Nice — your uploaded **OpenAPI schema** gives a full view of your carpooling app’s current backend structure. Based on it, here’s what your app already supports and what you can **add next** to make it more complete, production-ready, and feature-rich 👇

**✅ Existing Core Functionalities**

From your schema, your app already includes:

* **Authentication** via JWT (/api/token/, /api/token/refresh/)
* **User Management**
  + Registration (/user/register/)
  + Profile management for driver, passenger, and admin
  + Verification workflow for drivers (driver-verification, approve-verification-requests)
* **Rides**
  + Create, update, cancel, complete rides
  + View rides and “my rides”
* **Bookings**
  + Book a ride
  + Cancel booking
  + View “my bookings”
* **Vehicles**
  + Manage vehicle makes, models, and owner vehicles
* **Driver Documents** upload and management

**🚀 Recommended Enhancements**

**🧭 1. Ride Matching & Search Optimization**

Add intelligent ride matching features:

* **Geolocation-based search**: integrate Google Maps or OpenStreetMap to find rides near user’s location.
* **Dynamic filtering** (price, time, driver rating, distance).
* **Real-time seat availability update** (using WebSockets or Django Channels).

**🗺 2. Real-time Ride Tracking**

Integrate:

* **Driver location sharing** (via GPS or simulated API updates)
* **Passenger tracking**: show driver ETA.
* **Trip progress updates** using WebSockets or Firebase.

**💬 3. In-App Communication**

* **Chat system** between passenger and driver (using Django Channels or third-party chat API like Twilio Conversations).
* Message notifications when ride is accepted or canceled.

**💰 4. Payment & Wallet Integration**

* **Payment gateway** (Razorpay, Stripe, PayPal)
* **Wallet balance** tracking
* **Refunds** for canceled rides
* **Transaction history** model and APIs

**🌟 5. Ratings & Feedback**

* Allow passengers to **rate drivers and rides**
* Compute **average driver rating**
* Add **feedback comments** model with moderation by admin

**📧 6. Notification System**

You already have Celery — extend it for:

* **Email notifications** (ride confirmation, cancellation, verification approved)
* **SMS or Push notifications** using Twilio / FCM
* Central notification model for in-app alerts

**👮 7. Admin Dashboard Features**

For /admin role:

* Approve/reject rides and documents
* View reports: total rides, revenue, cancellations, top drivers
* Manage users & suspend accounts

**🧾 8. Ride History & Analytics**

* Add **ride history** endpoints per driver & passenger
* Use Django ORM aggregation to show:
  + Total rides
  + Average fare
  + Miles covered
* Add a **/analytics/** endpoint for reports

**🔐 9. Security & Verification**

* Add **two-step verification** for login
* Email or SMS OTP for password reset
* Admin audit log for all critical actions

**🧑‍🤝‍🧑 10. Referral & Loyalty System**

* Users get credits when they refer friends
* Track referral bonuses and reward active users

**🧠 11. Smart Suggestions (optional AI features)**

* Suggest frequent routes (home ↔ office)
* Predict ride duration using previous data
* Dynamic fare estimation