

CLIENT-SIDE FUNDAMENTALS

Topic	Description
HTML	Markup for web pages. Elements, forms, tables, semantics.
CSS	Layout, colors, responsiveness, Flexbox, Grid, animations.
JavaScript	DOM manipulation, events, API calls (fetch/AJAX).
Frameworks	React, Angular, Vue — build SPAs, reusable components.
State Management	Managing data in the browser — useState, Redux, Context API.
Routing	Changing views without page reload — React Router, Vue Router.
Client-side Storage	Cookies, LocalStorage, SessionStorage, IndexedDB.
API Consumption	Calling REST APIs, GraphQL from the client.
Auth(Client)	Handling tokens (JWT) in storage, sending headers.
Performance	Lazy loading, code splitting, image optimization.
Accessibility (a11y)	Making websites usable for everyone.
Responsive Design	Mobile-first, media queries.
Build Tools	npm, Webpack, Vite, Babel.
Testing	Unit & integration testing — Jest, React Testing Library.

SERVER-SIDE FUNDAMENTALS

Topic	Description
Servers	Physical or cloud machines that run your app — e.g., Apache, NGINX, Node.js server.
HTTP Protocol	Request/Response, methods (GET, POST, PUT, DELETE), status codes.
APIs	RESTful APIs, GraphQL APIs, SOAP (older).
Routing	URL routing, controllers — e.g., Express routes in Node.js.
Middleware	Functions that handle requests/responses in pipelines — logging, auth, parsing JSON.
Databases	SQL (PostgreSQL, MySQL) or NoSQL (MongoDB). CRUD operations.
Authentication	Sessions, JWT, OAuth, cookies.
Authorization	Role-based access, permissions.
Business Logic	Validations, calculations, workflows.
Environment Variables	Secure config — .env files.
File Uploads	Handling multipart forms, storing files.
Security	CORS, rate limiting, sanitizing input, HTTPS, CSRF, XSS prevention.
Error Handling	Graceful responses, logging, fallback.
Testing	Unit, integration, end-to-end — Jest, Supertest.
Scalability	Load balancing, horizontal scaling, stateless design.
Deployment	Deploying to cloud (AWS, GCP, Azure, Vercel), (Docker).
Serverless	Functions-as-a-service (AWS Lambda, Vercel functions).
Monitoring	Logs, health checks, metrics, APM tools.

CLIENT–SERVER INTERACTION

Topic	Description
Requests & Responses	How the browser talks to the server (fetch, axios).
JSON	Data format for APIs.
CORS	Cross-Origin Resource Sharing — allows/restricts API calls from other origins.
Cookies & Sessions	Maintaining user sessions.
WebSockets	Real-time communication — e.g., chats.
GraphQL	Flexible API query language.
RESTful Principles	Resources, stateless operations, versioning.