

Présentation du diagramme de cas d'utilisation – Projet EcoRide

Pour répondre aux exigences du projet EcoRide, j'ai réalisé ce diagramme de cas d'utilisation en m'appuyant sur les 13 User Stories décrites dans l'énoncé. Mon objectif était de représenter de manière claire et fonctionnelle les interactions entre les différents acteurs et les fonctionnalités principales de l'application.

Les acteurs

J'ai identifié quatre acteurs principaux :

Visiteur : utilisateur non connecté, qui peut consulter les informations et rechercher des covoiturages.

Utilisateur EcoRide : une personne inscrite sur la plateforme, pouvant être passager ou chauffeur.

Employé : chargé de la modération des avis et de la gestion des incidents.

Administrateur : responsable de la gestion des comptes employés et du suivi statistique de la plateforme.

Organisation des fonctionnalités

Chaque User Story est représentée dans une boîte dédiée, avec ses sous-fonctionnalités :

US1 à US5 : concernent la navigation, la recherche et la consultation des covoiturages.

US6 à US8 : couvrent la création de compte, la participation à un covoiturage et la gestion des crédits.

US9 à US11 : sont liées à la gestion des trajets par les chauffeurs (création, historique, démarrage/arrêt).

US12 et US13 : sont réservées aux rôles internes (employé et administrateur).

J'ai également intégré des fonctions transversales comme :

La connexion/authentification,

La gestion des véhicules et des marques,

L'envoi de notifications par email,

Et le système de crédits, essentiel au fonctionnement de la plateforme.

Pourquoi ce choix de structure ?

J'ai choisi une organisation par acteur, avec des blocs fonctionnels bien séparés, pour faciliter la lecture et la compréhension du diagramme :

Cela permet de visualiser rapidement les responsabilités de chaque rôle.

Les sous-fonctionnalités dans chaque User Story montrent que j'ai bien pris en compte les détails de l'énoncé.

J'ai aussi intégré les interactions système pour montrer que certaines actions ne dépendent pas directement d'un utilisateur (ex. : envoi de mails, mise à jour des crédits).

✓ Conclusion

Ce diagramme me permet de valider l'analyse fonctionnelle du projet, en montrant que toutes les exigences du cahier des charges sont bien prises en compte. Il constitue une base solide pour le développement, en particulier pour la séparation front-end / back-end et la gestion des rôles.