

Mobile Backend Development with Node and Express

Course Duration: 32 Hours

Course code: MBDNE

1. Course Overview

This course focuses on building robust and scalable backends for mobile applications using Node.js and Express.js. Learners will gain practical skills in developing RESTful APIs, managing authentication, handling databases, and integrating mobile applications with backend services. Through hands-on coding and project-based learning, students will learn to design, develop, and deploy efficient backend systems tailored for mobile apps.

2. What you'll learn?

By the end of this course, you will be able to:

- Understand the role of backends in mobile applications.
- Set up a Node.js and Express.js development environment.
- Build and test RESTful APIs for mobile apps.
- Implement CRUD operations for mobile data management.
- Use databases (SQL or NoSQL) with Node and Express.
- Secure mobile backends with JWT authentication and authorization.
- Manage file uploads and media handling for mobile apps.
- Apply error handling, logging, and debugging techniques.
- Deploy backend services to cloud platforms (AWS, Azure, or Heroku).
- Connect mobile applications (iOS/Android/React Native/Flutter) to backend APIs.

3. Target Audience

- Mobile app developers who want to build their own backends.
- Web developers transitioning into mobile backend development.
- Students and beginners interested in full-stack development.

- Professionals seeking to learn Express.js for mobile APIs.

4. Pre-Requisites

- Basic knowledge of JavaScript (ES6).
- Familiarity with mobile app development (React Native, Flutter, or native).
- Understanding of HTTP and APIs (recommended but not mandatory).

5. Course content

Day 1 – Introduction to Node.js & Express: Building Your First REST API

Theory (2.5 hrs)

- What is Node.js? Use cases for mobile backend
- Node.js Architecture (Event Loop, Call Stack, Non-blocking I/O)
- What is Express? Features and Ecosystem
- RESTful API concepts (Resources, Verbs, Status Codes)
- Project structure best practices

Hands-on (5.5 hrs)

- Install Node.js and initialize a project (npm init)
- Install Express, create your first API server
- Setup routes: GET, POST endpoints (/status, /ping)
- Use nodemon for auto-restart
- Build and test APIs using Postman
- Add error handling middleware and organize routes using Express Router

Day 2 – Routing, Middleware, and Data Handling

Theory (3.0 hrs)

- Express Router: Route grouping and modularization
- Middleware: types, order of execution
- Parsing incoming data (body-parser, JSON, URL-encoded)
- HTTP request/response lifecycle

- Introduction to API validation & basic error handling

Hands-on (5.0 hrs)

- Create modular routes: /users, /auth
- Implement custom middleware (logger, validator)
- Handle JSON and form data in requests
- Create a basic user registration/login flow (without database)
- Return meaningful status codes and error messages

Day 3 – MongoDB Integration & Secure Authentication

Theory (3.0 hrs)

- Introduction to NoSQL and MongoDB
- Overview of MongoDB Atlas and Compass
- Using Mongoose: schema, model, CRUD operations
- Authentication overview: Password hashing, JWT
- Securing APIs with middleware, CORS, Helmet

Hands-on (5.0 hrs)

- Connect to MongoDB Atlas using Mongoose
- Define User model with Mongoose schema
- Create registration endpoint with hashed password (using bcryptjs)
- Create login endpoint, issue JWT tokens
- Implement token-based authentication middleware
- Protect routes like /profile, /dashboard using auth middleware

Day 4 – API Features for Mobile + Deployment & Final Project

Theory (2.5 hrs)

- Structuring scalable mobile backend APIs
- Pagination, filtering, sorting for mobile responses
- Deployment options (Railway, Render, Vercel, Heroku)

- Managing environment variables securely
- Introduction to versioning APIs (/api/v1)

Hands-on (5.5 hrs)

- Add pagination and filtering to a /tasks endpoint
- Secure endpoints using JWT and role-based access (admin vs. user)
- Create .env config for secrets and DB connections
- Deploy project to Render/Railway with GitHub integration
- Final Project:
 - Build a mobile task manager backend
 - Features: user auth, CRUD on tasks, filtering, protected routes

