Intermediate: Backend Development

(Node.js or Python Track)

Description:

This course introduces students to robust backend development using **Node.js** or **Python**. Learners will explore how to build and manage backend servers, create RESTful APIs, implement authentication and security, and integrate with **relational** (MySQL, PostgreSQL) and **non-relational** (MongoDB) databases. The curriculum is hands-on and project-based, preparing students to develop scalable and secure server-side applications.

Learning Outcomes:

By the end of the course, learners will:

- Understand the fundamentals of backend/server-side programming.
- Build and deploy RESTful APIs for web and mobile clients.
- Implement secure authentication and authorization mechanisms.
- Connect and interact with MySQL, PostgreSQL, and MongoDB databases.
- Structure backend applications following best practices for scalability.
- Deploy and maintain backend servers with performance and security in mind.

Course Duration:

- **Total Duration:** 6 Weeks
- **Sessions per Week:** 2 sessions
- **Session Duration:** 1.5 hours per session
- **Total Contact Hours:** 18 hours (plus assignments and practicals)

6-Week Course Plan:

Week 1: Introduction to Backend & Environment Setup

Topics Covered:

- What is backend development?
- Overview of Node.js and Python for backend
- Environment setup with Express (Node.js) or Flask/FastAPI (Python)
- Creating a simple server

Deliverables:

- Basic server setup using chosen language
- First working API endpoint

Week 2: Routing, REST APIs & CRUD Operations

Topics Covered:

- REST architecture fundamentals
- Creating routes and handling HTTP methods
- Middleware and request handling

- CRUD operations using in-memory data

Deliverables:

- Functional REST API with basic CRUD routes
- Postman collection for testing

Week 3: Database Integration (SQL & NoSQL)

Topics Covered:

- Introduction to MySQL, PostgreSQL, and MongoDB
- Connecting backend to database
- Performing CRUD operations using database
- Intro to ORMs (e.g., Sequelize, SQLAlchemy, Mongoose)

Deliverables:

- API integrated with a real database
- ER diagram or schema representation

Week 4: Authentication & Authorization

Topics Covered:

- User authentication strategies: session vs token
- Implementing JWT (JSON Web Tokens)
- Role-based access control
- Hashing passwords with bcrypt or similar tools

Deliverables:

- User registration & login system

- Token-based authentication implementation

Week 5: Advanced Features & Error Handling

Topics Covered:

- Middleware for error handling
- File uploads and storage
- Sending emails or notifications
- Logging and monitoring basics

Deliverables:

- Backend features like file upload or notifications
- Complete error handling implemented

Week 6: Deployment & Final Project

Topics Covered:

- Project structure and best practices
- Deployment with Render, Heroku, Railway, or VPS
- Environment variables and configuration
- Final project presentation

Deliverables:

- Fully functional backend app
- Deployed API with documentation
- Final code on GitHub with README

Final Deliverables:

- Complete backend project (Node.js or Python)
- RESTful API documentation (Postman/Swagger)
- User authentication system
- GitHub repo with codebase and instructions
- Deployed API endpoint
- CodeHills Certificate of Completion

Course Fees (Pakistan):

- **Standard Fee:** PKR 15,000