

Back-End Prog. Syllabus (Web Development)

Module 1: Introduction to Back-End Development

- ✓ **Understanding Back-End Basics**
 - Role of the Back-End in Web Development
 - Differences between Front-End and Back-End
 - Overview of Client-Server Architecture
 - Importance of Databases and APIs in Back-End
- ✓ **Setting Up the Environment**
 - Installing Required Tools: Node.js, Package Managers (NPM/Yarn)
 - Overview of Text Editors and IDEs (e.g., VS Code)
 - Introduction to Terminal Commands and Git for Version Control

Module 2: Server-Side Programming

Node.js Basics

- ✓ **Introduction to Node.js**
 - What is Node.js and Why Use It?
 - Installing and Setting Up Node.js
 - Understanding the Event Loop and Non-Blocking I/O
- ✓ **Core Modules**
 - http for Creating Servers
 - fs for File Handling
 - path and os Modules

- ✓ **Building a Simple Server**
 - Serving Static Files
 - Routing Basics
- ✓ **Package Management with NPM**
 - Installing and Using Packages
 - Understanding package.json

Express.js Framework

- ✓ **Introduction to Express.js**
 - Why Use Express.js for Web Applications?
 - Setting Up an Express Project
- ✓ **Routing**
 - Defining Routes (GET, POST, PUT, DELETE)
 - Route Parameters and Query Strings
- ✓ **Middleware**
 - Understanding Middleware Functions
 - Creating Custom Middleware
- ✓ **Error Handling**
 - Implementing Error Handling Middleware
- ✓ **Serving Dynamic Content**
 - Using Templates (e.g., EJS, Handlebars)
 - Rendering Pages with Data

Module 3: Working with Databases

SQL Databases

✓ Introduction to Relational Databases

- Understanding Tables, Rows, and Columns
- Basics of Database Normalization

✓ SQL Basics

- CRUD Operations: SELECT, INSERT, UPDATE, DELETE
- Filtering Data with WHERE, GROUP BY, and HAVING Clauses
- Joins (INNER, LEFT, RIGHT)

✓ Working with MySQL/PostgreSQL

- Setting Up a Database
- Writing SQL Queries
- Connecting Node.js with SQL Databases (e.g., mysql, pg packages)

NoSQL Databases

✓ Introduction to NoSQL

- Differences Between SQL and NoSQL Databases
- Understanding Document-Based Databases

✓ Working with MongoDB

- Setting Up MongoDB and Mongoose
- CRUD Operations in MongoDB
- Schema Design and Validation
- Querying Data with MongoDB

Module 4: APIs and Web Services

RESTful API Development

- ✓ **Understanding REST Architecture**
 - Principles of REST
 - HTTP Methods (GET, POST, PUT, DELETE)
 - HTTP Status Codes and Headers
- ✓ **Building APIs with Express.js**
 - Defining API Endpoints
 - Parsing and Validating Request Data
- ✓ **Data Exchange Formats**
 - JSON vs. XML
 - Sending and Receiving JSON Data

Advanced API Topics

- ✓ **Authentication and Authorization**
 - Using JWT (JSON Web Tokens)
 - Role-Based Access Control (RBAC)
- ✓ **API Documentation**
 - Introduction to Swagger/OpenAPI
 - Documenting and Testing APIs
- ✓ **Rate Limiting and Security**
 - Preventing API Abuse
 - Securing APIs with CORS and HTTPS

Module 5: Authentication and Security

✓ **User Authentication**

- Understanding Authentication vs. Authorization
- Implementing Sessions and Cookies
- Password Hashing with Bcrypt

✓ **Secure Authentication with JWT**

- Generating and Verifying Tokens
- Protecting Endpoints with JWT

✓ **Back-End Security Best Practices**

- Avoiding SQL Injection and XSS
- Secure Data Storage (Encryption)
- Using Environment Variables for Sensitive Data

Module 6: Advanced Topics

Asynchronous Programming

✓ **Introduction to Asynchronous Programming**

- Callbacks, Promises, and Async/Await
- Error Handling in Asynchronous Code

✓ **Working with APIs**

- Making HTTP Requests (e.g., Axios, Fetch)
- Consuming External APIs (e.g., Weather, Payment Gateways)

Real-Time Communication

✓ **Introduction to WebSockets**

- Understanding Real-Time Data Flow
- Setting Up WebSockets with Node.js

✓ **Using Libraries**

- Implementing Real-Time Chat with Socket.IO

Performance Optimization

✓ **Optimizing Back-End Applications**

- Reducing Latency with Efficient Queries
- Caching Strategies (e.g., Redis)
- Load Balancing Basics

✓ **Scalability**

- Horizontal vs. Vertical Scaling
- Introduction to Microservices

Module 7: Deployment and DevOps

✓ **Web Hosting**

- Deploying Back-End Applications on Platforms (e.g., Heroku, AWS, Vercel)
- Setting Up Domains and SSL Certificates

✓ **Version Control and CI/CD**

- Using Git and GitHub for Source Control
- Setting Up Continuous Integration and Deployment Pipelines

✓ **Containerization with Docker**

- Introduction to Docker and Docker Compose
- Creating and Running Containers for Node.js Apps