

Documentation Technique – Application EcoRide 2025

1. Réflexions initiales technologiques

EcoRide-2025 est une plateforme de covoiturage écologique en ligne, développée dans le but de faciliter le partage de trajets avec une valorisation des véhicules électriques. Le projet utilise Symfony (PHP) pour le back-end, PostgreSQL comme base de données relationnelle principale, et MongoDB Atlas pour la journalisation (logs). Le front-end est conçu avec HTML5, CSS3, JavaScript et Twig. L'application suit une architecture MVC et respecte les bonnes pratiques de modularité et de sécurité.

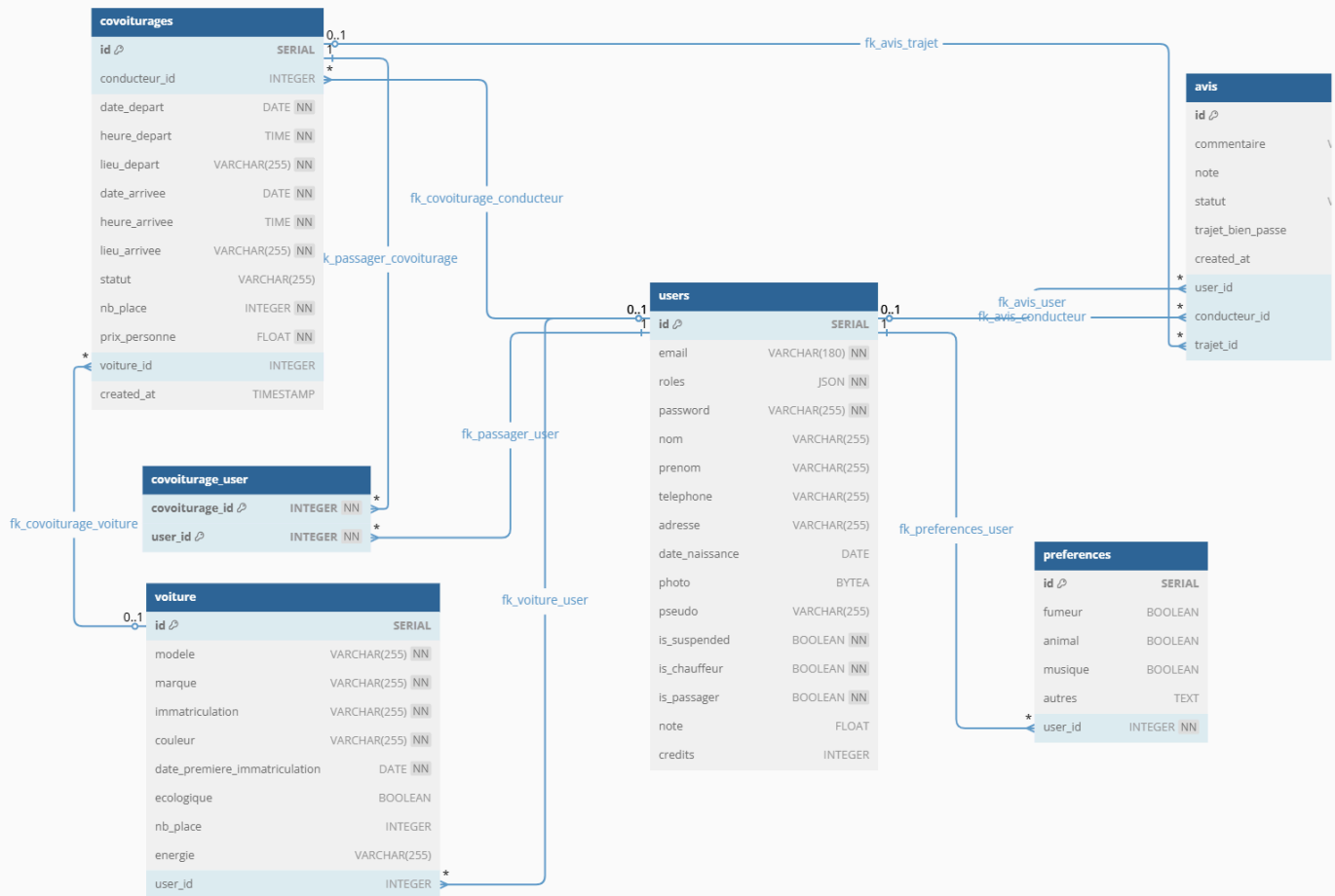
2. Configuration de l'environnement de travail

- Système d'exploitation : Windows 10
- Éditeur de code : Visual Studio Code
- Langages : PHP, HTML/CSS, JavaScript
- Framework : Symfony
- Base de données : PostgreSQL, MongoDB
- Outils : Docker, Composer, Git, Heroku
- Dépôt GitHub : <https://github.com/jo1208/EcoRide>

3. Modèle conceptuel de données (Diagramme de classe)

Le modèle conceptuel repose sur les entités suivantes :

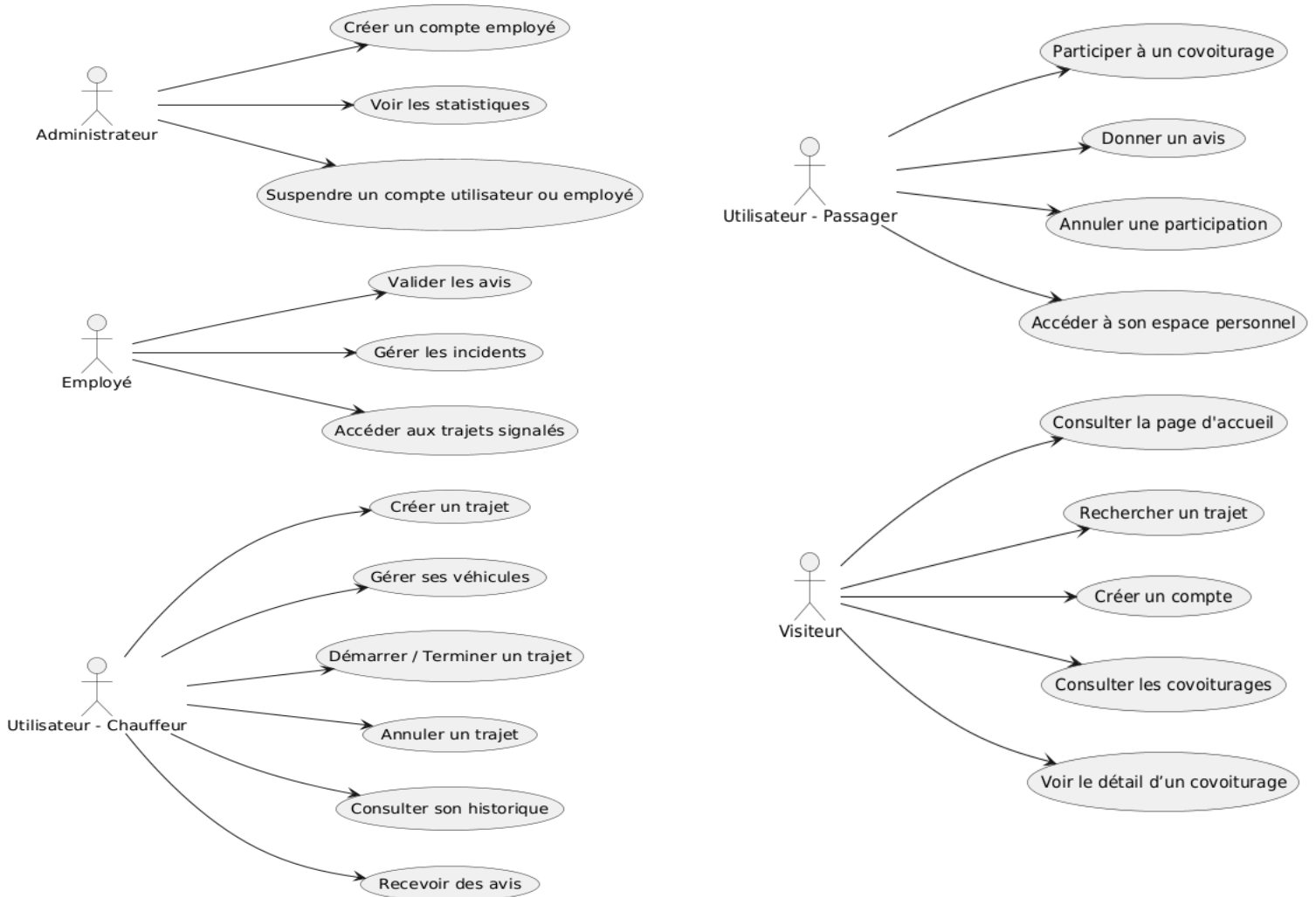
- users (utilisateurs avec rôles, crédits, etc.)
- voiture (véhicules des chauffeurs)
- covoiturages (trajets proposés)
- covoiturage_user (association entre passagers et trajets)
- avis (commentaires et notes)
- preferences (préférences de voyage des utilisateurs)



4. Diagrammes UML

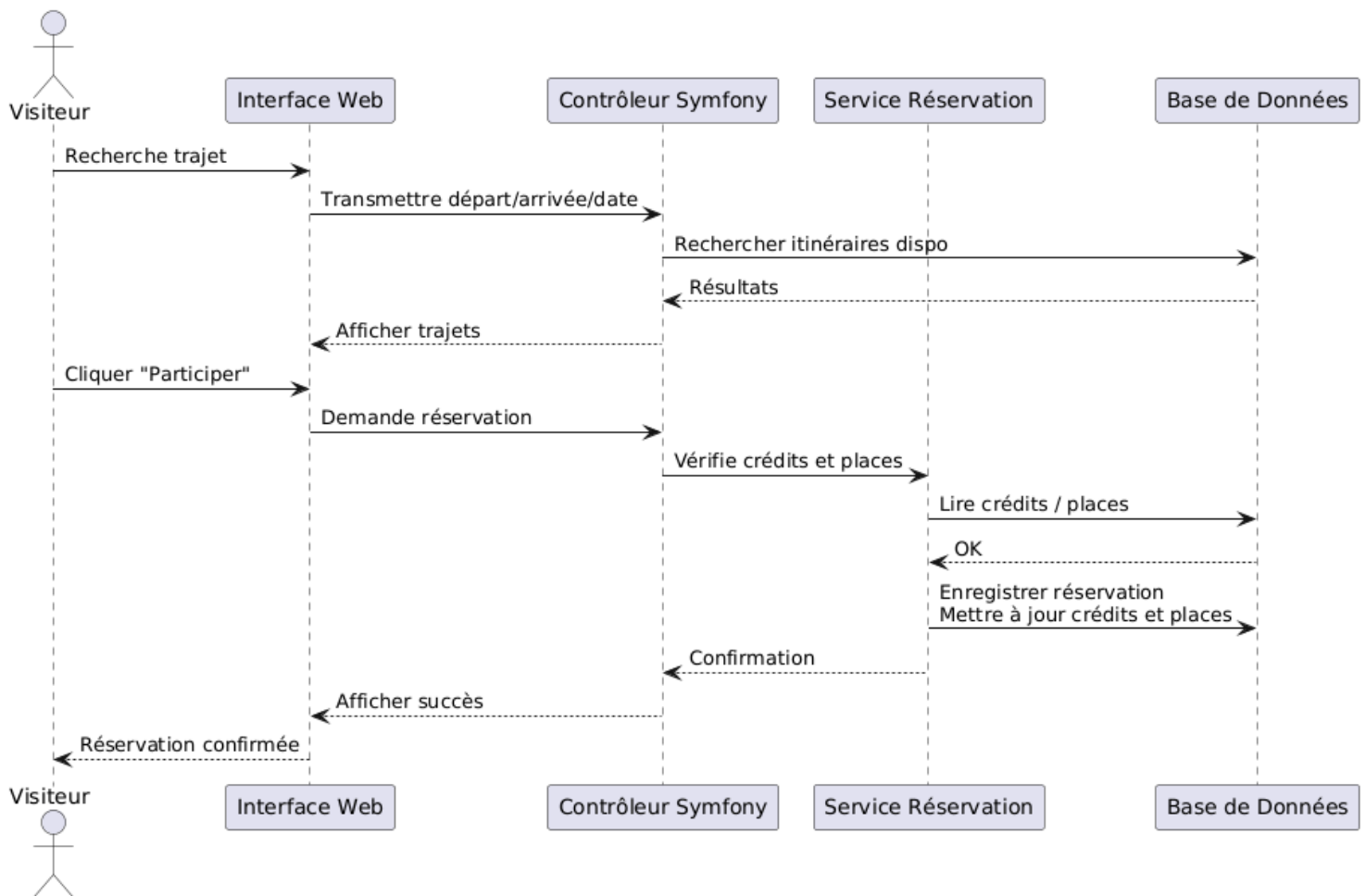
4.1 Diagramme de cas d'utilisation

Ce diagramme présente les différentes interactions possibles selon les rôles définis (Visiteur, Utilisateur, Employé, Administrateur).



4.2 Diagramme de séquence

Ce diagramme illustre le scénario de participation à un covoiturage par un utilisateur.



5. Documentation du déploiement

Le déploiement a été réalisé sur Heroku, en suivant les étapes suivantes :

1. Préparation du projet Symfony et des dépendances avec Composer.
2. Utilisation de Git pour la gestion de version avec une structure de branches propre (main, dev, features).
3. Création de l'application Heroku et ajout des buildpacks nécessaires (PHP).
4. Connexion à une base PostgreSQL.
5. Ajout des variables d'environnement nécessaires (.env).
6. Déploiement via `git push heroku main`.
7. Configuration de MongoDB Atlas pour la journalisation des logs.
8. Tests en ligne avec plusieurs rôles utilisateur.

Lien vers l'application déployée : <https://ecoride-2025-236ce68e6bd7.herokuapp.com>