

PROJET WEB

Introduction

Dans un contexte où le partage culinaire en ligne connaît un essor constant, la création d'un site web multilingue de gestion de recettes de cuisine répond à un besoin à la fois pratique et communautaire. Ce projet s'inscrit dans cette dynamique, avec pour objectif de développer une plateforme accessible en français et en anglais, permettant aux utilisateurs d'interagir autour de recettes variées, tout en respectant certaines contraintes alimentaires (végétarien, vegan, sans gluten, etc.).

Le point de départ de ce projet repose sur un fichier JSON contenant des recettes partiellement renseignées. Il s'agira donc d'enrichir ces données et de structurer l'information de manière cohérente et fonctionnelle au sein du site. L'application permettra également de gérer plusieurs types d'utilisateurs selon des rôles bien définis : cuisiniers, chefs, traducteurs et administrateur. Chacun disposera de droits spécifiques allant de la publication de commentaires et de photos jusqu'à la traduction, la création ou la validation de recettes, ainsi que la gestion des utilisateurs.

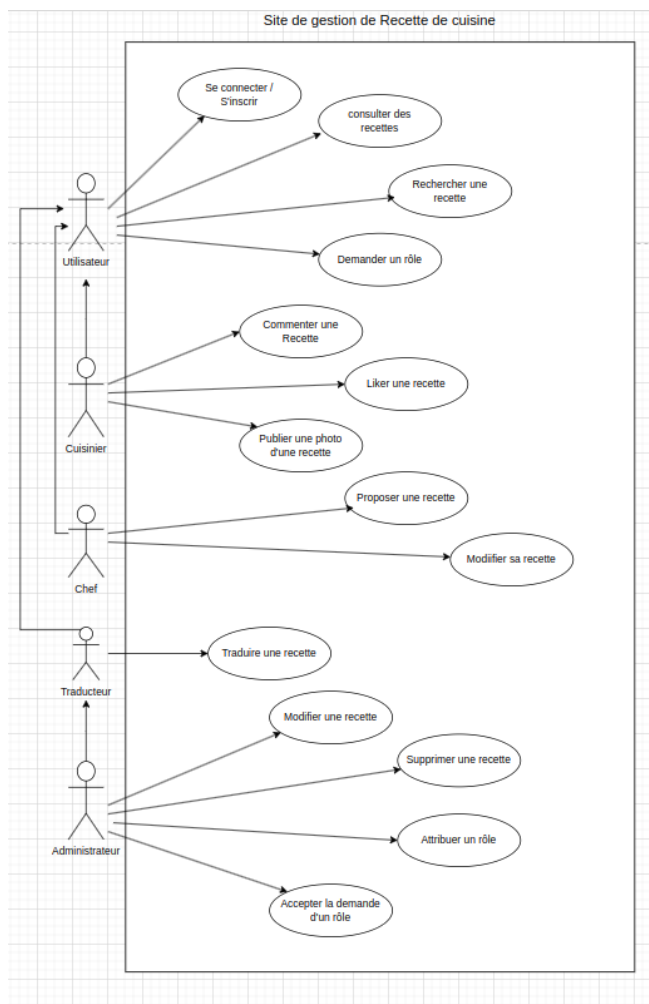
Ce rapport détaille les différentes étapes de conception et de développement du site, les choix techniques effectués, ainsi que les défis rencontrés et les solutions apportées au fil du projet.

DESIGN DU PROJET

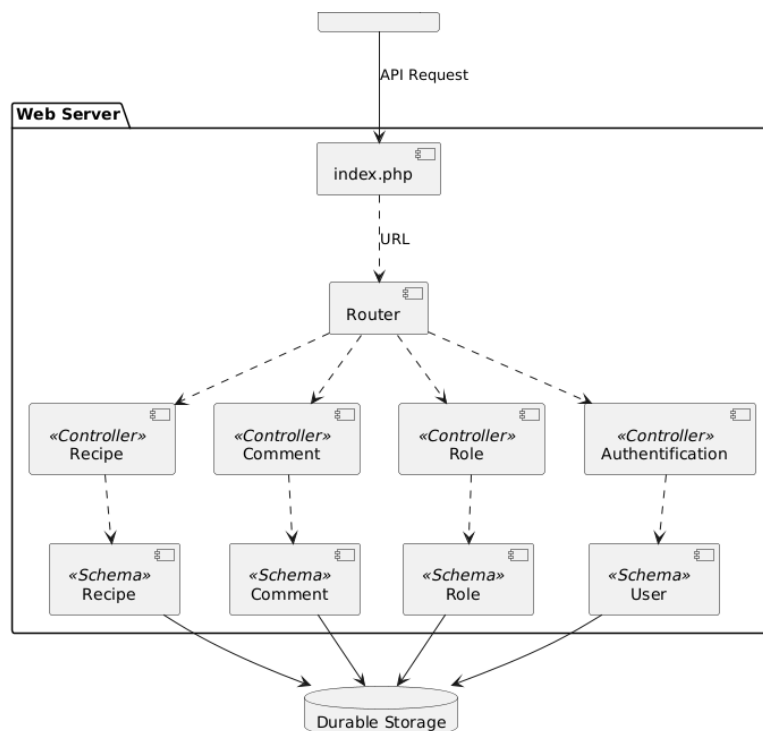
Avant de réellement commencer le projet nous avons fait plusieurs diagramme afin de visualiser à l'avance à quoi doit ressembler l'application.

Nous avons pour cela fait des diagrammes en cour, avant même de se lancer sur quelconque code.

Nous avons notamment ce diagramme de cas, qui nous montre bien tous les rôles et les actions qu'ils doivent être capable de réaliser



Nous avons aussi ce diagramme qui montre que les demandes, notamment les appels ajax, vont devoir être gérés par un routeur, lui-même géré par index.php qui sera comme la tour de contrôle qui redirige les demandes au bon endroit.



STRUCTURE DU PROJET

au niveau de la structure du projet, nous avons donc fait une classe php router, qui nous permet de donner des adresses à appeler pour nos appels AJAX, et qui nous permet aussi d'avoir une seule sur lequel on traite les liens entre les codes java script et les code php.

```
> index.php
<?php
header('Access-Control-Allow-Origin: http://localhost');
header('Access-Control-Allow-Methods: GET, POST, OPTIONS');
header('Access-Control-Allow-Headers: Content-Type, Accept, Accept-Language, Accept-Encoding');

require_once 'router.php';
require_once 'RecipiesManager.php';
require_once 'UserManager.php';

//Initialize the router and recipes manager
$router = new router();
$UserManager = new UserManager(__DIR__);
$RecipiesManager = new RecipiesManager(__DIR__);

//Handle the registration
$router->register('POST', '/registration', [$UserManager, 'handleRegisterRequest']);
//Handle the login
$router->register('POST', '/login', [$UserManager, 'handleLoginRequest']);
//Handle the Compte
$router->register('POST', '/compteManager', [$UserManager, 'handleCompteRequest']);
//Handle the modifier compte
$router->register('POST', '/modifierCompte', [$UserManager, 'handleModifierCompteRequest'])
```

Nous avons aussi une classe js par classe html, pour rendre plus lisible les codes et faciliter la compréhension du projet dans son ensemble. Cette approche nous a aussi facilité la tâche pour gérer certains bug apparu en cours de projet, ou pour modifier certaines fonctions précises quand cela était nécessaire.

Nous avons toujours séparé le html, le css, et le js dans des fichiers différents, encore une fois afin de rendre cela plus lisible et éviter les fichiers trop longs.

LES APPELS AJAX DETAILLÉS :

Nom	Route	Cœur de la requête	réponse	Description
S'inscrire	POST, '/registration'	email: email, username: username, password: password, Confirm: confirm, translator: translator, Chef: chef	/	On prend les informations du form Pour créer un nouvel utilisateur Dans le dossier donnees.json
Se connecter	POST, '/login'	email: email, password: password	true or false	On prend les données rentrées par L'utilisateur et on regarde si on Retrouve le couple email mdp dans Un des compte et si oui on Autorise la connexion
Prendre les infos du Compte	POST, '/compteManager'	email: Email	Array(User)	On récupère les informations de L'utilisateur qui c'est connecté Afin de les stocker dans la session
Modifier Compte	POST, '/modifierCompte'	email: Email, Password: nouvelPassword, username: nouvelUsername, Translator: nouvelTranslator, chef: nouvelChef	/	On modifie les informations de L'utilisateur qui est connecté
Ajouter une recette	POST, '/ajouterRecette'	JSON.stringify(Recipie),	/	On crée une nouvelle Recette
Afficher toutes les Informations d'une Recette	POST, '/recetteManager'	thisRecipeName: recipeName	Array(Recipe)	On récupère les informations de la Recette qui est observé afin D'afficher ses informations
Modifier Recette	POST, '/modifRecipe'	JSON.stringify(Recipie)	/	On modifie les informations de La recette si on le peut
Ajouter un commentaire	POST, '/ajouterComment'	username: username, comment: commentText, recipeName: recipeName	/	On ajoute un commentaire sous une Recette spécifique

QUESTIONNAIRES :

En générale, nous avons fait en sorte que quasiment aucun champs ne reste vide pour éviter tout problème par la suite, pour la création de compte par exemple, nous avons aussi fait une confirmation de mot de passe, le champ email a aussi des conditions, il doit contenir un @, par de guillemets etc. on vérifie aussi que les Email ne sont pas déjà prit, de même pour les noms d'utilisateurs.

Pour les création et modifications de recette, tous les champs doit être rempli, certain champ refuse des lettres, comme le Timer.

The image shows a registration form titled "INSCRIPTION" on a dark background. The form includes an email input field containing "essaie", which has a red underline. Below it, a white error message box with an orange exclamation mark icon states: "Please include an '@' in the email address. 'essaie' is missing an '@'." The form also features two password input fields, each with a series of dots. Below the password fields are two checkboxes: "Je veux être traducteur" and "Je veux être chef cuisinier". At the bottom of the form is a large blue button labeled "S'inscrire". Below the button, the text "Déjà un compte ?" is followed by a blue link "Connectez-vous ici".

APPARENCE VISUELLE

Nous avons opté pour un design propre et sobre afin de garantir une expérience utilisateur claire et agréable. L'objectif principal de notre interface est de faciliter la lecture et l'interaction, en mettant en avant les recettes de manière intuitive. Les éléments visuels ont été réduits à l'essentiel, avec une hiérarchisation de l'information bien pensée, afin que chaque utilisateur, qu'il soit chef, traducteur ou administrateur, puisse naviguer facilement dans l'application et accomplir ses tâches sans confusion.

page de recette


Crock Pot Roast

Roti à la cocotte

Auteur : Unknown

Sans :

Terminé : on



Ingrédients

- 1 beef roast (Meat)
- 1 package brown gravy mix (Baking)
- 1 package dried Italian salad dressing mix (Condiments)
- 1 package dry ranch dressing mix (Condiments)
- 1/2 cup water (Baking)

Étapes

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.

Timers

- 1
- 1
- 1
- 420

Modifier la recette

Retour à l'accueil

Commentaires

Écrivez votre commentaire ici....

Envoyer le commentaire

DIFFICULTÉS RENCONTRÉES :

Nous avons rencontré plusieurs difficulté durant notre projet, notamment sur le back-end :

- Récupérer les réponses des appels ajax dans le bon format :
Nous n'arrivions pas à récupérer les données que nous voulions afficher, car elle arrivaient toutes sous forme de string JSON au lieu de sous leur forme d'objet originel. Nous avons du les envoyer dans un Array à un seul objet et ensuite utiliser parseJSON afin de pouvoir les récupérer sous le bon format.
- Gérer tous les soucis lié aux modifications de compte. Changer le username ne modifie pas celui enregistré lors des commentaires et surtout celui utiliser lors de la création de recette pour les chefs, ce qui fait que leurs recette ne leur « appartiennent plus » donc nous avons du enlever cette possibilité.

FONCTIONNALITÉ ABSENTE :

L'administrateur n'as pas besoin d'accepter les rôles car toutes les demande sont par default accepté. Par manque de temps elle n'a pas pu être rajouter, par une mauvaise gestion de l'ordre de création des fonctionnalités, celle ci aurait du changer énormément de fonctions.