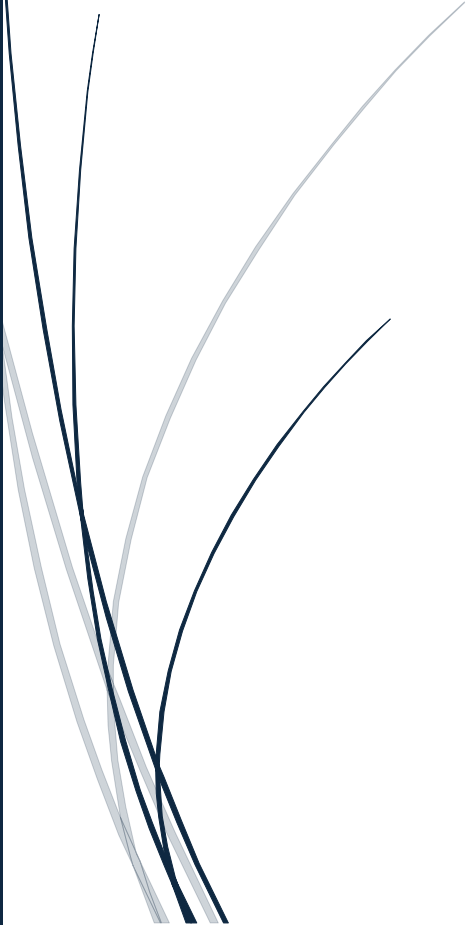


14/01/2025

# LyonEats

Cahier des Charges



EL BOUAZZAOUI Ali, MESSAOUDI Amin, BEN  
AMMAR Rayen, BENNANI Ilyass, NADIR Ayoub

## Table des matières

|      |                                   |   |
|------|-----------------------------------|---|
| 1.   | Présentation générale .....       | 1 |
| 2.   | Fonctionnalités principales ..... | 1 |
| 2.1. | Carte interactive .....           | 1 |
| 2.2. | Recherche et filtrage .....       | 1 |
| 2.3. | Détails du restaurant .....       | 1 |
| 2.4. | Avis et commentaires .....        | 1 |
| 2.5. | Itinéraires.....                  | 2 |
| 3.   | Technologies .....                | 2 |
| 3.1. | Frontend .....                    | 2 |
| 3.2. | Backend .....                     | 2 |
| 3.3. | Base de données .....             | 2 |
| 3.4. | Authentification.....             | 2 |
| 3.5. | API de cartographie.....          | 2 |
| 4.   | Sécurité .....                    | 2 |
| 5.   | Hébergement .....                 | 3 |
| 6.   | Maintenance .....                 | 3 |
| 7.   | Design .....                      | 3 |
| 8.   | Tests.....                        | 3 |

# 1. Présentation générale

**Le projet consiste à développer un site web permettant aux utilisateurs de consulter une carte interactive des restaurants de Lyon. Les utilisateurs pourront se connecter, rechercher des restaurants, consulter leurs détails, menus, horaires, donner des avis, écrire et répondre à des commentaires, et obtenir des itinéraires.**

## 2. Fonctionnalités principales

### 2.1. Carte interactive

- Afficher tous les restaurants de Lyon sous forme de marqueurs sur une carte.
- Les utilisateurs peuvent cliquer sur les marqueurs pour voir les détails du restaurant.

### 2.2. Recherche et filtrage

- Barre de recherche permettant de trouver un restaurant par nom.
- Filtrer les restaurants par catégorie (cuisine, prix, etc.).
- Possibilité de filtrer les restaurants en fonction de leur proximité ou des avis des utilisateurs.

### 2.3. Détails du restaurant

- Afficher les informations suivantes pour chaque restaurant :
  - Nom
  - Adresse
  - Numéro de téléphone
  - Horaires d'ouverture
  - Menu du jour
  - Photos
  - Avis des utilisateurs

### 2.4. Avis et commentaires

- Les utilisateurs peuvent donner une note sur 5 pour chaque restaurant.

- Les utilisateurs peuvent écrire des commentaires sur leur expérience.
- Les utilisateurs peuvent répondre aux commentaires existants.

## 2.5. Itinéraires

- Afficher un itinéraire depuis la position de l'utilisateur jusqu'au restaurant sélectionné.
- Options de transport : à pied, en voiture, par les transports en commun.

# 3. Technologies

## 3.1. Frontend

- Framework JavaScript : Utilisation de frameworks comme React.js ou Vue.js pour une interface utilisateur réactive.
- Carte interactive : Utilisation de Leaflet (licence BSD) pour afficher la carte et les marqueurs.

## 3.2. Backend

- Langage serveur : Utilisation de PHP pour la gestion des données et des utilisateurs.
- Framework PHP : Utilisation de Symfony (licence MIT) pour structurer l'application.

## 3.3. Base de données

- Utilisation de MySQL (licence GPL) ou PostgreSQL (licence PostgreSQL) pour stocker les informations des restaurants et les avis des utilisateurs.

## 3.4. Authentification

- Utilisation de OAuth2 pour l'authentification des utilisateurs (bibliothèques comme OAuth2-server-php sous licence MIT).

## 3.5. API de cartographie

- Utilisation de OpenStreetMap pour les données de cartographie (licence ODbL).

# 4. Sécurité

- Authentification des utilisateurs pour la soumission d'avis et de commentaires.
- Protection contre les injections SQL et les attaques XSS.

- Utilisation du HTTPS pour sécuriser les échanges de données.

## 5. Hébergement

- Choisir un hébergeur web compatible avec les technologies utilisées (par exemple, un serveur Apache ou Nginx).

## 6. Maintenance

- Mise à jour régulière des données des restaurants.
- Maintenance des serveurs et de la base de données.

## 7. Design

- Interface utilisateur intuitive et responsive.
- Design attractif et convivial.

## 8. Tests

- Tests unitaires et fonctionnels pour s'assurer que toutes les fonctionnalités fonctionnent correctement.
- Tests de performance pour garantir que le site est rapide et réactif.