

MEAN Stack Development

Build structured, testable web applications with MongoDB, Express, Angular and Node.js

Full-Stack Development · intermediate · Live online, instructor-led

This syllabus is a draft awaiting instructor review and may change.

A 96-hour live course in full-stack TypeScript: modern Angular with standalone components, signals and RxJS, Express APIs, MongoDB modelling and indexing, authentication, testing, Docker and a reviewed capstone.

Length and schedule

96 hours of live teaching, on either schedule:

- Weekday Intensive (Mon–Fri): about 20 weeks before holidays
- Weekend Evening (Sat & Sun evenings): about 24 weeks before holidays

Fees per student, whole course

- Group: Rs. 17,999
- Duo: Rs. 29,999
- One-to-One: Rs. 54,999

Launch fee — GST treatment to be confirmed.

What you will be able to do

- Write typed, asynchronous TypeScript for both browser and server
- Build Angular applications with standalone components, signals, services and dependency injection
- Use the Angular router, reactive forms and RxJS to build real user flows
- Choose a sensible state-management approach, from signal-based services to NgRx
- Design and build a validated REST API with Node.js and Express
- Model MongoDB data for real access patterns and keep queries fast with indexes
- Secure the application with authentication, role-based authorisation and OWASP practices
- Test, containerise and deploy the complete application

Before you start

- Basic programming in any language: variables, conditions, loops and functions
- HTML and CSS basics: you can build and style a simple static page
- Some exposure to JavaScript is helpful but not required; it is revised in the first module
- A laptop with at least 8 GB RAM (16 GB recommended for Docker) running Windows, macOS or Linux

Tools

Node.js 24 LTS, TypeScript, Angular and the Angular CLI, RxJS, NgRx, Angular Material, Express, MongoDB 8, Mongoose, MongoDB Compass, Zod, Jasmine and Karma or Jest, Supertest, Playwright, Docker and Docker Compose, GitHub Actions, Git and GitHub, Visual Studio Code

Syllabus

Module 1: TypeScript for full-stack development

The modern JavaScript and TypeScript foundations that Angular and the Node.js API are both built on.

1. Course setup and modern JavaScript (2 h)
2. Functions, objects, arrays and modules (2 h)
3. Asynchronous JavaScript (2 h)
4. TypeScript fundamentals (2 h)
5. Classes, generics and decorators in TypeScript (2 h)
6. TypeScript review and doubt-clearing (2 h)

Module 2: Angular architecture and components

How a modern Angular application is structured, and how to build it from standalone components and signals.

7. Angular architecture and project setup (2 h)
8. Standalone components and templates (2 h)
9. Component communication (2 h)
10. Signals and change detection (2 h)
11. Directives, styling and accessibility (2 h)
12. Angular components review and doubt-clearing (2 h)

Module 3: Services, dependency injection, routing and RxJS

Share logic through services, talk to APIs with HttpClient and RxJS, and turn screens into a routed application.

13. Services and dependency injection (2 h)
14. HttpClient and interceptors (2 h)
15. RxJS fundamentals (2 h)
16. RxJS with signals (2 h)
17. Routing and navigation (2 h)
18. Routing and RxJS review and doubt-clearing (2 h)

Module 4: Reactive forms and state management

Build robust forms and choose an approach to application state that fits the size of the application.

19. Typed reactive forms (2 h)
20. State with signal-based services (2 h)
21. NgRx overview and forms review (2 h)

Module 5: Node.js and Express API

Design and build a clean, versioned REST API with Express and TypeScript, with validation and consistent errors.

22. Node.js runtime and HTTP (2 h)
23. Express with TypeScript (2 h)
24. REST API design (2 h)
25. Layered architecture, validation and errors (2 h)
26. Configuration, uploads and connecting Angular (2 h)
27. API review and doubt-clearing (2 h)

Module 6: MongoDB data modelling and performance

Model data for MongoDB, work with it through Mongoose, and keep queries fast as data grows.

- 28. MongoDB fundamentals (2 h)
- 29. Data modelling for documents (2 h)
- 30. Mongoose models and validation (2 h)
- 31. Indexing and query performance (2 h)
- 32. Aggregation and data consistency (2 h)
- 33. Data layer review and doubt-clearing (2 h)

Module 7: Authentication and security

Sign users in safely, enforce what each role may do on the server, and defend against common web vulnerabilities.

- 34. Authentication, sessions and JWTs (2 h)
- 35. Role-based authorisation in the API (2 h)
- 36. Authentication in Angular (2 h)
- 37. OWASP basics and security review (2 h)

Module 8: Testing Angular and the API

Write the right tests at the right level so the application can change with confidence.

- 38. Testing Angular components and services (2 h)
- 39. API testing with Supertest (2 h)
- 40. End-to-end testing and doubt-clearing (2 h)

Module 9: Docker, CI and deployment

Package the application, automate checks on every change and deploy it.

- 41. Docker and Docker Compose (2 h)
- 42. Continuous integration with GitHub Actions (2 h)
- 43. Deployment, logging and doubt-clearing (2 h)

Module 10: Capstone project and final assessment

Build, harden and present the capstone application, with structured reviews and the final assessment.

- 44. Capstone kickoff and planning (2 h)
- 45. Capstone build lab (2 h)
- 46. Capstone mid-point review (2 h)
- 47. Final assessment (2 h)
- 48. Capstone presentations and final review (2 h)

Final project

Build and deploy a field-service job management system. A small appliance-repair company needs a web application where office coordinators log customer service requests, assign them to technicians and track each job from booking to completion; technicians see their daily schedule and record work done and parts used; and managers see open jobs, turnaround times and parts consumption. You receive a written brief with user stories and acceptance criteria, and build it individually or in a pair. Deliverables- An Angular front end built from standalone components, with lazy-loaded routes, route guards for UX, reactive forms with validation, signal-based state and loading, empty and error states on every data view.- An Express and TypeScript API with versioned routes, Zod validation, a consistent error format and role-based access for coordinators, technicians and managers.- A MongoDB schema with documented indexes for the main queries and an aggregation report on job turnaround times per technician.- Authentication with hashed passwords and secure session cookies or short-lived tokens, and server-side checks proving a technician

cannot read jobs assigned to someone else.- Automated tests: Angular component and service tests, Supertest API tests including authorisation denials, and one Playwright journey for assigning and closing a job.- A Dockerfile and Docker Compose setup, a GitHub Actions workflow that runs lint, tests and a production build, and a live deployment with no secrets in the repository.- A README with setup steps, an architecture diagram and known limitations. The project is presented and reviewed live in the final week, with feedback on code quality, security and design decisions.

Certificate

- At least 80% attendance
- All assignments submitted
- Projects passed
- Final assessment passed (60%)

A ZORK Academy course completion certificate; paying for the course does not by itself earn it.

Next step

Book a free consultation: <https://zorkacademy.online/consultation?course=mean-stack-development>

We help you prepare for work; we do not guarantee jobs, placements or salaries.