**The Challenge**

Design an Architecture diagram, Sequence Diagrams, End Point List/ Contract and Database Diagram for complete web application (FE + BE) of a Car Auction Platform.

1. Next.js:

Reason: It is easy to build a website using it and have multiple library for ReactJS

2. Sequelize (MySQL):

Reason: It provides ORM (Object-Relational Mapper) that make it easier to interact to database including migration and seeder.

3. React-Toasty:

Reason: It offers a user-friendly way to display alert messages without additional state management for simple notifications.

1. JSON Web Token (JWT):

Reason: A popular choice for authentication as it's stateless and allows storing user information securely on the client-side. Although it is best to use next-auth library for Next.js but it having an issue with Sequelize

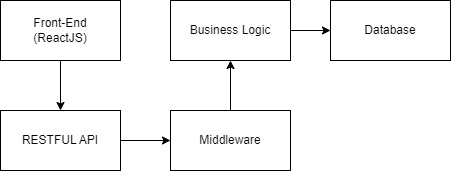
5. Bootstrap:

Reason: It helps me create the website with pre-built components and since it's mobile responsive.

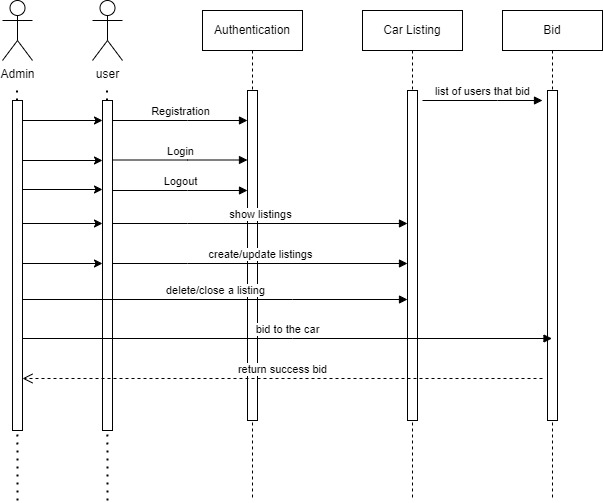
Additional Notes:

Integrating Socket.IO would enable real-time updates on bids and auction statuses. Although, I prioritize the main functionality to reach the deadline.

**Architecture Diagram**

****

**Sequence Diagram**

****

**Endpoint List/Contract:**

1. Login: POST /api/auth/login
2. Logout: POST /api/auth/logout
3. Register: POST /api/auth/sign-up
4. Current Logged User: GET /api/auth/current-user
5. All Listings: GET /api/~/listings
6. Get Listing: GET /api/~/listings/<listing\_id>
7. Update Listing: POST /api/~/listings/modify
8. Create Listing: POST /api/~/listings/create
9. Delete Listing: POST /api/~/listings/delete
10. Bid: POST /api/~/bid

**Database Diagram:**

