**Module 1: Introduction to Node.js**

* **What is Node.js?**
  + Overview of Node.js
  + Features of Node.js
  + How Node.js differs from traditional web servers
  + Advantages and disadvantages
* **Setting up Node.js**
  + Installing Node.js and NPM
  + Exploring Node.js command line
  + Writing and running your first Node.js script
  + REPL (Read-Eval-Print-Loop) in Node.js

**Module 2: Core Concepts of Node.js**

* **Event-Driven Architecture**
  + Event Loop, Event Emitter, Callbacks
  + Synchronous vs Asynchronous programming
  + Handling asynchronous code with callbacks, promises, and async/await
* **Modules in Node.js**
  + What are modules?
  + require and exports keywords
  + Built-in modules (fs, path, http, os, etc.)
  + Creating and using custom modules
  + Third-party modules and NPM

**Module 3: File System & Streams**

* **File System Module**
  + Reading and writing files asynchronously
  + Working with directories
  + File permissions and stats
* **Streams and Buffers**
  + What are streams?
  + Types of streams (Readable, Writable, Duplex, and Transform)
  + Handling streams and events
  + Buffering in Node.js
  + Piping streams

**Module 4: HTTP & Web Servers**

* **Creating an HTTP Server**
  + Building a simple HTTP server
  + Handling requests and responses
  + Working with different HTTP methods (GET, POST, etc.)
* **Serving Static Files**
  + Routing and serving static content (HTML, CSS, JS)
  + Parsing query strings and request bodies
* **RESTful API with Node.js**
  + Introduction to REST principles
  + Building a simple RESTful API
  + JSON handling in Node.js
  + Using tools like Postman to test APIs

**Module 5: Express.js Framework**

* **Introduction to Express.js**
  + What is Express.js and why use it?
  + Installing and setting up an Express application
  + Routing in Express (GET, POST, PUT, DELETE)
* **Middleware in Express**
  + What are middleware functions?
  + Using built-in and third-party middleware
  + Custom middleware development
* **Handling Forms and File Uploads**
  + Handling POST requests
  + File uploads using multer
* **Error Handling in Express**
  + Express error-handling middleware
  + Best practices for error handling
  + Sending appropriate HTTP status codes

**Module 6: Working with Databases**

* **Database Integration**
  + What is MongoDB? Introduction to NoSQL databases
  + Setting up MongoDB locally or on the cloud
  + Installing and configuring MongoDB with Node.js using mongoose or mongodb package
* **CRUD Operations**
  + Creating, reading, updating, and deleting records
  + Querying documents in MongoDB
  + Schema design in MongoDB
  + Relationships and population in Mongoose
* **Working with Relational Databases**
  + Using MySQL or PostgreSQL with Node.js
  + Performing SQL operations (SELECT, INSERT, UPDATE, DELETE)
  + Using Sequelize ORM for relational databases

**Module 7: Authentication & Authorization**

* **User Authentication**
  + What is JWT (JSON Web Token)?
  + Implementing JWT-based authentication
  + Password hashing with bcrypt
  + Securing routes and middleware for authentication
* **OAuth and Third-Party Authentication**
  + Introduction to OAuth
  + Integrating with Google, Facebook, and other OAuth providers
* **Role-Based Access Control**
  + Defining roles and permissions
  + Protecting routes with roles

**Module 8: Advanced Topics**

* **WebSockets and Real-Time Communication**
  + Introduction to WebSockets
  + Using socket.io for real-time communication
  + Building a chat application with WebSockets
  + Broadcasting messages and handling rooms
* **Task Scheduling in Node.js**
  + Using cron jobs in Node.js
  + Task automation and scheduling
* **Node.js Clustering and Load Balancing**
  + Clustering in Node.js for multi-core performance
  + Horizontal scaling strategies
  + Load balancing techniques

**Module 9: Testing in Node.js**

* **Introduction to Testing**
  + Importance of testing in software development
  + Types of testing (Unit, Integration, End-to-End)
* **Unit Testing with Mocha & Chai**
  + Writing unit tests for Node.js code
  + Using test libraries like Mocha and Chai
  + Running tests and handling assertions
* **Integration Testing**
  + Writing integration tests for APIs
  + Using supertest for testing HTTP routes
* **Test Coverage**
  + Using Istanbul/NYC for measuring test coverage

**Module 10: Error Handling & Debugging**

* **Error Handling in Node.js**
  + Using try/catch for synchronous code
  + Handling asynchronous errors in Node.js
  + EventEmitter error handling
  + Express error-handling middleware
* **Debugging Node.js Applications**
  + Debugging using console.log and built-in Node.js debugger
  + Using tools like Chrome DevTools, Visual Studio Code Debugger
  + Analyzing stack traces and error messages

**Module 11: Deployment & Production**

* **Node.js Application Deployment**
  + Hosting Node.js applications on Heroku, AWS, or DigitalOcean
  + Configuring environment variables for production
  + Understanding process managers like PM2 for managing production servers
* **Containerization with Docker**
  + Introduction to Docker
  + Containerizing a Node.js application
  + Writing Dockerfiles and using Docker Compose
* **Security in Node.js Applications**
  + Preventing common security vulnerabilities (SQL Injection, XSS, CSRF)
  + Using Helmet.js to secure Express apps
  + Rate limiting and input validation

**Module 12: Performance Optimization**

* **Profiling and Monitoring**
  + Profiling Node.js applications using Node’s built-in tools
  + Memory leaks and garbage collection
  + Using tools like New Relic, PM2 Monitoring, and others for performance analysis
* **Caching Strategies**
  + Introduction to caching
  + Implementing in-memory caching with Redis
  + Optimizing database queries and response times

**Module 13: Microservices & Distributed Systems**

* **Introduction to Microservices Architecture**
  + What are microservices?
  + Pros and cons of microservice architecture
  + Designing a microservice-based system using Node.js
* **Inter-Process Communication (IPC)**
  + Using message queues (RabbitMQ, Kafka)
  + Implementing RESTful APIs between microservices
  + Using gRPC for high-performance communication
* **API Gateways and Service Discovery**
  + Implementing API gateways
  + Handling service discovery in microservice architectures

**Module 14: GraphQL with Node.js**

* **Introduction to GraphQL**
  + What is GraphQL and why use it?
  + Differences between GraphQL and REST APIs
* **Setting up GraphQL in Node.js**
  + Building a GraphQL API with Apollo Server
  + Writing queries, mutations, and subscriptions
  + Integrating with databases (MongoDB, MySQL)

**Module 15: Practical Projects**

* **Building a Blog Application**
  + RESTful API with user authentication, CRUD operations for blog posts
* **Real-Time Chat Application**
  + Using WebSockets and socket.io for real-time communication
* **E-Commerce Platform**
  + Building an online store with product management, shopping cart, and payments
* **To-Do List API**
  + Simple CRUD-based to-do list API with user authentication and database integration