Documentation Technique

–

Application EcoRide

Auteur : Emmanuelle CADILHAC
Client : EcoRide
Date de publication : 03/05/2025
Version V1.0

Table des matières

Auteur : Emmanuelle CADILHAC
Client : EcoRide
Date de publication : 03/05/2025
Version V1.0

I. Réflexions initiales technologiques

Pour concevoir EcoRide, nous avons cherché un équilibre entre robustesse, rapidité de développement et maintenabilité.

Le choix du PHP 8 avec PDO pour la couche back-end garantit une gestion sécurisée et performante de la base de données relationnelle (MySQL).

Côté front-end, l'utilisation d'HTML5, de CSS3 et de JavaScript natif associé à Bootstrap 5 permet d'obtenir une interface responsive, légère et conforme aux standards du web.

MongoDB a été retenu pour la partie NoSQL afin de stocker les réclamations de façon flexible, sans contraindre la structure des documents.

Cette architecture « mixte » profite de la solidité des bases relationnelles pour les données structurées et de l'agilité des bases documentaires pour les contenus libres comme les réclamations.

II. Configuration de l'environnement de travail

Le développement s'effectue dans Visual Studio Code, complété par les extensions PHP Intelephense et les linters HTML/CSS.

Le serveur local repose sur XAMPP, intégrant PHP 8 et MySQL, tandis que MongoDB est installé en tant que Community Edition.

Les fonctionnalités back-end sont vérifiées à l'aide de PHPUnit pour les tests unitaires et d'intégration.

Le contrôle de version est assuré par Git, avec un hébergement du code sur GitHub.

Pour la gestion de projet et le suivi des tâches, nous utilisons Notion en mode Kanban.

En production, tant le front-end que le back-end sont déployés sur des dynos Heroku.

Cette configuration assure un hébergement entièrement managé, la mise à l'échelle automatique des instances et la gestion transparente des certificats SSL.

III. Modèle conceptuel de données (MCD)

Le MCD d'EcoRide s'articule autour de cinq entités principales : utilisateur, voiture, trajet, réservation et avis.

Un utilisateur peut jouer simultanément le rôle de passager et de chauffeur.

Chaque trajet est lié à un chauffeur (utilisateur) et à une voiture.

Les Réservations enregistrent la relation entre passager et trajet, avec un état de validation et un coût en crédits.

Les réclamations, stockées dans MongoDB pour plus de souplesse, contiennent un commentaire et les métadonnées associées (identifiants utilisateur, date, etc.).

Le diagramme de classe complet, exporté depuis draw.io, est joint en annexe pour illustrer toutes les cardinalités et contraintes.

IV. Diagrammes d'utilisation et de séquence

Le diagramme de cas d'utilisation présente les interactions entre les acteurs (visiteur, utilisateur, chauffeur, employé, administrateur) et les cas d'usage clés : recherche de trajet, réservation, publication de trajet, gestion de compte et modération des avis.

Le diagramme de séquence décrit, pas à pas, le processus de réservation : le passager initie la requête, le serveur vérifie le solde de crédits, met à jour la capacité du trajet et notifie le chauffeur.

Ces deux diagrammes sont disponibles en annexe.

V. Documentation de déploiement

Le déploiement s'appuie sur un Dockerfile multi-stage pour construire une image PHP-FPM optimisée, garantissant la portabilité entre les environnements.

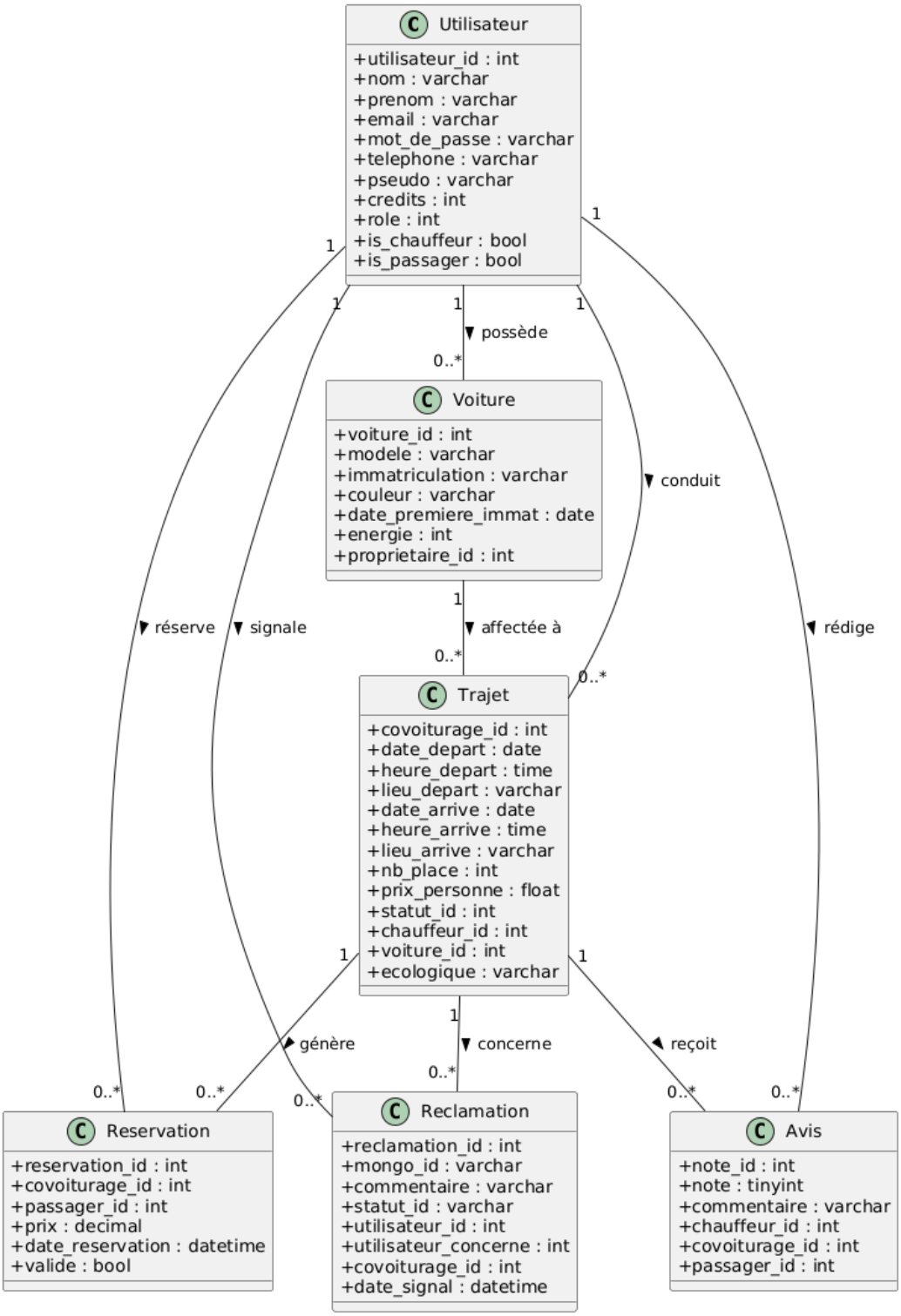
Après avoir initialisé et poussé le dépôt sur GitHub, les variables d'environnement (.env) configurent les accès à la base de données MySQL et aux services externes, ainsi que les clés secrètes pour la gestion des tokens CSRF.

Le front-end et le back-end sont tous deux déployés sur des dynos Heroku via la Heroku CLI intégrée au pipeline GitHub Actions, qui exécute composer install, phpunit --coverage puis pousse automatiquement l'image Docker sur Heroku.

Après chaque déploiement, un jeu de tests automatisés valide les principaux workflows : connexion, recherche, réservation, publication et modération.

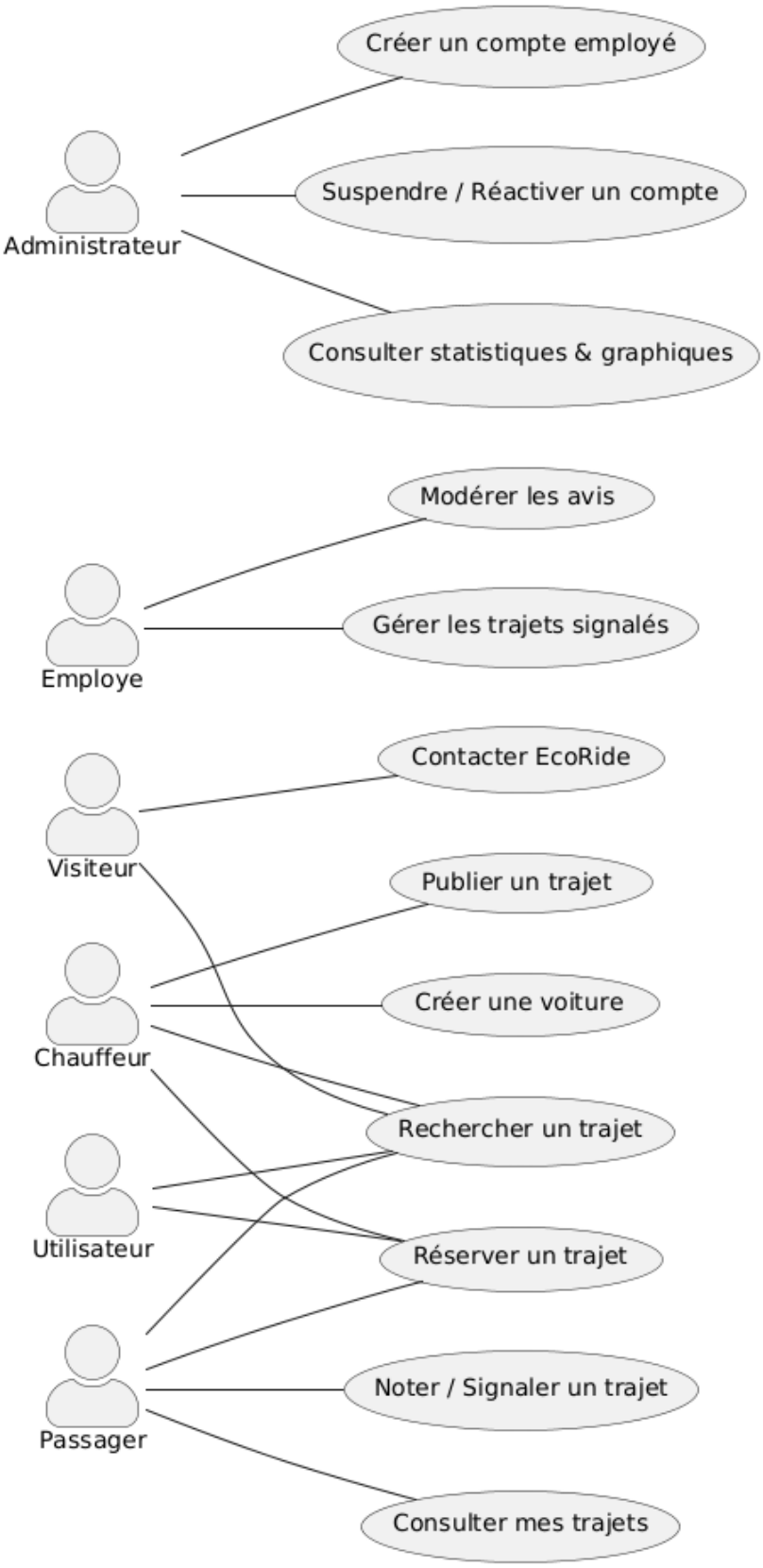
VI. Annexes :

1. Diagramme de classe (Modèle Conceptuel de Données)



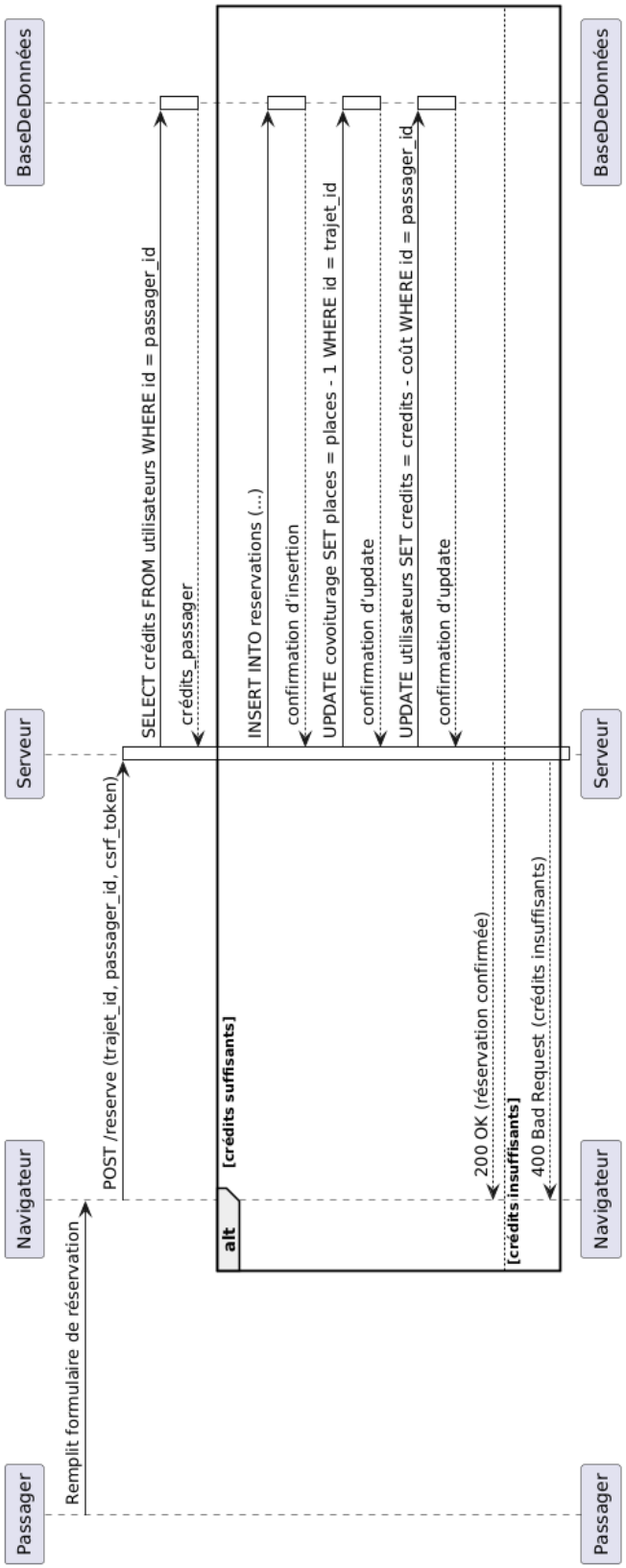
Auteur : Emmanuelle CADILHAC
Client : EcoRide
Date de publication : 03/05/2025
Version V1.0

2. Diagramme de cas d'utilisation



Auteur : Emmanuelle CADILHAC
Client : EcoRide
Date de publication : 03/05/2025
Version V1.0

3. Diagramme de séquence réservation



Auteur : Emmanuelle CADILHAC
Client : EcoRide
Date de publication : 03/05/2025
Version V1.0