

Cahier des charges:

Contexte et objectifs:

Chaque année, des milliers d'étudiants Erasmus arrivent en France et cherchent des lieux de vie (bars, musées, parcs, boîtes, bibliothèques, monuments). Il existe déjà des plateformes comme TripAdvisor ou Booking, mais elles ne sont pas adaptées à la vie étudiante ni à la dimension sociale (qui va où aujourd'hui ?).

On souhaite donc créer une application web qui répertorie les lieux de vie dans les grandes villes de France, avec une interaction sociale permettant aux utilisateurs de signaler qu'ils participent à une activité ("j'y vais aujourd'hui").

Étude de la concurrence:

Acteurs existants :

- **TripAdvisor** : très complet, mais trop généraliste, pas centré sur les étudiants.
- **Booking** : orienté hébergements, peu pertinent pour les sorties quotidiennes.
- **Meetup** : plateforme sociale, mais plus orientée événements que lieux fixes.

Différenciation de notre projet :

- Dimension **sociale et instantanée** ("qui va aujourd'hui ?").
- Ciblage **étudiants Erasmus** (profil, âge, pays d'origine, réseau social).
- Interface **simple et intuitive** inspirée de TripAdvisor mais plus communautaire.

Fonctionnalités prévues (par priorité):

Priorité 1 – MVP (fonctionnalités minimales):

- Catalogue de villes (Paris, Marseille, Lyon, Toulouse, Nice, Nantes, Montpellier, Strasbourg, Bordeaux, Lille, Rennes).
- Lieux répertoriés avec fiche descriptive (nom, type, adresse, horaires, description, photo).
- Gestion utilisateurs (inscription, connexion, mot de passe oublié).
- Interaction sociale : bouton "j'y vais aujourd'hui" avec compteur visible.

Priorité 2 – Fonctionnalités utiles

- Carte interactive (France → ville → lieux).
- Profils enrichis (photo, réseaux sociaux, ville d'origine).
- Recherche / filtres (par type de lieu).

Priorité 3 – Finalités (vision long terme)

- Gamification (badges, classements).
- Recommandations personnalisées (basées sur profil et historique).
- Intégration API externes (Google Maps, Instagram, Facebook Events).

Architecture technique :

Backend :

- **Langage** : Java
- **Framework** : Spring Boot
- **Modules** : Spring Web, Spring Data JPA, Spring Security (JWT), Lombok
- **BDD** : PostgreSQL (tables : users, cities, places, participations, badges, partners)

Frontend :

- **Langage** : JavaScript/TypeScript
- **Framework** : React.js
- **Librairies** : React Router, Axios, TailwindCSS, React-Leaflet (cartes)

Communication :

- API REST (JSON) entre backend et frontend.

Hébergement futur :

- Conteneurisation avec Docker.

Contraintes et risques

- Temps limité pour le développement.
- Nécessité de données fiables (catalogue initial des lieux).
- Gestion de la sécurité (protection des données utilisateurs).

Perspectives d'évolution

- Extension aux autres pays européens (pas seulement France).
- Intégration d'un module événements (concerts, soirées étudiantes).
- Application mobile native (React Native).
- Système de messagerie ou chat entre utilisateurs.