**Cahier de conception– Application Web de Vente de Voitures**

**1.1 Architecture générale**

**Architecture MVC (MTV Django) :**

Model → Base de données

Template → Interface

Vue → Logique métier

Schéma :

**Utilisateur → Interface Web → Django Views → Model → Base de données**

**1.2 Diagramme de cas d’utilisation (UML)**

**Acteurs : Acheteur, Vendeur, Admin**

**Cas principaux :**

* S’inscrire / Se connecter
* Publier voiture
* Rechercher voiture
* Acheter voiture
* Gérer annonces
* Administrer plateforme

**(Représentation textuelle)**

- Acheteur ── Rechercher voiture

- Acheteur ── Acheter voiture

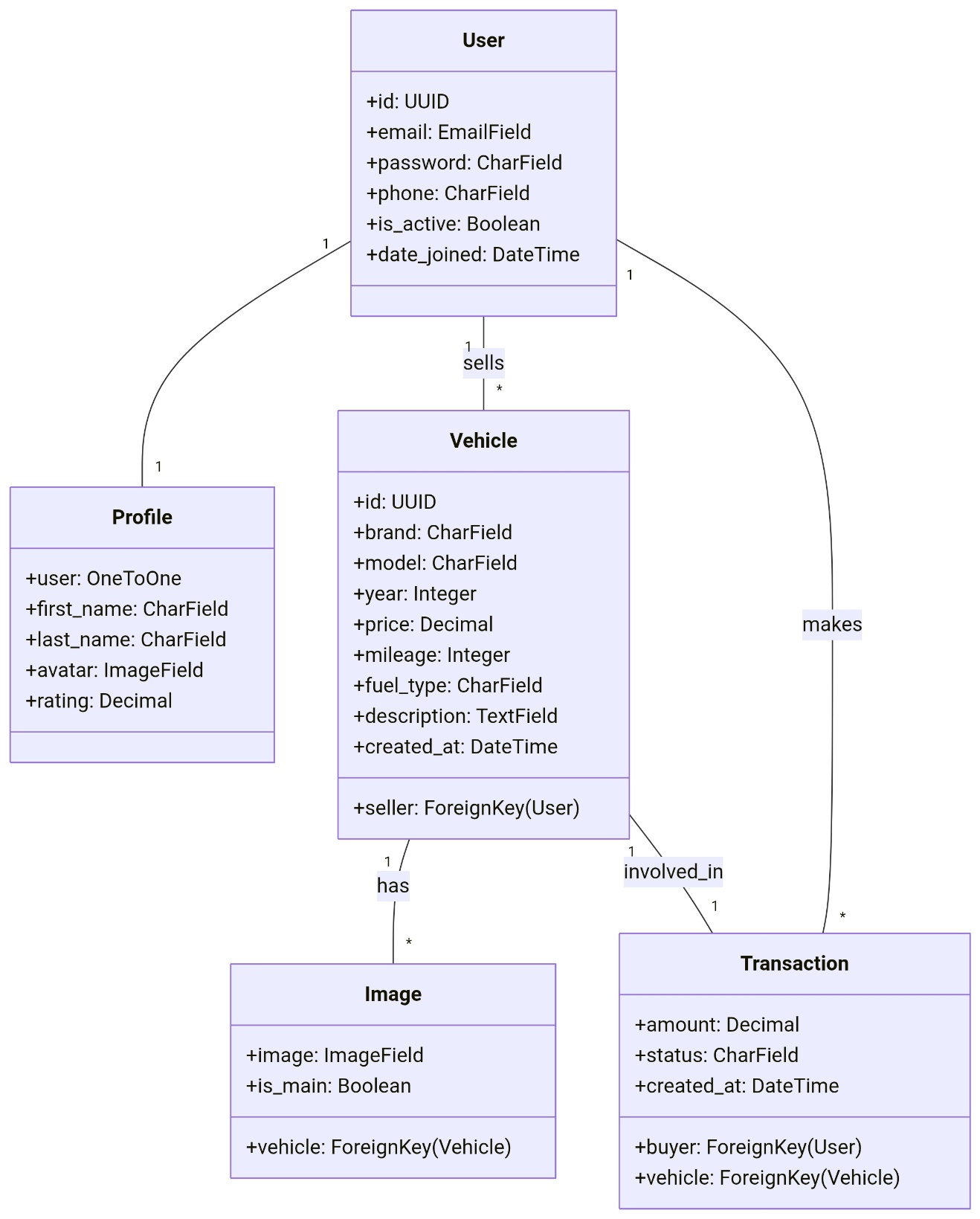
- Acheteur ── Contacter vendeur

- Vendeur ── Publier voiture

- Vendeur ── Modifier annonce

- Admin ── Valider annonces

- Admin ── Gérer utilisateurs



**1.3 Diagramme de classes (UML)**

**Classes principales**

**User**

* Id
* Nom
* Email
* mot\_de\_passe
* rôle

**Voiture**

* Id
* Marque
* Modèle
* Année
* Prix
* Kilométrage
* Type carburant
* Description
* vendeur\_id

**Image**

* id
* voiture\_id
* url

**Commande**

* Id
* acheteur\_id
* voiture\_id
* Statut
* Date

**Message**

* Id
* expéditeur\_id
* destinataire\_id
* Contenu
* Date

**1.4 Diagramme de séquence (Achat)**

Acheteur → Site : Rechercher voiture

Site → BD : Charger résultats

Acheteur → Site : Voir détail

Acheteur → Site : Acheter

Site → Paiement API : Transaction

Paiement → Site : Confirmation

Site → BD : Enregistrer commande

**1.5 Modélisation base de données (MLD)**

**Tables** :

users

* Voitures
* Images
* Commandes
* Messages
* Paiements

**Relations** :

1 User → N Voitures

1 Voiture → N Images

1 User → N Commandes