



**café
sans-fil**

Café sans-fil

Axel ZAREB & Southidej OUDANONH

Supervisé par Louis-Edouard LAFONTANT

IFT 3150 – Projet d'informatique

Université 
de Montréal



**café
sans-fil**

Sommaire

1. Introduction
2. Analyse et conception
3. Implémentation
4. Démonstration
5. Tests et déploiement
6. Roadmap
7. Conclusion



Problématique

- Pour la plupart des cafés, il faut se rendre sur place pour toutes infos
 - Connaître les horaires ou fermetures exceptionnelles
 - Modes de paiement acceptés
 - Prendre connaissance du menu et savoir ce qui est encore disponible
- L'information en ligne est dispersée et n'est pas fiable
 - Chaque café a un canal de communication différent
- La gestion du menu se fait manuellement
- Les commandes se font uniquement sur place

→ **Créer une plateforme commune pour les cafés**



**café
sans-fil**

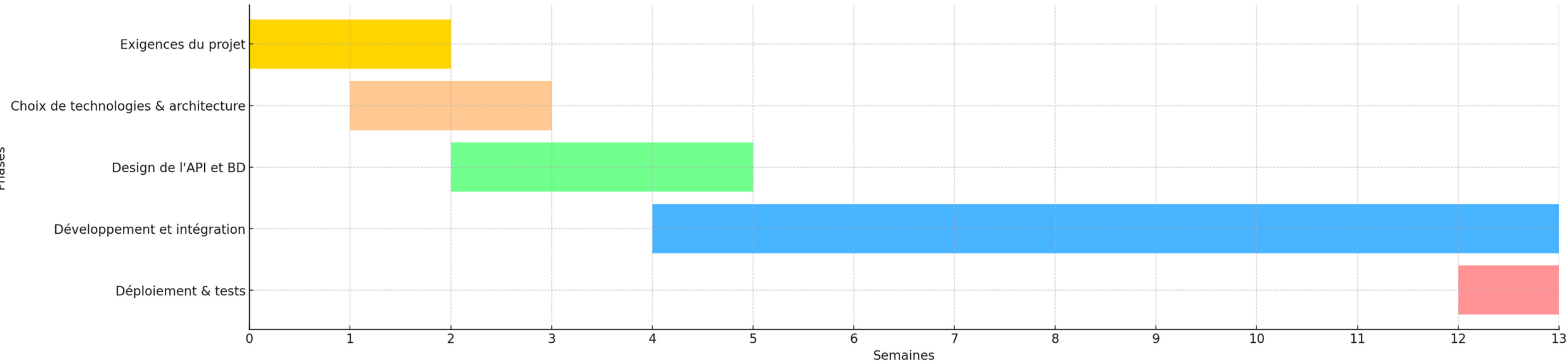
Café sans-fil

- Application web facilitant la recherche de cafés et produits, la prise de commandes et la gestion du menu et de l'inventaire.
 - Développé en open source sur GitHub: <https://github.com/ceduni/udem-cafe>
 - Version preview disponible: <https://cafesansfil.onrender.com/>

Roles dans l'application

- Étudiant
- Bénévole
- Gérant

Timeline du projet

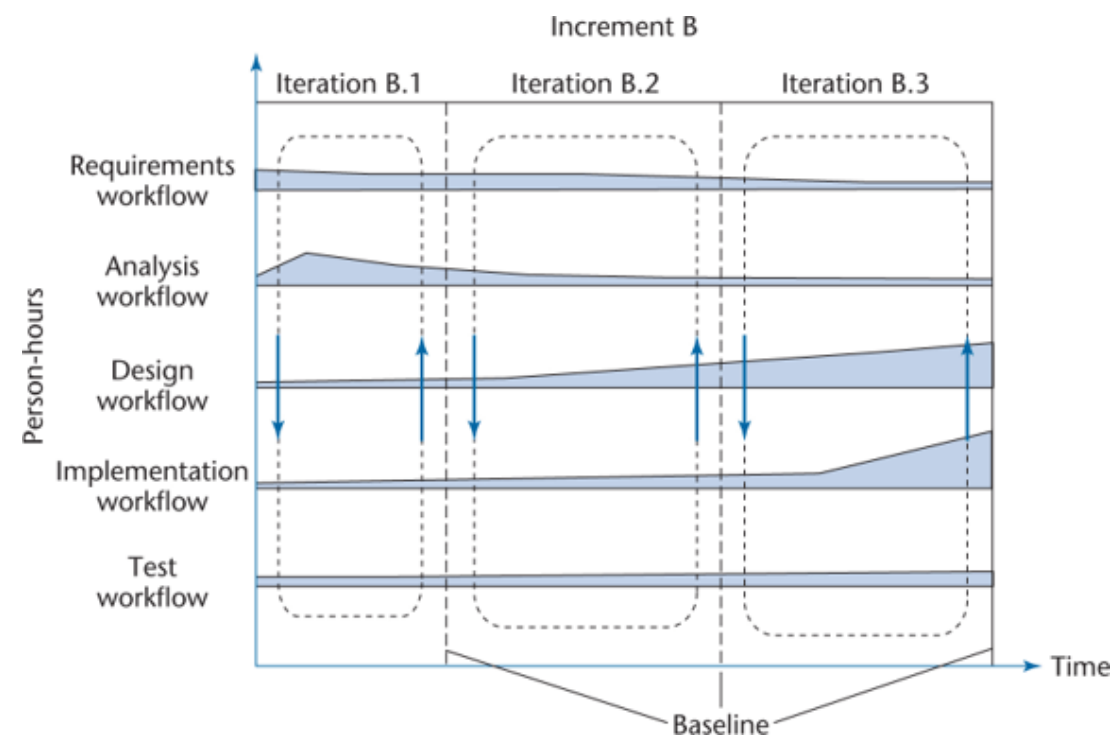




café
sans-fil

Analyse

Développement itératif-incrémental



- Chaque itération produit une version améliorée du logiciel.
- Les fonctionnalités sont développées, testées et livrées de manière incrémentale, c'est-à-dire en petites parties fonctionnelles.
- Premier prototype tôt



café
sans-fil

Analyse

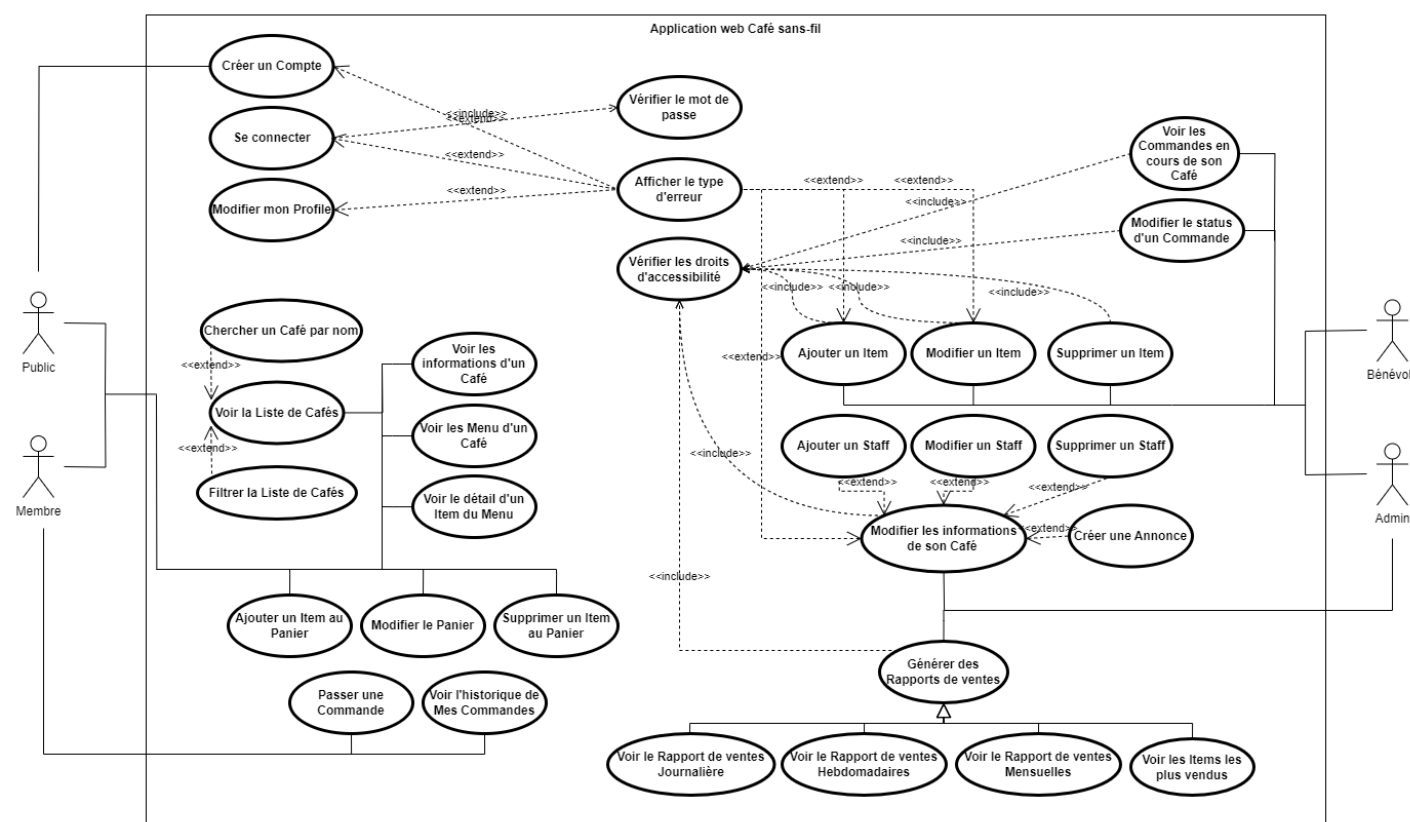
Besoins fonctionnels

Pour le grand public

- Voir la liste des cafés,
- Chercher un café ou un item
- Voir les informations d'un café
- Voir le menu d'un café

Pour les clients (membres de l'UdeM)

- Création compte, modification profil, connexion
- Passer commande et voir historique





café
sans-fil

Analyse

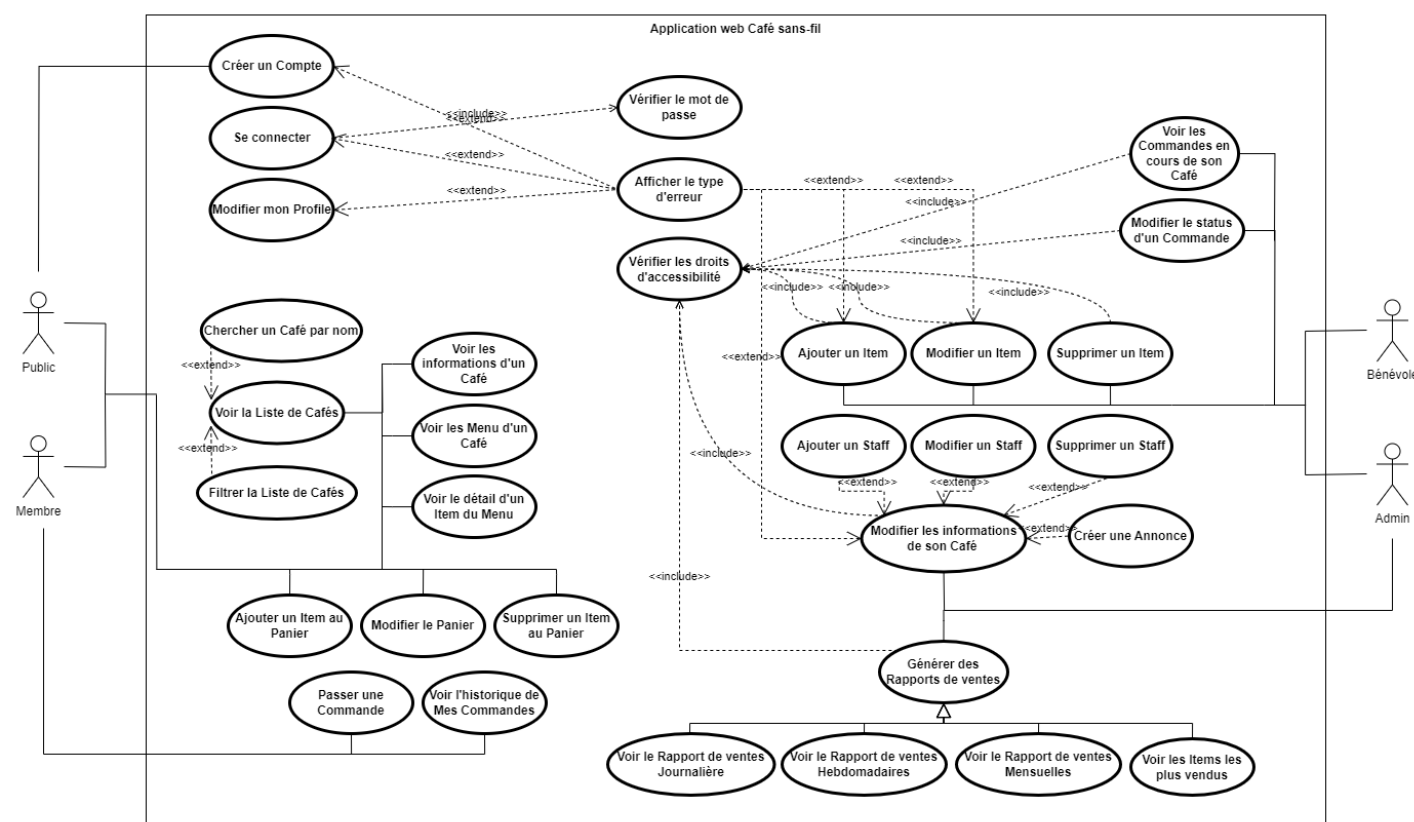
Besoins fonctionnels

Pour les bénévoles

- Afficher les commandes à traiter et modifier leur statut
- Modifier le menu du café

Pour les gérants

- Modifier les informations du café
- Modifier la liste des bénévoles du café
- Afficher rapports de vente et statistiques





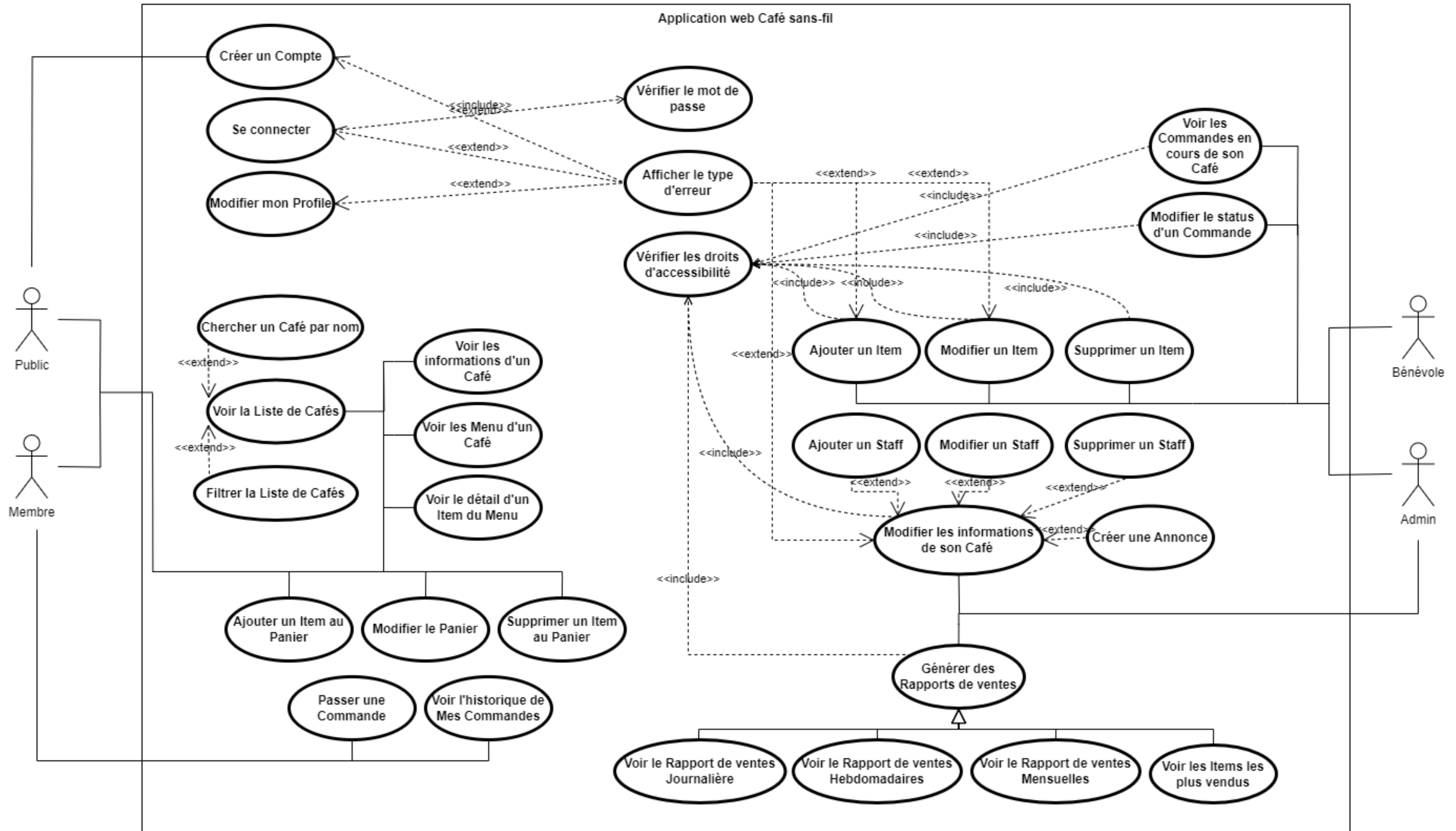
**café
sans-fil**

Analyse

Besoins non fonctionnels (essentiels)

- **Performance**
 - Application réactive, requêtes API rapides
- **Sécurité**
 - Empêcher accès non autorisé, protéger données
- **Utilisabilité**
 - Interface intuitive, plaisante, responsive
- **Maintenabilité**
 - Code bien documenté, structuré; Langages et librairies populaires

Cas d'utilisation





**café
sans-fil**

Conception



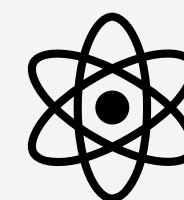
MongoDB

Base de données NoSQL orientée document, pour une gestion efficace des données



FastAPI

Framework Python facilitant le développement d'API de style REST



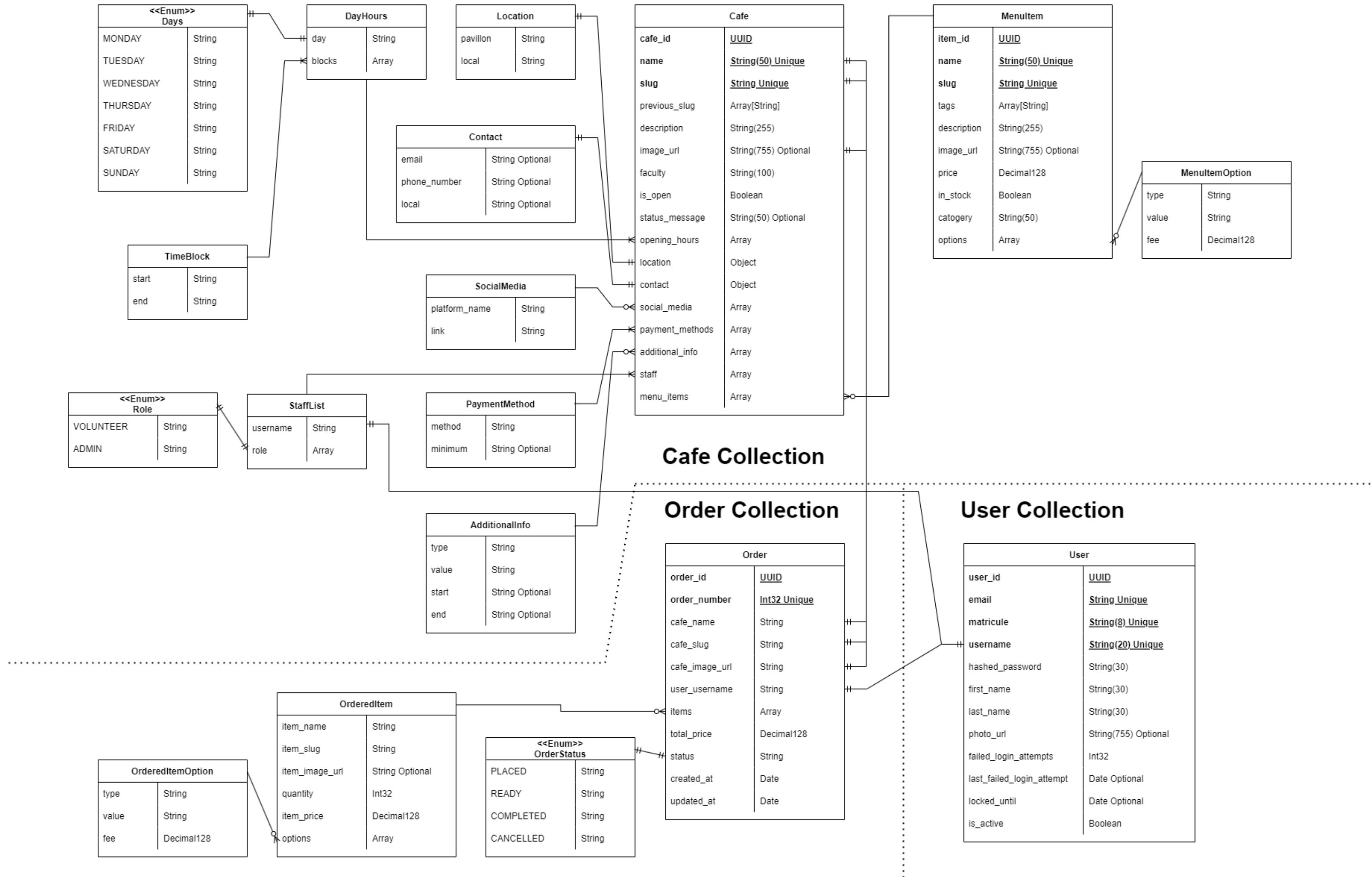
React

Librairie JavaScript facilitant le développement de single-page application (SPA)



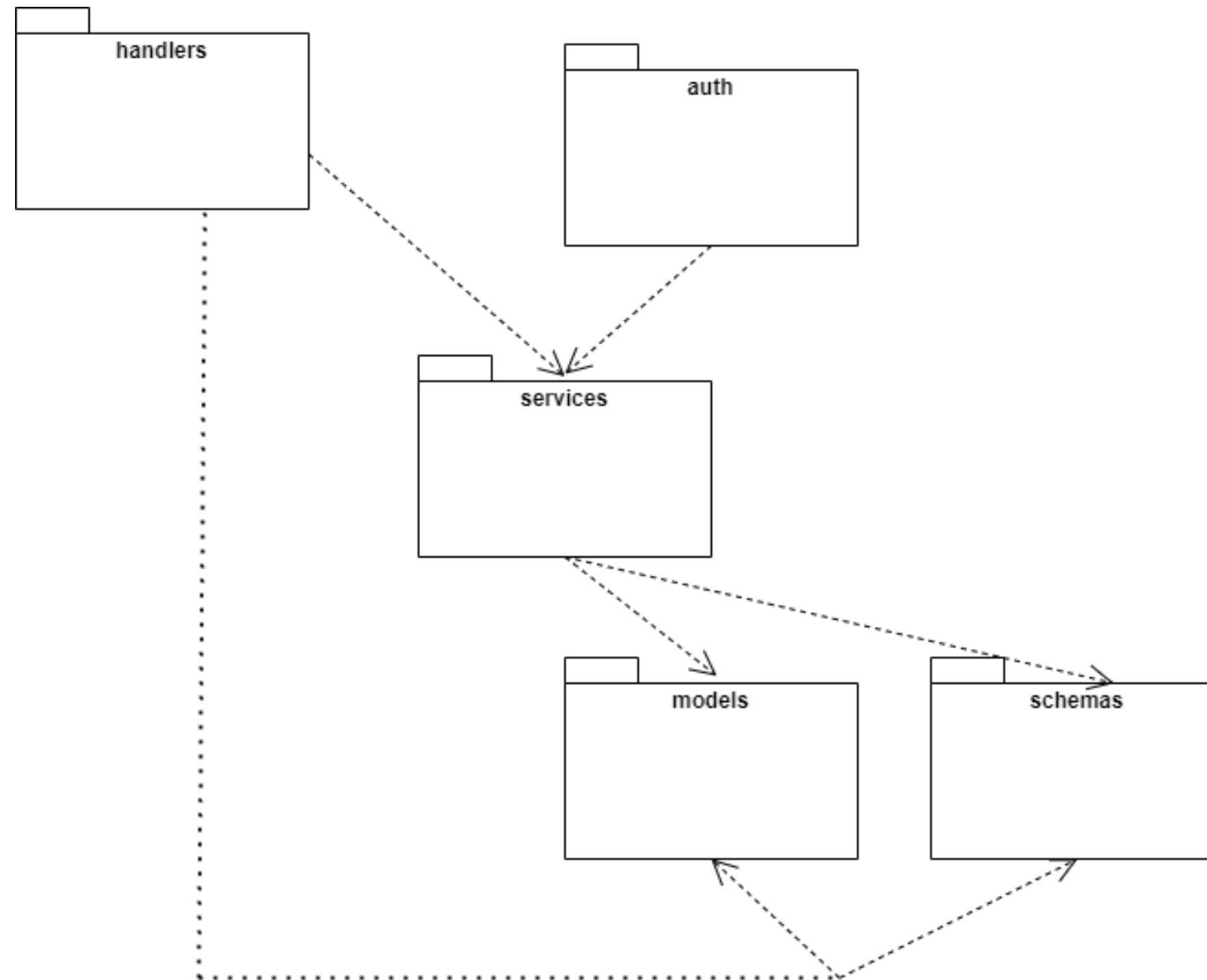
FARM Stack

Conception BD (diagramme entité-relation)



Conception API

Diagramme de paquets UML





**café
sans-fil**

Implémentation

Front-end

Utilisation de **React**



Réactivité

L'application paraît rapide, les modifications sont immédiates sans rechargement de la page contrairement à PHP.



Modularité

Le code est séparé en composants bien distincts, reprise du code simple. Chaque composant a sa logique. On sait où faire un changement



Écosystème complet

React étant un des standards du marché, beaucoup de librairies simplifient le développement de fonctions complexes



**café
sans-fil**

Implémentation

Front-end

Utilisation de **TailwindCSS**



Open source

TailwindCSS et globalement toutes les librairies externes utilisées sont open source



Collaboration facile

Pour faciliter la reprise du code et la collaboration. Facile à prendre en main, répandu, maintenance simple, une modification ne cassera rien



Léger et customisable

Contrairement à Bootstrap, seul le code utilisé est ajouté au projet lors du build. Les composants de base ne sont pas prédéfinis mais customisables

Démonstration

1. Vue public (utilisateur non connecté)

- Tentative de passage de commande

2. Création de compte

- Page de profil

3. Vue étudiant (utilisateur de base)

- Passage de commande

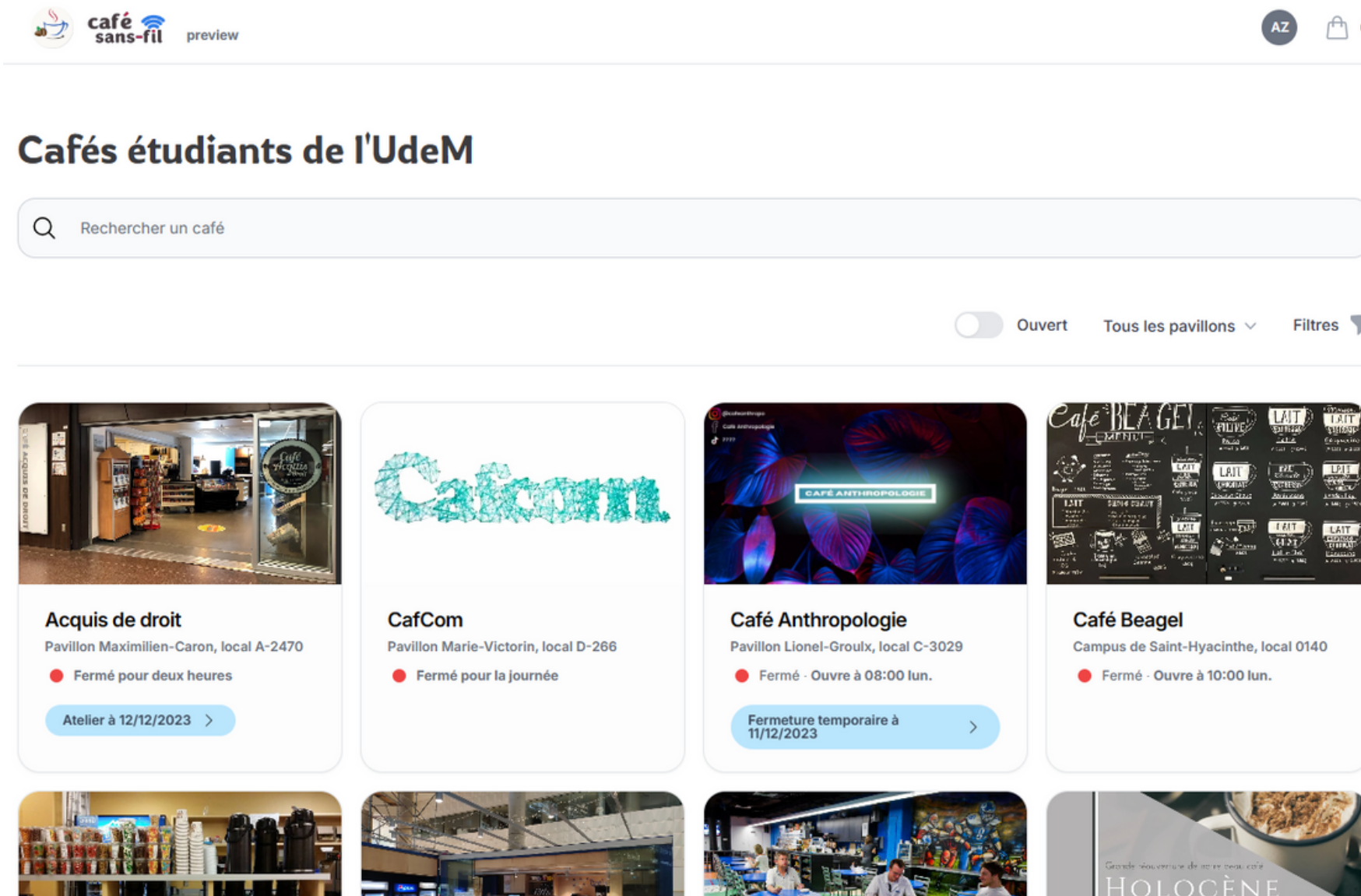
4. Vue gérant

- Affichage de la commande passée
- Modification du café / menu



café
sans-fil

<https://cafesansfil.onrender.com>





**café
sans-fil**

Tests

Tests fonctionnels

Utilisation de Postman
pour tester les routes

(voir démo)

Tests unitaires

Création de tests
unitaires pour tester
l'API

(voir démo)

Autres scripts

Création d'un script
Python permettant de
générer des données
aléatoires de test

Tests utilisateur

Feedback récolté
auprès de gérants de
cafés et de nos
proches sur
l'interface et les
fonctionnalités



**café
sans-fil**

Déploiement



Solution cloud pour compiler et déployer facilement des applications web.



Utiliser pour récolter du feedback

- Déploiement lors du push sur branche [preview](#)



**café
sans-fil**

Roadmap

Vers la version 1.0

- Améliorer l'infrastructure de l'app web
- Tests d'utilisabilité et d'acceptation
- Améliorer la performance de l'API

Améliorations futures

- Actualisation des commandes en temps réel
- Canal de communication cafés-étudiants
- Gestion des événements
- Paiement en ligne
- Support du multilingue
- Téléverser et modifier des médias (photos)



café
sans-fil

Conclusion

- ✓ Tous les cas d'utilisation identifiés ont été implémentés
- ✓ Base de données fonctionnelle
- ✓ API fonctionnelle, testée et documentée
- ✓ Application web implémentée avec UI agréable (à valider)
- ✓ Beaucoup appris sur la gestion de projet et les technologies utilisées



café
sans-fil

Merci d'avoir suivi!

Des questions?