encore automatiquement remplacées par une image par défaut, qui pourrait améliorer l'affichage des recettes.

Nous avons rencontré un problème avec les cookies de session car **JavaScript utilise des millisecondes** tandis que **PHP utilise des secondes** pour le temps. Cette différence causait une mauvaise interprétation du temps d'expiration côté serveur. Résultat : la session expirait immédiatement après connexion.

## Conclusion

Ce projet nous a permis d'acquérir des compétences en **HTML, CSS, JSON, JavaScript**, et plus particulièrement en **AJAX**, un outil essentiel pour la fluidité et la réactivité des sites web. Nous avons appris à établir une **liaison entre le côté client et le côté serveur**, notamment pour **stocker les informations de session des utilisateurs**, les **réutiliser sur différentes pages**, et les **afficher dynamiquement**. Nous avons également manipulé des fichiers JSON pour **lire, filtrer, extraire et afficher des recettes**, mais aussi pour **écrire de nouvelles données** dans ces fichiers. Nous nous sommes inspirés de sites web professionnels disposant d'un certain **niveau de sécurité** afin de **protéger les données des utilisateurs**, en scriptant les mots de passe. Ce projet nous a ainsi offert une vision concrète du fonctionnement d'un site web interactive et sécurisée.