

সহজ (Easy) টেকনিক্যাল প্রশ্ন ও উত্তর

প্রশ্ন 1: JWT কী এবং কেন ব্যবহার করা হয়?

উত্তর: JWT (JSON Web Token) একটি স্বাক্ষর করা টোকেন যা ইউজারকে authenticated থাকার প্রমাণ দেয়। সার্ভার টোকেন সাইন করে পাঠায়, ক্লায়েন্ট সেটি cookie অথবা localStorage-এ রাখে। আগ্রহী কারণে: stateless authentication, স্কেলিং সহজ, token expiry নির্দিষ্ট করা যায়। DreamNest-এ আমরা HTTP-only cookie তে JWT রাখবো যাতে XSS বুঁকি কমে।

প্রশ্ন 2: HTTP-only cookie কেন ব্যবহার করবেন token রাখার জন্য?

উত্তর: HTTP-only cookie ব্রাউজার জাভাস্ক্রিপ্ট থেকে অ্যাক্সেস করা যায় না, তাই XSS আক্রমণ থেকে টোকেন রক্ষা পায়। CSRF রোধ করতে additional কৌশল (SameSite, CSRF token) ব্যবহার করা উচিত।

প্রশ্ন 3: Mongoose Schema-তে User মডেলে কি ফিল্ড থাকবে?

উত্তর: প্রধান ফিল্ড: name, email (unique), password (hashed), phone, role ("user" | "admin"), createdAt। password bcrypt দিয়ে hash করে সংরক্ষণ করা হয়।

প্রশ্ন 4: Property মডেলে কোন ফিল্ডগুলো দরকার?

উত্তর: title, description, price, type ("flat" | "land"), location (string), images (string[] - image URLs), createdAt।

প্রশ্ন 5: Booking কি রকম হবে?

উত্তর: Booking মডেল: userId (ref User), propertyId (ref Property), status (Pending|Approved|Rejected), date (booking date/createdAt)।

প্রশ্ন 6: Image upload via link কীভাবে implement করবেন?

উত্তর: ফ্রন্টএন্ডে user image URL ইনপুট করবে; ব্যাকএন্ডে সেটি গ্রহণ করে সরাসরি images array-তে রাখবে। যদি সার্ভার সাইডে ডাউনলোড/বকে আপলোড করা দরকার হয় (Cloudinary), তখন ব্যাকএন্ড ওই লিংক থেকে ইমেজ fetch করে cloud-এ upload করে URL সেভ করবে।

প্রশ্ন 7: Protected route কীভাবে করবেন React-এ?

উত্তর: React Router ব্যবহার করে একটা PrivateRoute বা HOC বানাবেন। লোড হওয়ার সময় Axios interceptor বা context থেকে ইউজার অবস্থা (isAuthenticated, role) চেক করে না হলে /login-এ redirect করবেন।

প্রশ্ন 8: Axios interceptor কেন দরকার এবং কী করবে?

উত্তর: Interceptor response এ token expired হলে refresh token লজিক বা auto logout handle করবে এবং request এ auth header বা credentials attach করবে। DreamNest-এ token HTTP-only cookie-তে থাকলে axios.defaults.withCredentials = true দেবেন।

প্রশ্ন 9: Role-based Authorization কিভাবে করবেন?

উত্তর: ব্যাকএন্ডে middleware লিখে JWT verify করে user role চেক করবেন (e.g., if (req.user.role != 'admin') return res.status(403)). ফ্রন্টএন্ডেও UI hide/route protection করবেন।

প্রশ্ন 10: Nodemailer দিয়ে কনফার্মেশন ইমেল কিভাবে পাঠাবেন?

উত্তর: Nodemailer SMTP বা service (Gmail/SendGrid) ব্যবহার করে নতুন বুকিং/রেজিস্ট্রেশন ইমেল পাঠাবেন। ব্যাকএন্ডের relevant event (booking creation) এ async ইমেল টাঙ্ক কল করা উচিত।

৩. মাঝারি (Intermediate) টেকনিক্যাল প্রশ্ন ও উত্তর

প্রশ্ন 11: MongoDB-তে relation handling কেমন করবেন (User-Booking-Property)?

উত্তর: Mongoose populate ব্যবহার করা হয়। Booking model-এ userId ও propertyId রাখলে Booking.find().populate('userId', 'name email').populate('propertyId', 'title price') করে ইউজার ও প্রপার্টি ডিটেইল আনা যায়।

প্রশ্ন 12: Search ও Filter কিভাবে করবেন Frontend/Backend-এ?

উত্তর: Backend-এ query params গ্রহণ করবেন (q, type, minPrice, maxPrice, location)। Mongoose query বানিয়ে filter করবেন। Frontend-এ debounce করা search input এবং ট্যাগ/ড্রপডাউন ফিল্টার ব্যবহার করবেন।

প্রশ্ন 13: Google Maps Integration প্রপার্টি পেজে কিভাবে?

উত্তর: React Google Maps library (e.g., @react-google-maps/api) ব্যবহার করে property location (latitude/longitude) দেখাবেন। Address থেকে lat/lng পেতে Geocoding API ব্যবহার করা যেতে পারে বা admin panel-এ lat/lng ইনপুট নিতে পারেন।

প্রশ্ন 14: Admin dashboard এর জন্য performance optimization কিভাবে করবেন?

উত্তর: Pagination + server-side aggregation (count, recent bookings limited), indexes (MongoDB) on frequently queried fields (e.g., createdAt, type, price), reduce payload size, frontend memoization (React.memo) এবং lazy loading charts।

প্রশ্ন 15: Security best practices?

উত্তর:

- Password hashing (bcrypt)
 - JWT secret strong ও expiration
 - Rate limiting, helmet middleware, CORS সঠিক কনফিগার
 - Input validation (Joi / express-validator)
 - Store token in HTTP-only cookie
 - Use prepared queries / avoid string building
-

৪. অ্যাডভান্সড (Advanced) টেকনিক্যাল প্রশ্ন ও উত্তর

প্রশ্ন 16: MongoDB aggregation দিয়ে কী ধরণের ড্যাশবোর্ড ডেটা তুলবেন?

উত্তর: মোট properties count, মোট ব্যবহারকারী count, মোট বুকিং count, শেষ 7 দিনের বুকিং গ্রাফ (group by date), property-type distribution (group by type)। Aggregation pipeline দিয়ে \$group, \$match, \$sort, \$limit ইত্যাদি ব্যবহার করা হবে।

প্রশ্ন 17: How to implement server-side rendering (SSR) or SEO-friendly property pages?

উত্তর: Next.js ব্যবহার করলে SSR/SSG সহজ। DreamNest-এ যদি SEO গুরুত্বপূর্ণ হয়, property page-গুলো Next.js-এ SSG (getStaticProps) বা SSR (getServerSideProps) করে রেন্ডার করা যাবে; metadata (og tags) সার্ভার-সাইড injectable করা যাবে।

প্রশ্ন 18: How to handle concurrent booking attempts for same property?

উত্তর: Transactional control বা optimistic locking ব্যবহার করা যেতে পারে। সহজভাবে, booking create করার আগে property availability চেক করে এবং DB-level এ unique reservation flag (or status) দিয়ে transactionally সেট করা। MongoDB transactions (replica set) ব্যবহার করলে নিরাপদ।

প্রশ্ন 19: Testing strategy কী হবে?

উত্তর: Unit tests (Jest) ও Integration tests (Supertest for APIs). Frontend unit tests (React Testing Library). E2E tests (Cypress) critical flows: register/login, booking creation, admin approve flow।

৫. নন-টেকনিক্যাল (Non-Technical) প্রশ্ন ও উত্তর

প্রশ্ন 20: DreamNest প্রজেক্টের MVP কী হবে?

উত্তর: MVP: ইউজার রেজিস্টার/লগইন, প্রপার্টি লিস্টিং ও ডিটেইল পেজ, বুকিং ফর্ম (manual confirmation), অ্যাডমিন লেভেলে প্রপার্টি যোগ/আপডেট/ডিলিট এবং বুকিং স্ট্যাটাস ম্যানেজ করা।

প্রশ্ন 21: টেকনিক্যাল চ্যালেঞ্জগুলো কী হতে পারে?

উত্তর: Authentication flow, secure token handling, image upload/scaling, concurrent booking prevention, deployment CORS ও environment configuration, এবং email deliverability।

প্রশ্ন 22: টিমের পাশে আপনি কীভাবে প্রোজেক্ট ডেলিভার করবেন?

উত্তর: Agile approach — 2-সপ্তাহের sprint, daily standup, Git workflow (feature branches + PR), code review, CI (build + tests), staging environment, এবং release checklist।

প্রশ্ন 23: ইউজারকে কিভাবে অনবোর্ড করবেন (growth) ?

উত্তর: সহজ ইউআই, clear CTAs (contact/book), email notifications, basic SEO, social sharing, local SEO (Google My Business), এবং referral incentives।

প্রশ্ন 24: কীভাবে প্রজেক্টের সিকিউরিটি এবং প্রাইভেসি বজায় রাখবেন?

উত্তর: Minimum data collection, encrypted transport (HTTPS), secure cookie, role-based access, password strength requirement, privacy policy ও user consent (GDPR/নিয়ম মনে রেখে)।

৬. দ্রুত (Quick) প্রশ্ন-উত্তর—শর্ট ফরম্যাট (প্রেজেন্টেশনের জন্য)

1. Q: JWT vs Session — কোনটা?

A: JWT for stateless, sessions for server-side state; choose JWT + secure cookies for DreamNest.

2. Q: Password hashing algorithm?

A: bcrypt (salt rounds = 10–12)

3. Q: Image storage কীভাবে?

A: Cloudinary / S3 preferred; link-based upload allowed!

4. Q: Deployment hosts?

A: Frontend: Vercel; Backend: Render/Heroku/DigitalOcean; DB: MongoDB Atlas.

5. **Q:** Server-side validation କୁଣ୍ଡାଳିରେ କିମ୍ବା କିମ୍ବା?

A: express-validator ବା Joi.