API Creation and Integration using Node.js & Express

Duration: 3 Days
Technology Stack: Node.js, Express.js, REST APIs, Postman, JSON, JWT, SQL/MongoDB

Prerequisites

Participants should have:

- Basic programming knowledge
- Understanding of JavaScript fundamentals
- Basic knowledge of HTTP and web applications
- Familiarity with databases is helpful

Day 1: REST API Design and Development

Topics

API Fundamentals

- Introduction to APIs
- Role of APIs in modern applications
- REST architecture and RESTful services
- Resources and endpoints
- Request-response lifecycle
- JSON data exchange

HTTP Fundamentals for APIs

- HTTP methods:
  - GET
  - POST
  - PUT
  - PATCH
  - DELETE
- HTTP status codes
- Request headers and response headers
- Path parameters
- Query parameters
- Request body

Node.js and Express.js for API Development

- Node.js architecture overview
- Creating a Node.js project
- npm and dependency management
- Express.js fundamentals
- Creating an Express server
- Routes and controllers
- Request and response handling

- Middleware
- Organizing API projects using:
    - Routes
    - Controllers
    - Services

REST API Design Practices

- Resource-oriented URL design
- API naming conventions
- API versioning concepts
- Consistent response structures
- Error response standards
- Centralized exception handling

Hands-on Labs

- Create and configure a Node.js and Express API project
- Build an Employee Management REST API
- Implement GET, POST, PUT/PATCH and DELETE endpoints
- Work with route, query and body parameters
- Test API endpoints using Postman
- Implement centralized error handling

Outcome

Participants will be able to design and develop structured REST APIs using Node.js and Express.js.

Day 2: Database Integration, Validation and API Security

Topics

Database Integration

- API and database architecture
- Connecting backend APIs with databases
- SQL and NoSQL database integration overview
- CRUD operations through APIs
- Repository and service patterns
- Database validation
- Handling database errors
- Transactions overview

API Input Validation

- Validating request parameters
- Validating JSON payloads
- Required and optional fields
- Input sanitization
- Handling invalid requests

- Creating meaningful validation responses

Authentication and Authorization

- Authentication vs authorization
- User registration and login workflow
- Password security
- Password hashing
- JSON Web Tokens
- Token generation and validation
- Authentication middleware

API Authorization

- Role-Based Access Control
- Securing API endpoints
- Public vs protected APIs
- User and administrator permissions

API Security

- CORS
- Secure HTTP headers
- Environment variables
- Managing secrets securely
- Preventing common API vulnerabilities
- OWASP API security awareness
- Logging and audit trails

Hands-on Labs

- Connect the Employee Management API to SQL or MongoDB
- Implement persistent CRUD operations
- Add API request validation
- Build registration and login APIs
- Generate and validate JWT tokens
- Secure selected API endpoints
- Implement role-based authorization

Outcome

Participants will be able to create database-driven APIs with validation, authentication, authorization and essential API security controls.

Day 3: API Integration, Testing and Production Readiness

Topics

API Integration

- Understanding API consumers

- Integrating APIs with:
    - Web applications
    - Mobile applications
    - Desktop applications
- Calling REST APIs using Fetch
- Async/await for API communication
- Sending headers and authentication tokens
- GET and CRUD integration
- Handling JSON responses

External API Integration

- Consuming third-party APIs
- API keys
- Authentication headers
- Query parameters
- Processing external API responses
- Handling API failures and unavailable services
- Timeout and retry concepts

Integration Error Handling

- Network errors
- API errors
- HTTP status handling
- Loading and error states
- Centralized client-side API services

API Testing

- API testing using Postman
- Creating Postman collections
- Environment variables
- Positive and negative testing
- Authentication testing
- CRUD workflow testing
- Integration testing concepts

Advanced API Practices

- Pagination
- Search
- Sorting
- Filtering
- File upload concepts
- Logging
- API documentation
- Introduction to OpenAPI/Swagger
- API deployment considerations
- Environment-based configuration

- Monitoring and troubleshooting APIs

Hands-on Labs

- Integrate a client application with the REST API
- Implement authenticated API calls
- Integrate a third-party API
- Implement search, filtering and pagination
- Build a Postman collection for complete API testing
- Document API endpoints
- Perform end-to-end API integration testing

Final Mini Project

Participants will build an **API-driven business application backend** containing:

- RESTful CRUD APIs
- Database integration
- Input validation
- User registration and login
- JWT authentication
- Role-based authorization
- Search, filtering and pagination
- Centralized error handling
- API testing with Postman
- External API integration
- API documentation