Frontend Technologies:

React.js:

Role: Provides an easy way to develop a responsive interface which is important for all users of the application.

Allows the creation of user interfaces that can adapt to screen sizes and reusable menus.

Socket.io:

Role: Allows the application to send and receive real time updates for riders, restaurant owners and riders.

Used for enabling instant updates on order status, including "order placed," "order picked up," "out for delivery," and "delivered."

Ensure riders can update their status dynamically, and customers can track deliveries in real-time.

Backend Technologies:

Express.js:

Role: The backend framework to handle any API requests and routing.

Used for handling user authentication, order processing, and restaurant management.

Enables RESTful API endpoints for seamless communication between the frontend and backend.

Node.js:

Role: Provides an environment to run JavaScript on the server.

Used for handling asynchronous operations efficiently, such as order placements, real-time updates, and user authentication.

Allows the use of event-driven programming to enhance performance and responsiveness.

MySQL:

Role: Database for storing restaurant details, accounts, orders, bills.

Used for maintaining structured and relational data, ensuring consistency and quick retrieval.

Enables secure storage of user credentials, restaurant menus, order histories, and transaction records.

Multer:

Role: Middleware for handling file uploads in Node.js.

Used for uploading restaurant images, food item pictures, and user profile photos before storing them in MySQL.

Ensures smooth image handling by managing file storage efficiently and securely.