Car Hire Management System.

A- Requirements:

Details: The main focus of the business is renting cars and vans, and the database is to manage the booking system.

1. Vehicles are categorized into small cars (suitable for carrying up to 4 people), family cars (suitable for carrying up to 7 adults), and vans.
2. Information stored for each booking includes customer, car, date of hire and date on which the vehicle is to be returned.
3. A customer cannot hire a car for longer than a week.
4. If a vehicle is available, the customer's details are recorded (if not stored already) and a new booking is made.
5. Potential or existing customers can book a vehicle up to 7 days in advance depending on availability.
6. Customers must pay for the vehicle at the time of hire.
7. On receiving an enquiry, employees are required to check availability of cars and vans.
8. An invoice is written at the time of booking for the customer.
9. If the booking has been made in advance, a confirmation letter will be sent to the customer.
10. A report is printed at the start of each day showing the bookings for that particular day.

B- Deliverables: (all deliverables should be placed on GIT (your personal Github account). and we may want to check your commits and branches.)

1. An ERD diagram describes the DB design, field types, relationships, constraints, etc. (a screenshot on your repo is fine)
2. SQL which implements above ERD. (MySQL)
3. A Python microservice implemented using Flask microframework that should connect to MySQL DB and have the following endpoints:
   an endpoint to add new customer.
   a. an endpoint to update customer
   b. an endpoint to delete customer
   c. an endpoint to get customer.

Please don’t use ORMs and follow solid principles.