Project Design Phase-II

Technology Stack (Architecture & Stack)

Date	14 APRIL 2025
Team ID	SWTID1742640402
Project Name	MyRide
Maximum Marks	4 Marks

Technology Stack

To build a scalable, secure, and high-performance **Cab Booking App**, we carefully selected the following technologies based on project goals, team expertise, and industry standards.

1. **Frontend (React.js)**

Technology	Purpose	Justification
React.js	UI development	Component-based architecture, fast rendering with Virtual DOM.
Vite	Build tool	Faster builds and HMR (Hot Module Replacement) than Webpack.
Context API	State management	Lightweight for global state (user auth, booking data).
Axios	HTTP requests to backend	Promise-based, interceptors for JWT auth.
Google Maps API	Real-time maps and location services	Industry standard for geolocation and route plotting.
Tailwind CSS	Styling	Utility-first CSS for rapid mobile-first development.

2. Backend (Node.js + Express.js)

Technology	Purpose	Justification
Node.js	Backend runtime environment.	Non-blocking I/O for scalability.
Express.js	Framework for RESTful APIs.	Minimalist + middleware support.
MongoDB	NoSQL database for flexible data storage.	Geospatial queries for proximity search.
Mongoose	ODM (Object Data Modeling) for MongoDB.	Schema validation + hooks.
JWT	Token-based authentication.	Stateless + secure.
Socket.io	Real-time driver tracking and notifications.	WebSocket support.

Why This Stack?

- **Performance**: React (Virtual DOM) + Node (non-blocking I/O) ensure speed.
- **Scalability**: MongoDB sharding + Redis caching handle growth.
- **Safety**: JWT + HTTPS + rate limiting for secure APIs.
- **Cost-Effective**: Open-source tools reduce licensing costs.