

# Documentation de gestion de projet – EcoRide

Auteur : Seifeddine Maachaoui

Projet : EcoRide – Application web de covoiturage écoresponsable

Type : Front-end (HTML / CSS / JavaScript / localStorage)

Version : 0.1

## 1. Introduction

Je présente ici la documentation de gestion de projet d'EcoRide, une application web que j'ai conçue et développée dans le cadre de ma formation Graduate Développeur Web Full Stack. Mon objectif était de réaliser une solution de covoiturage écoresponsable, intuitive et accessible, tout en appliquant les principes du développement front-end moderne : modularité, structure MVC et gestion locale des données.

## 2. Choix technologiques et architecture

### 2.1. Langages et outils utilisés

J'ai développé l'application en utilisant exclusivement les technologies front-end suivantes : HTML5 pour la structure et la sémantique des pages, CSS3 pour la mise en forme et le responsive design, JavaScript (ES6) pour la logique métier et la gestion dynamique des interactions, LocalStorage pour simuler une base de données et stocker les utilisateurs, trajets et réservations, et PHP uniquement pour l'inclusion des éléments communs comme le header et le footer.

### 2.2. Architecture du projet (pattern MVC)

J'ai structuré mon application autour du modèle MVC (Modèle – Vue – Contrôleur), afin d'assurer une séparation claire des responsabilités. Le Model gère les données (utilisateurs, trajets, réservations) à travers des classes JavaScript. La View s'occupe de la présentation et de la mise à jour du DOM. Le Controller fait le lien entre la logique métier et l'affichage.

## 3. Organisation du développement

### 3.1. Découpage du travail

J'ai commencé par définir les fonctionnalités principales : création d'un espace utilisateur avec connexion et déconnexion, gestion du rôle (passager, chauffeur ou passager/chauffeur), ajout des informations sur le véhicule, consultation et participation à des trajets, gestion des réservations et système de notifications locales. Le développement s'est déroulé en plusieurs étapes : conception de la structure, mise en place du CSS, intégration du JavaScript et finalisation du responsive design.

### 3.2. Utilisation du localStorage

Je me suis appuyé sur le localStorage pour simuler un back-end. Chaque entité (utilisateur, trajet, réservation, notification) possède un modèle JavaScript dédié qui charge, met à jour et sauvegarde les données localement au format JSON. Cette approche reproduit une logique CRUD complète (Create, Read, Update, Delete) sans serveur.

#### **4. Gestion du code et modularité**

J'ai veillé à rendre chaque partie du code indépendante et réutilisable. Le fichier auth.js gère la session, les models centralisent les données, et les controllers assurent la logique métier. Cette modularité facilite la maintenance et l'ajout de nouvelles fonctionnalités à long terme.

#### **5. Responsive design et accessibilité**

J'ai adopté une approche mobile first en concevant d'abord l'interface pour smartphone avant de l'adapter aux tablettes et ordinateurs. Sur mobile : une seule colonne, texte compact, boutons centrés. Sur tablette : deux colonnes. Sur desktop : trois colonnes et typographie augmentée. La palette de couleurs vertes rappelle l'éco-responsabilité et améliore la lisibilité.

#### **6. Gestion du temps et de la progression**

Le projet a été réalisé en plusieurs semaines. J'ai structuré mon travail en trois grandes phases : conception (diagrammes et wireframes), développement (intégration et logique JS) et tests (ergonomie et cohérence des données). Cette méthode itérative m'a permis de valider chaque étape avant de passer à la suivante.

#### **7. Conclusion**

Ce projet m'a permis d'appliquer concrètement la logique MVC en JavaScript et de renforcer mes compétences en architecture front-end. En travaillant sans back-end, j'ai appris à structurer des données localement, à gérer des interactions complexes et à organiser mon code de façon modulaire. EcoRide constitue donc un projet complet mêlant conception, design, logique métier et gestion de données, tout en respectant les principes du développement web moderne.