Half Life assignment- Courier Service App

You have been tasked with building a courier service application for a courier service. The application will allow clients of the courier service to create and track shipments as users via their own account. The application should include the following features:

1. User Registration: Users should be able to register with the application by providing their email address and suitable client information. Implement the necessary backend API endpoints to handle user registration.

2. User Login: Registered users should be able to log in to the application using their email address and password. Implement the necessary backend API endpoints to handle user login.

3. Shipment Creation: Authenticated users (client users) should be able to create shipments by providing the following details: sender name & sender address (captured from client user profile), recipient name, recipient address, and shipment details (identify appropriate shipment details to capture) . Implement the necessary backend API endpoints to handle shipment creation.

4. Shipment Tracking: Users should be able to track the status of their shipments by providing the shipment tracking number. Implement the necessary backend API endpoints to handle shipment tracking.

5. User Dashboard: Authenticated users should have access to a dashboard where they can view and manage their shipments. The dashboard should display the list of shipments created by the user, along with their current status. (Optional).

6. Admin user: admin user will have the same dashboard & able to view all shipments created. Change status as needed.

7. Frontend Interface: Create a user-friendly frontend interface to interact with the backend API endpoints. Use React components to implement the user registration, login, shipment creation, shipment tracking, and user dashboard features.

8. Data Persistence: Use PostgeSQL to store user information, shipment details, and tracking status. Design the necessary database schema to store this information.

9. Error Handling: Implement appropriate error handling mechanisms on both the frontend and backend to provide meaningful error messages to users in case of any failures.

10. Authentication and Authorization: Implement authentication and authorization mechanisms to ensure that only authenticated users can create shipments and view their dashboard.

11. Documentation: Provide documentation on how to set up and run the application locally. Include instructions on how to run the frontend and backend components, as well as any additional setup steps required.

12. Create user stories for a minimum of 5 features to be built in for the next iteration of the above product as you see fit focusing on a hypothetical courier service organization using this product. The key objectives of the courier service organization is to provide ease of use for their clients while bringing in efficiencies to their logistics operations. NOTE: You're **not** needed to implement these features.

Constraints

- Tech Stack to Use: PERN (PostgreSQL, Express, React, Node) written in Typescript

You can use any additional libraries, or tools that you find necessary to complete the assignment efficiently.

- Focus on implementing the core functionality outlined above, rather than spending too much time on UI design.
- Use best practices for code organization, readability, and maintainability.
- You don't need to worry about deploying the application to a production environment. It should be runnable locally.

Evaluation criteria

1. Functionality (2 marks):
2. Code Quality (3 marks):
3. Error Handling (1 mark)
4. Database and Data Handling (1.5 mark):
5. User Experience (1 mark):
6. Authentication and Authorization (1 mark):
7. Documentation (0.5 marks):

Creation of user stories will be evaluated separately.

Submission

Push the code to a public github repo & provide us with the repo details. Provide clear instructions on running the code locally documented on the readme.md file

Good luck with the assignment! If you have any questions, feel free to ask.