Ride Hailing Service Backend: Machine Coding Problem

We need to build a backend for the ride hailing platform like Uber.

Mandatory Functional Requirements:

- Register a user
- Register a driver
- Book a ride if any driver is available within a certain radius.
- End a ride and get the cost for it.
- Update location of a cab.
- Implement a tiered pricing strategy: one of the examples could be -
  - Minimum ride price - 50 rs
  - First 2 KMs at 10 Rs / KM
  - 3-5 KMs at 8 Rs / KM
  - 6 KM and beyond at 5 rs / KM
- Show ride history for a user: ongoing and completed rides.
- Show ride history for a driver: ongoing and completed rides.

Bonus Requirements:

Add support for different types of cars.

- Sedan and Hatchback cars.
- Different pricing rates for hatchback and sedan.
- When hailing a cab, if hatchback is not available upgrade the user to a sedan at no extra cost.

Add support for coupon codes.

- Add a coupon code
- Delete a coupon code
- Apply a valid coupon code when starting a ride to get it at the discounted price.

Points to Note:

Feel free to use whatever technologies you're the most comfortable with. This includes any sort of open-source third-party frameworks.
Your priorities should be:

- Workable solution (Shortcuts are fine, but be prepared to justify them and explain better solutions you would have implemented with more time)
- Clean Code & Design (How code is divided into proper readable separations)
- Extensibility of the solution (How easily can it solve for some other strategies to achieve similar / better outcome)

Using a database is optional. You can store data in-memory but modeling should be complete.

You have an option to build a CLI app or an API based web app.

Feel free to assume any other detail. Call out all the assumptions before the start of the demo.

Also, plan to give a 30 minutes demo on your submission.

10 minutes - Showing your working solution in action and how it performs

10 minutes - Explaining approach and walking through the code

10 minutes - Talking about extensibility, other use cases or ways to solve this problem better and Q&A

Please reach out if you are stuck anywhere or need any help. We want this to be an enjoyable experience for you.