

Table des matières

1 – Réflexions technologiques initiales	2
1.1. Stack technologique retenu.....	2
1.2. Suggestions d'architecture	2
1.2.1. Structure MVC légère	2
1.2.2. Sécurité.....	2
1.2.3. Routing propre.....	2
1.3. Statistiques utiles avec Chart.js	2
1.4. Bonnes pratiques de développement	2
2 – Configuration de l'environnement de travail.....	3
2.1. Outils et logiciels.....	3
2.2. Organisation des dossiers.....	3
2.3. Séparation des environnements.....	4
3 – Modèle de données.....	4
4 – Diagrammes d'utilisation et de séquence	5
5 – Architecture technique.....	6
5.1. Architecture générale (MVC étendu)	6
5.2. Organisation des rôles et accès	6
5.3. Modules fonctionnels	6
6 – Wireframes	7
6.1. Détails	8
7 – Sécurité	9
8 – Déploiement	9
8.1. Prérequis :.....	9
8.1. Configuration packages :	9
8.1. Accès PROD :.....	12
9 – Conclusion	13

1 – Réflexions technologiques initiales

Voici mon cheminement :

1.1. Stack technologique retenu

Technologie	Usage principal	Recommandations clés
PHP 8.x (PDO OO)	Backend, logique métier, accès BDD	Utiliser des classes DAO, validation côté serveur, gestion des exceptions
HTML5 / CSS3 (Bootstrap)	Structure et design responsive	Personnaliser Bootstrap pour une identité visuelle forte
JavaScript	Dynamisme client, interactions	Modulariser avec ES6, utiliser fetch API pour les appels AJAX
MySQL / MariaDB	Stockage des données	Indexation, relations bien définies, procédures stockées si besoin
Chart.js	Visualisation des statistiques	Afficher les économies de CO ₂ , trajets partagés, etc.
SVG icon library	Icônes légères et scalables	Optimiser l'accessibilité et le contraste
XAMPP (Windows)	Environnement local de développement	Séparer les environnements dev/test/prod à terme

1.2. Suggestions d'architecture

1.2.1. Structure MVC légère

- **Modèle / Vue / Contrôleur**
- Facilite la maintenance et l'évolution du projet

1.2.2. Sécurité

- Validation côté serveur + client
- Protection CSRF/XSS
- Hashage des mots de passe avec password_hash()

1.2.3. Routing propre

- Utiliser .htaccess pour des URLs lisibles.

1.3. Statistiques utiles avec Chart.js

- Émissions de CO₂ évitées par utilisateur
- Nombre de trajets partagés par mois
- Répartition des trajets par région
- Taux de remplissage moyen des véhicules

1.4. Bonnes pratiques de développement

- Versionner avec Git (GitHub ou GitLab)

- Commenter ton code intelligemment
- Créer une base de données de test
- Utiliser des sessions PHP pour la gestion utilisateur
- Implémenter des logs d'erreurs pour le débogage

2 – Configuration de l'environnement de travail

Pour garantir un développement fluide, sécurisé et collaboratif, l'environnement de travail du projet repose sur les éléments suivants :

2.1. Outils et logiciels

Outil / Logiciel	Usage principal	Remarques clés
XAMPP (Windows)	Serveur local (Apache + MySQL)	Permet de simuler l'environnement de production
VS Code	Éditeur de code	Création d'un projet contenant toutes les ressources
Git + GitHub/GitLab	Versionning du code	Branches par fonctionnalité, pull requests
MySQL Workbench	Gestion de la base de données	Visualisation des schémas, requêtes SQL
Navigateur (Chrome/Firefox)	Tests frontend et responsive design	Utiliser les outils développeur intégrés

2.2. Organisation des dossiers

```

Projet_EcoRide/
├── backend/
│   ├── config/                      # Connexion à la base de données, config SQL
│   ├── controllers/                # Logique métier (gestion utilisateurs, etc.)
│   └── logs/                       # Fichiers de logs (accès, erreurs, debug)
└── frontend/
    ├── pages/                      # Pages accessibles (index.php, etc.)
    │   ├── assets/                  # Ressources statiques
    │   │   ├── css/                 # Feuilles de style CSS
    │   │   ├── img/                 # Images métier (illustrations, icônes)
    │   │   └── js/                  # Scripts JavaScript (interactions, formulaires)
    │   └── includes/                # Templates réutilisables (navbar/header/footer)
    └── vendors/                    # Dépendances installées via Composer
        ├── composer/                # Fichiers liés au gestionnaire Composer
        ├── graham-campbell/
        │   ├── phoption/              # Package utilitaire (ex: configuration, helpers)
        │   └── symfony/               # Composants Symfony (dotenv, routing, etc.)
        ├── vlucas/
        │   └── vladimir/lucas/phpdotenv # Package vlucas/phpdotenv gérer variables d'environnement
        └── .env                         # Fichier d'environnement (variables sensibles)
    └── composer.json                # Fichier de config Composer (dépendances)
    └── composer.lock                # Fichier de verrouillage des versions packages
    └── README.md                     # Documentation du projet

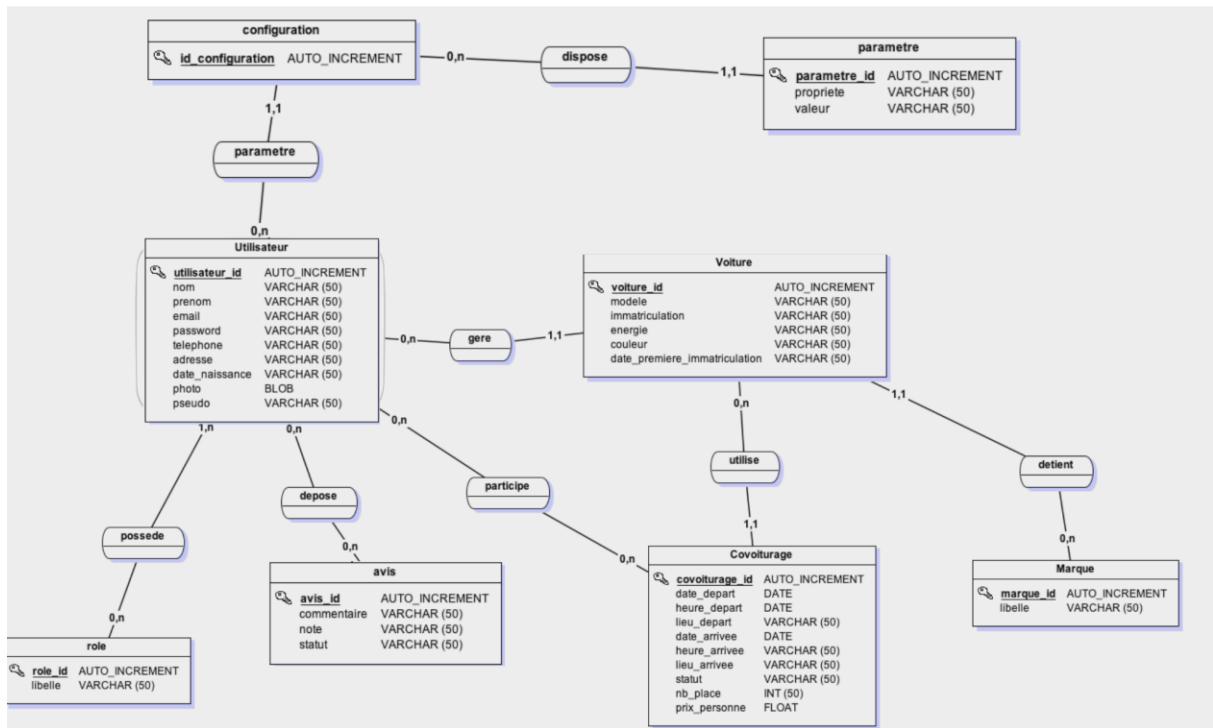
```

2.3. Séparation des environnements

- **Développement** : XAMPP local, base de données de test, logs activés
- **Pre-Prod** : Données simulées, validation des modules
- **Production** : Serveur distant sécurisé, base de données réelle, HTTPS

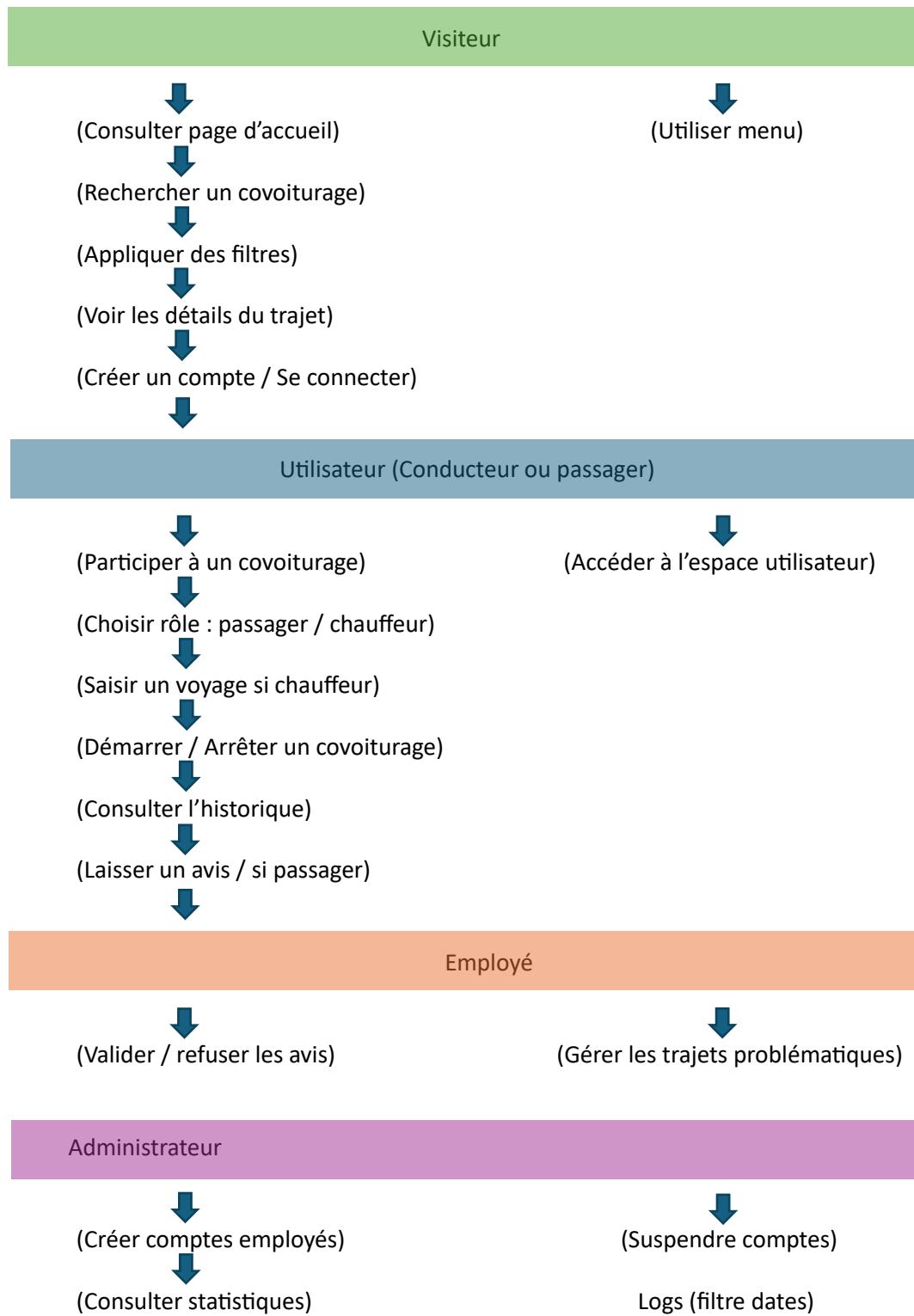
3 – Modèle de données

Le modèle est fourni



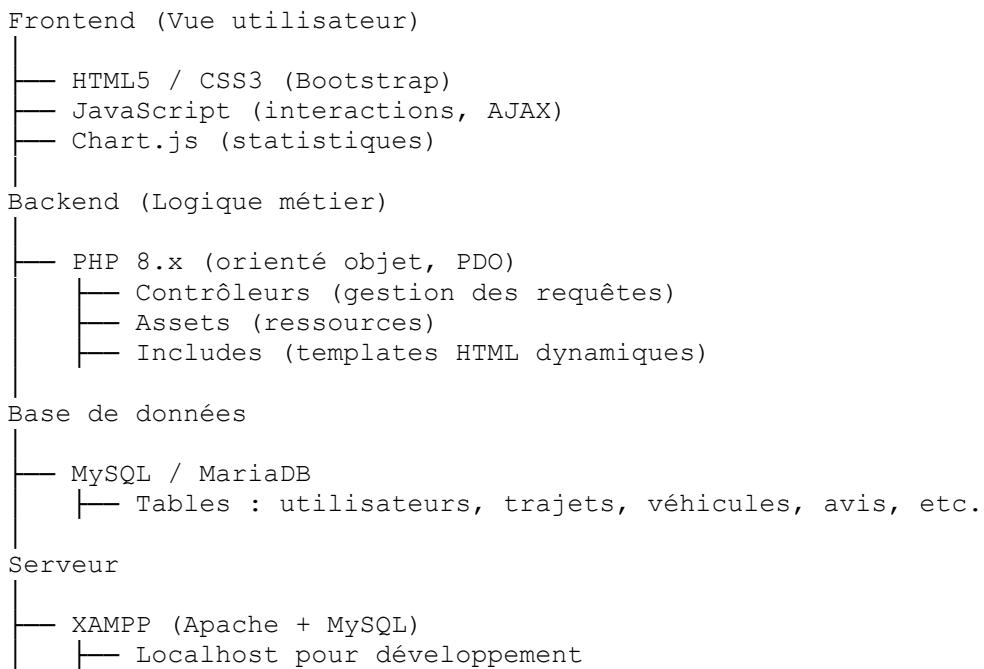
4 – Diagrammes d'utilisation et de séquence

Diagramme et séquence :



5 – Architecture technique

5.1. Architecture générale (MVC étendu)



5.2. Organisation des rôles et accès

Rôle	Accès aux fonctionnalités principales
Visiteur	Page d'accueil, recherche, inscription
Utilisateur	Réservation, espace perso, historique
Chauffeur	Création de trajets, gestion véhicule
Passager	Réservation, avis, crédits
Employé	Modération des avis, gestion incidents
Administrateur	Statistiques, gestion des comptes

5.3. Modules fonctionnels

5.3.1. Authentification

- PHP Sessions
- Vérification des rôles
- Sécurité : mot de passe hashé, validation côté serveur

5.3.2. Gestion des trajets

- Formulaire de recherche
- Filtres dynamiques (écologie, prix, durée, note)
- Vue détaillée avec préférences du conducteur

5.3.3. Statistiques

- Chart.js pour :
 - Nombre de trajets par jour
 - Crédits générés
 - Répartition écologique

5.3.4. Gestion des crédits

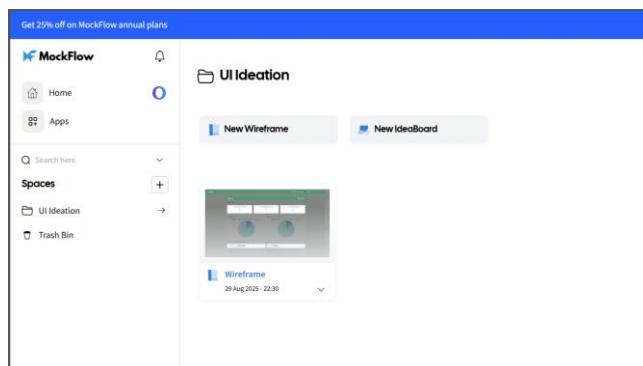
- Crédit initial à la création
- Déduction automatique à la réservation
- Mise à jour après validation du trajet

5.3.5. Modération

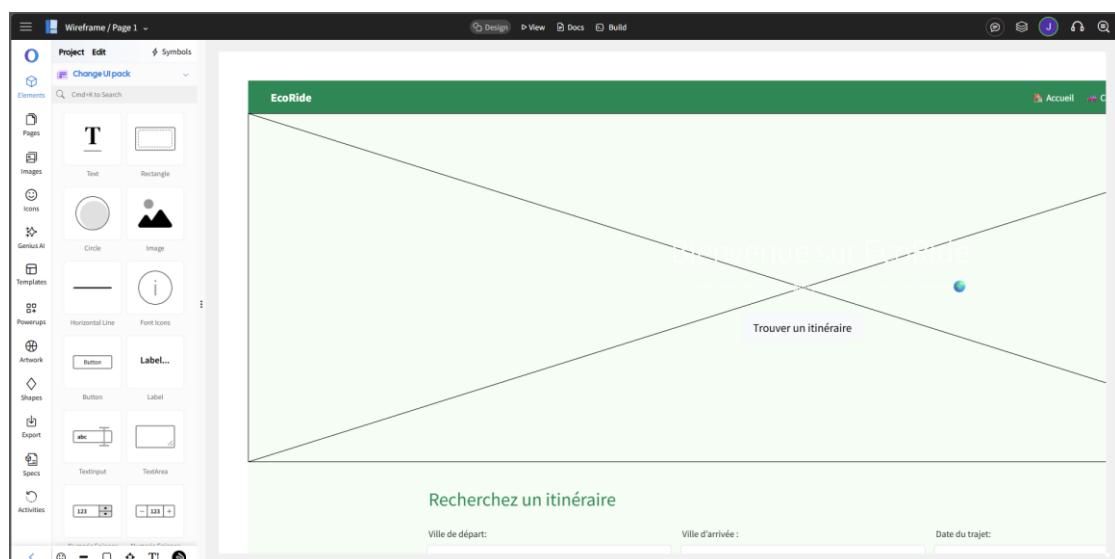
- Interface employé : avis à valider, trajets problématiques
- Interface administrateur : création d'employés, suspension de comptes

6 – Wireframes

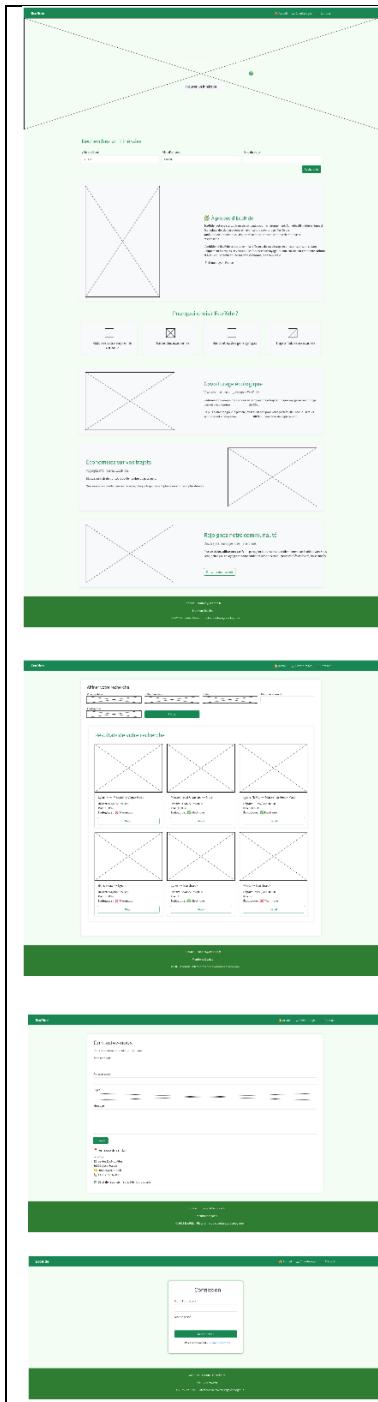
Utilisation de l'outil [MockFlow](#)



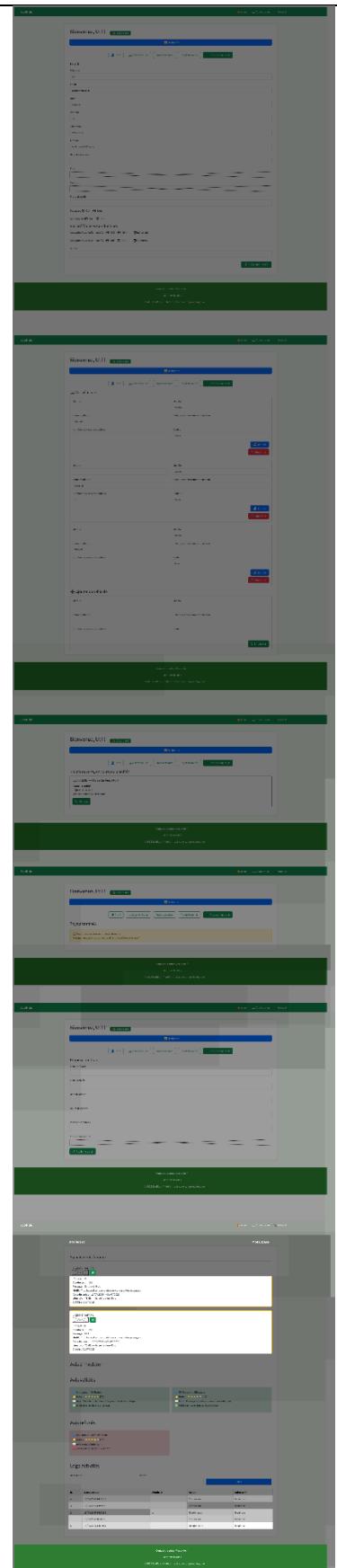
Pages design



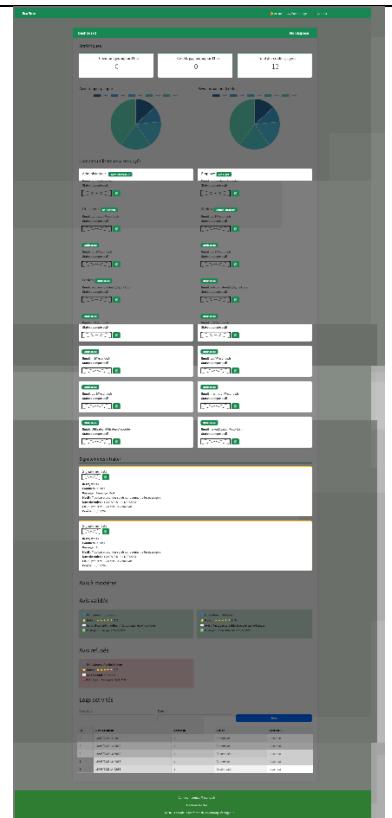
6.1. Détails



Pages du menu



Pages utilisateur / modérateur



Pages Administrateur

7 – Sécurité

Le choix de la solution PHP + MySQL/MariaDB et Bootstrap, on permet de mettre en place une couche de sécurité intégré. Voici les différents éléments intégrés

- Validation des entrées (serveur + client)
- Protection CSRF/XSS
- Requêtes préparées avec PDO
- Gestion des erreurs et logs
- Structure MVC pour séparation des responsabilités

8 – Déploiement

La solution de déploiement sera effectuée sur un NAS pour une question de coût personnel

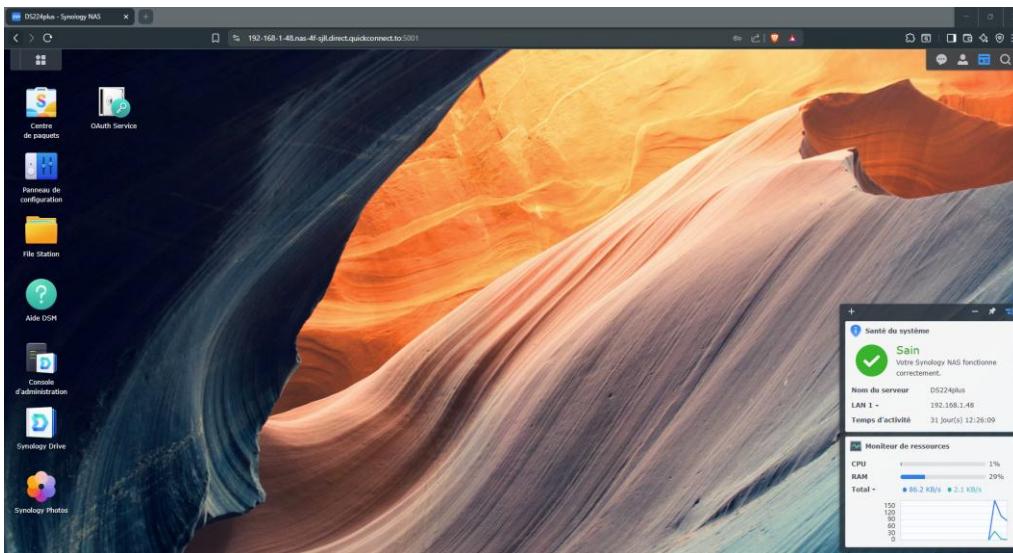
8.1. Prérequis :

Configuration NAS

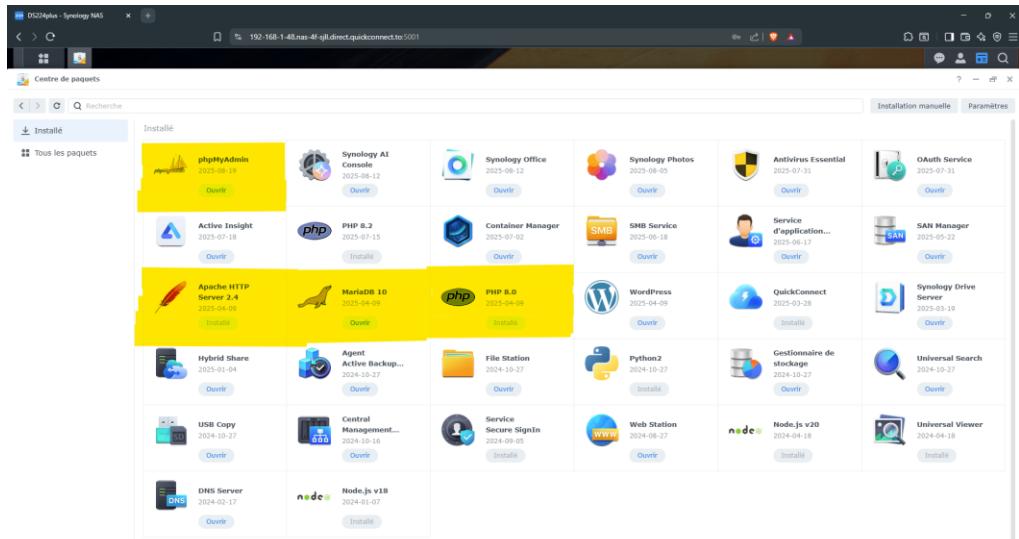
Environnement	Descriptifs
Matériel :	Synology DS224 +
System d'exploitation	DSM 7.2
Pacquages installés	Apache HTTP 2.4 PhpMyAdmin MariaDB 10 PHP 8.x
Url externe	https://Quickconnect.to/nas-4f-sjll

8.1. Configuration packages :

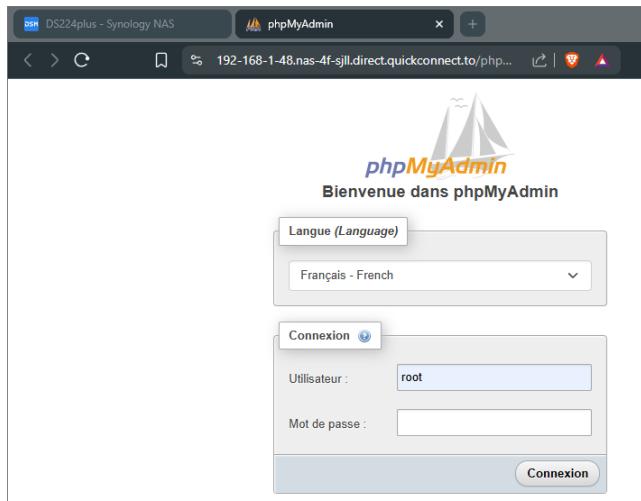
Compte administrateur NAS



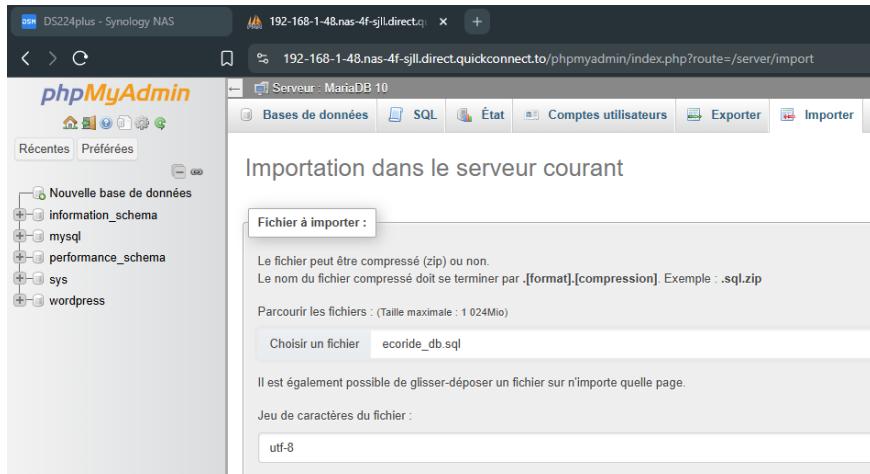
Contrôle des packages installés



Connection au compte root de phpMyAdmin



Import du fichier de création de base présent dans le dossier SQL du Github (ecoride_db.SQL)



Exécution de l'import

L'importation a réussi. 35 requêtes exécutées. (ecoride_db.sql)

MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0 0004 seconde(s))

-- Création de la base CREATE DATABASE IF NOT EXISTS ecoride_db CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci;

[Éditer en ligne] [Éditer] [Créer le code source PHP]

Error: #1046 Aucune base n'a été sélectionnée

MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0 0001 seconde(s))

USE ecoride_db;

[Éditer en ligne] [Éditer] [Créer le code source PHP]

Error: #1046 Aucune base n'a été sélectionnée

MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0 0001 seconde(s))

-- D'abord : suppressions si existant (ordre contraintes OK) SET FOREIGN_KEY_CHECKS = 0;

[Éditer en ligne] [Éditer] [Créer le code source PHP]

Error: #1046 Aucune base n'a été sélectionnée

MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0 4700 seconde(s))

DROP TABLE IF EXISTS role;

[Éditer en ligne] [Éditer] [Créer le code source PHP]

Note: #1051 Table 'ecoride_db.role' inconnue

MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0 6066 seconde(s))

DROP TABLE IF EXISTS utilisateur;

Structure de la base de données déployée

Table	Action	Lignes	Type	Interclassement	Taille	Perte
avis	Parcourir Structure Rechercher Insérer Vider Supprimer	16,0	InnoDB	utf8mb4_general_ci	16,0 kio	-
configuration	Parcourir Structure Rechercher Insérer Vider Supprimer	16,0	InnoDB	utf8mb4_general_ci	16,0 kio	-
covolurage	Parcourir Structure Rechercher Insérer Vider Supprimer	16,0	InnoDB	utf8mb4_general_ci	16,0 kio	-
covolurage_participants	Parcourir Structure Rechercher Insérer Vider Supprimer	16,0	InnoDB	utf8mb4_general_ci	16,0 kio	-
logs_activite	Parcourir Structure Rechercher Insérer Vider Supprimer	16,0	InnoDB	utf8mb4_general_ci	16,0 kio	-
marque	Parcourir Structure Rechercher Insérer Vider Supprimer	16,0	InnoDB	utf8mb4_general_ci	16,0 kio	-
parametre	Parcourir Structure Rechercher Insérer Vider Supprimer	16,0	InnoDB	utf8mb4_general_ci	16,0 kio	-
preferences_conducteur	Parcourir Structure Rechercher Insérer Vider Supprimer	16,0	InnoDB	utf8mb4_general_ci	16,0 kio	-
preferences_trajet	Parcourir Structure Rechercher Insérer Vider Supprimer	16,0	InnoDB	utf8mb4_general_ci	16,0 kio	-
role	Parcourir Structure Rechercher Insérer Vider Supprimer	16,0	InnoDB	utf8mb4_general_ci	16,0 kio	-
signalements	Parcourir Structure Rechercher Insérer Vider Supprimer	16,0	InnoDB	utf8mb4_general_ci	16,0 kio	-
status	Parcourir Structure Rechercher Insérer Vider Supprimer	3	InnoDB	utf8mb4_general_ci	16,0 kio	-
utilisateur	Parcourir Structure Rechercher Insérer Vider Supprimer	32,0	InnoDB	utf8mb4_general_ci	32,0 kio	-
voiture	Parcourir Structure Rechercher Insérer Vider Supprimer	32,0	InnoDB	utf8mb4_general_ci	32,0 kio	-
Somme		9	InnoDB	utf8mb4_general_ci	256,0 kio	0 o

Utilisation de File Station pour déposer les fichiers du github

File Station

DS224plus

- docker
- home
- homes
- web
- web_packages
 - docker
 - phpmyadmin
 - Projet_EcoRide
 - backend
 - frontend
 - SQL
 - wordpress

web_packages > Projet_EcoRide

Nom	Type	Date de modification
backend	Do...	2025-08-19 13:37:30
frontend	Do...	2025-08-19 13:37:28
SQL	Do...	2025-08-19 13:37:30

Configuration du portail web

The screenshot shows the 'Portail défini par l'utilisateur' section of the Web Station interface. It lists several entries:

Statut	Service	Nom d'hôte	Port	Alias	Lien
Normal	If4digital (project)	www.if4digital.fr	80 / 443	-	
Normal	ldp (project)	www.lame-des-pierres.fr	80 / 443	-	
Normal	WordPress	www.lame-des-pierres.frr	80 / 443	-	
Normal	ecoride	-	-	ecoride	
Normal	Serveur par défaut	-	80 / 443	-	
Normal	WordPress	-	-	wordpress	
Normal	phpMyAdmin	-	-	phpmyadmin	

Configuration du service web

The screenshot shows the 'Service par défaut' section of the Web Station interface. It lists various services and their configurations:

Type de service	Nom	Statut	Version du service/de la base de données	Description
Service par défaut	WebStation Default Server Service	Normal	--	
Site Web du paquet tiers	wordpress	Normal	PHP 8.0	
phpMyAdmin	phpMyAdmin	Normal	PHP 8.0	
Site Web en langage de script conteneurisé	Container Manager	Normal	--	
Container Manager	If4digital (project)	Normal	--	
Container Manager	ldp (project)	Normal	--	
Site Web en langage de script natif	PHP	ecoride	Normal	PHP 8.0 Projet Ecoride

8.1. Accès PROD :

Accès au site depuis l'adresse en https

<https://192-168-1-48.nas-4f-sjll.direct.quickconnect.to/ecoride/>

The screenshot shows a web browser window with the address bar containing the URL: <https://192-168-1-48.nas-4f-sjll.direct.quickconnect.to/ecoride/>. The page itself is the 'EcoRide - Covoiturage écologique' homepage. The header features the 'EcoRide' logo and navigation links for 'Accueil', 'Covoitages', 'Contact', 'Utilisateur', and 'Déconnexion'. The main content area has a green background with a central image of a car driving through a forest. A message box at the top right says 'Jérôme FONTAINE (bib2510@hotmail.com) est connecté'. Below it, the text 'Bienvenue sur EcoRide' and 'La plateforme de covoiturage pensée pour l'environnement' is displayed. A button labeled 'Trouver un itinéraire' is visible at the bottom of the message box.

9 – Conclusion

La plateforme de covoiturage a été pensée pour réduire l'impact environnemental des déplacements.

Grâce à une architecture solide basée sur PHP 8 orienté objet, une base de données relationnelle optimisée, et une interface utilisateur responsive, l'application répond aux différents besoins fonctionnels des différents profils : visiteurs, utilisateurs, employés et administrateurs.

L'approche MVC facilite la maintenance et l'évolution du projet, tandis que les modules de sécurité assurent la fiabilité des échanges et la protection des données.

L'intégration de Chart.js permet une visualisation claire des statistiques de la plateforme.

La modération des avis, le suivi des trajets sont autant de fonctionnalités qui garantissent une expérience utilisateur fluide, sécurisée et responsable.

Ce document technique constitue les fondations d'un projet ambitieux, aligné avec les enjeux écologiques et les attentes des utilisateurs.