Technical Interview Questions & Answers

Q1: Can you give a high-level overview of your project?

A: My project is a student inquiry management system where students can submit inquiries about courses. The admin can view these inquiries and approve or reject them. The system includes form handling, validation, email notifications, and database storage.

Q2: What are the key features of your project?

A: - Students can submit inquiries using a form.
- Form validation using react-hook-form and yup.
- Admin can view and approve/reject inquiries.
- Email notifications sent via Nodemailer.
- Database managed with MongoDB & Mongoose.

Q3: Why did you choose React.js for this project?

A: React.js provides a component-based architecture, improves performance with a virtual DOM, and efficiently manages state updates.

Q4: What is react-hook-form, and why did you choose it over Formik?

A: React-hook-form is a lightweight form-handling library preferred over Formik for better performance, fewer re-renders, and easy integration with validation libraries like yup.

Q5: How does yup help in form validation?

A: Yup is a schema validation library that defines validation rules and integrates with react-hook-form to provide real-time validation.

Q6: How did you manage global state in your project?

A: I used the Context API for state management instead of Redux. It helps avoid prop drilling and allows global state sharing without extra dependencies.

Q7: Why did you use moment.js for date formatting?

A: Moment.js provides comprehensive date formatting and manipulation features, making it easier to format dates in a readable manner.

Q8: Why did you choose Express.js over Koa.js?

A: Express.js is widely used and has built-in middleware support, making it easier to handle requests, routes, and authentication.

Q9: Why did you use MongoDB instead of MySQL?

A: MongoDB is a NoSQL database, making it easier to store unstructured data. It scales better and does not require predefined schemas like MySQL.

Q10: What is Mongoose, and how does it help?

A: Mongoose is an ODM (Object Data Modeling) library for MongoDB. It helps define schemas and models for structured data storage.

Q11: How does Nodemailer work in your project?

A: Nodemailer is used to send email notifications when an inquiry is submitted or approved. It connects to an SMTP server to send emails programmatically.

Q12: How do you handle errors in Express.js?

A: I use a centralized error-handling middleware to ensure that all errors return a consistent response.

Q13: How do you optimize database queries?

A: Using indexes for faster lookups, using lean queries in Mongoose, and fetching only required fields using .select().

Q14: How do you handle API request errors in Axios?

A: By using the catch method to handle failed API requests without breaking the application.

Technical Interview Questions & Answers

Q15: How do you prevent memory leaks in a React app?

A: By cleaning up event listeners, using useEffect cleanup functions, and avoiding unnecessary re-renders.

Q16: How did you deploy your project?

A: - Frontend: Hosted on Vercel.
- Backend: Hosted on Render.
- Database: MongoDB Atlas.

Q17: How do you secure API routes?

A: Using JWT authentication, implementing rate limiting, and sanitizing input to prevent SQL injection & XSS attacks.

Q18: How do you optimize React performance?

A: Using lazy loading, memoizing expensive calculations with useMemo(), and using React Profiler for performance analysis.

Q19: What are the benefits of using Context API over Redux?

A: Context API is simpler and avoids unnecessary re-renders, making it more suitable for small applications.

Q20: How do you test your application?

A: I use Jest and React Testing Library for frontend testing and Postman for API testing.