SEO QUARTZ

Industrial Training Programme | seoquartz.com

Web Development

Back End, Full Stack and React | Industrial Training Syllabus — 45 Days | 90 Days | 6 Months

---

About the Programme

This programme takes students from writing their first HTML page to building and deploying complete applications with a React front end, a Node.js back end and a real database. Every module ends in working code, and students keep a public GitHub profile through the course that grows into their portfolio.

Programme Options

Duration	Approx. Training Hours	Level & Outcome
45 Days	Around 65 hours	Beginner. HTML, CSS, JavaScript, React basics and a Node back end, finishing with one full-stack application.
90 Days	Around 130 hours	Beginner to intermediate. Adds TypeScript, advanced React, authentication, payments and deployment, with two portfolio projects.
6 Months	Around 240 hours	Beginner to job-ready. Full-stack engineering with testing, deployment, system design basics, team projects and interview preparation.

Who Can Join

- Students of B.Tech, BCA, MCA, B.Sc (IT/CS), BBA, MBA, B.Com and similar streams
- Final-year students who need industrial training or an internship for their college requirement
- Graduates looking to start a career in this field
- Working professionals and business owners who want practical, hands-on skills
- No prior experience needed for the 45-day and 90-day programmes; basic computer use is enough

How the Training Runs

- Classroom and online batches, weekday and weekend options
- Roughly 2 hours of instruction per day, five days a week, plus daily practice time
- Every topic is taught on real tools and real accounts, not slides alone
- Small batches so each student gets individual feedback on their work.

45-Day Programme — Web Development Foundation

A fast, code-first course. Students write code from day one and finish with a working full-stack application deployed on a live URL.

Module Plan

Week	Topics Covered
Week 1 **Web Fundamentals, HTML & CSS**	How the web works — client, server, requests and responses; domains, hosting and DNS; HTML5 structure and semantic tags; forms and input types; CSS selectors, the box model and specificity; Flexbox and CSS Grid; responsive layouts; an introduction to Tailwind; setting up VS Code, Git and a GitHub account.
Week 2 **JavaScript Core**	Variables, data types and operators; conditions and loops; functions, parameters and scope; arrays and their methods; objects; ES6 features — arrow functions, template literals, destructuring, spread and modules; selecting and changing elements in the DOM; handling events; small builds such as a calculator and a to-do list.
Week 3 **JavaScript Advanced & APIs**	Callbacks, promises and async/await; the fetch API and working with REST endpoints; reading and shaping JSON; error handling; browser storage; form validation; debugging in Chrome DevTools; project — an app that pulls live data from a public API.
Week 4 **React Fundamentals**	Why component-based UI exists; JSX; components, props and state; useState and useEffect; rendering lists and using keys; conditional rendering; handling forms; React Router for multiple pages; folder structure and reusable components; project — a multi-page React application.
Week 5 **Back End with Node.js**	Node.js and npm; building a server with Express; routing and middleware; designing REST APIs; MongoDB and Mongoose; create, read, update and delete operations; testing endpoints with Postman; environment variables and keeping secrets out of code.
Week 6 **Full-Stack Integration & Deployment**	Connecting a React front end to your own API; login and registration with JWT; protected routes; handling CORS; MongoDB Atlas in the cloud; Git branches and pull requests properly.
Final Days **Capstone & Career Prep**	Capstone — a complete full-stack application with authentication, deployed and live; code walkthrough with a reviewer; cleaning up the GitHub profile and writing README files; resume and mock technical interview.

Practical Work in This Programme

- Three or four small JavaScript builds during the first weeks
- One React application with routing and API calls
- A REST API with a database behind it
- A deployed full-stack application with login

90-Day Programme — Full-Stack Web Development

This programme includes the foundation course and then covers the tools teams actually use at work — State management, secure authentication, payments, testing and a proper deployment workflow.

Module Plan

Weeks	Topics Covered
Weeks 1–2 **Foundations**	HTML5 and semantic markup; CSS3, Flexbox and Grid; responsive design; Tailwind CSS; Git and GitHub including branching and pull requests; the command line; Chrome DevTools; deploying a static site.
Weeks 3–5 **JavaScript in Depth**	Core JavaScript and ES6 and later features; the DOM and event handling; asynchronous programming with promises and async/await; working with REST APIs; closures, hoisting and how 'this' behaves; array and object patterns used daily; modules and bundlers; browser storage; an introduction to TypeScript — types, interfaces and why teams use it; three small projects.
Weeks 6–8 **React in Depth**	Components, props and state; useState, useEffect, useRef, useMemo and useContext; writing custom hooks; the Context API; React Router including nested and protected routes; forms with React Hook Form and validation; API integration with Axios; loading, error and empty states; code splitting and lazy loading; component libraries; building and deploying a production React build.
Weeks 9–11 **Back End & Database**	Node.js and Express in depth; designing REST APIs and versioning them; MongoDB with Mongoose, schema design and relationships; indexes and query performance; aggregation basics; pagination, filtering and search in APIs; authentication and authorisation with JWT, password hashing and common security mistakes; file uploads with Multer and Cloudinary; payment gateway integration with Razorpay or Stripe; sending transactional email; documenting an API.
Week 12 **Testing, Tooling & Deployment**	Postman collections and API testing; unit testing basics with Jest; environment configuration across development and production; continuous integration with GitHub, Deployment

Weeks	Topics Covered
Week 13 **Projects & Placement**	Two portfolio projects such as an e-commerce store or a booking platform; working in a Git workflow with other students; Resume Feedback

Projects Completed

- A React and TypeScript application with proper state management
- A secure REST API with role-based access control
- An e-commerce or booking platform with payments integrated
- An admin dashboard with charts and filters pulling from your own API
- Everything deployed, documented and running on live URLs

6-Month Programme — Professional Full-Stack Development

The complete programme. Students cover the full stack, the engineering practices that surround it, and finish with team projects run in sprints plus structured interview preparation.

Phase Plan

Phase	Topics Covered
Phase 1 **Weeks 1–3** **Web & Front-End Foundations**	HTML5 and semantic markup; CSS3, Flexbox and Grid; Tailwind CSS; responsive design and cross-browser behaviour; accessibility basics; the command line; Git and GitHub workflow; VS Code setup and useful extensions; deploying a static site on Vercel.
Phase 2 **Weeks 4–7** **JavaScript & TypeScript**	JavaScript from fundamentals to advanced — scope, closures, prototypes, 'this', the event loop; asynchronous patterns and error handling; working with APIs; object-oriented and functional approaches; modules and bundlers; TypeScript — types, interfaces, generics, utility types and using TypeScript with React; writing clean, readable code and reviewing others' code.
Phase 3 **Weeks 8–11** **React in Depth**	React core and all the common hooks; writing custom hooks; state management with Context, routing and route protection; forms and validation; performance optimisation, memoisation and lazy loading; error boundaries; testing components; Build reusable components; accessibility inside React applications; structuring a large front-end codebase.
Phase 4 **Weeks 12–16** **Back-End Engineering**	Node.js and Express at depth; REST API design, versioning and consistent response shapes; MongoDB and Mongoose — authentication, sessions, JWT and role-based access; security practices covering input validation, rate limiting and the common web vulnerabilities; file handling and cloud storage; background

Phase	Topics Covered
	tasks and scheduled jobs; integrating third-party services for payments, email and SMS; writing API documentation.
Phase 5 **Weeks 17–20** **Testing, Deployment & Architecture**	Git branching strategies and code review culture; unit, integration and end-to-end testing; environment and secret management;; deploying front-end applications on Vercel; MongoDB Atlas in production;
Phase 6 **Weeks 21–24** **Live Projects & Placement**	A team project runs in agile sprints with stand-ups, tickets and code review; Portfolio and GitHub Cleanup; Resume and LinkedIn

Live Work and Portfolio Output

- A team-built application delivered across several sprints
- At least one production-grade full-stack project with tests and continuous integration in place
- A front end on Vercel and an API on Render, both running under custom domains
- A documented API other students have consumed
- A GitHub profile with clean commit history, README files and live demo links

Assessment and Certification

How Students Are Evaluated

Component	Weight	Details
Daily and weekly assignments	25%	Practical tasks submitted after each topic and reviewed by the trainer
Module tests	20%	Short written and practical tests at the end of every module
Projects	35%	Graded on approach, execution, quality of work and documentation
Final viva and presentation	15%	Student presents the capstone project to a review panel
Attendance and participation	5%	Minimum 75% attendance required to sit the final review

What Every Student Receives

- Industrial training certificate from SEO Quartz with the duration mentioned
- Separate project completion certificate and a signed project report for college submission
- Complete study material, checklists, templates and recorded sessions
- A portfolio of finished work that can be shown to employers or clients
- Resume and LinkedIn profile review, mock interviews and placement assistance
- Continued access to the trainer group for doubts after the course ends

Tools and Platforms Covered

VS Code, Git and GitHub, Chrome DevTools, Node.js and npm, Express, React, Redux Toolkit, React Router, TypeScript, Tailwind CSS, SASS, Bootstrap, MongoDB and Mongoose, MongoDB Atlas, Postman, Jest, Docker, GitHub Actions, Vercel, Render, Cloudinary, Razorpay and Stripe.

Roles Students Can Apply For

- Front-End Developer (React)
- Back-End Developer (Node.js)
- Full-Stack Developer / MERN Stack Developer
- JavaScript Developer
- Web Application Developer
- Software Engineer Trainee and Junior Software Engineer

- API Developer
- Freelance developer building applications for clients
