urbancode
Training and Solutions

.Net + Angular
Full Stack Development

Become a Front-End / Full-Stack Developer

Course Curriculum

urbancode.in

98787 98797

Introduction to Full-Stack Development

- Overview of Front-End, Back-End, and Database
- Role of Angular, C#, .NET Core, and SQL Server in Full-Stack Development
- Setting Up the Development Environment: Visual Studio, SQL Server, Angular CLI, and Node.js

Front-End Development

Module-1: HTML, CSS, and Bootstrap

- HTML Fundamentals: Structure of an HTML Document, Semantic HTML5, Forms, Tables, and Multimedia
- CSS Basics: Selectors, Box Model, Flexbox, Grid Layout, Styling Forms, Animations, and Transitions
- Bootstrap: Grid System and Breakpoints, Components (Navbar, Cards, Buttons, Forms), Customizing Themes
- Responsive Design: Using Media Queries, Mobile-First Design Principles

Module-2: Angular Framework

- Introduction to Angular: SPA (Single Page Application) Overview, Setting up an Angular Project, Angular Architecture (Modules, Components, Templates)

▪ Forms in Angular: Template-Driven and Reactive Forms, Validation and Error Handling

▪ Angular Material: Material Design Components, Navigation, Dialogs, Tables, and Theme Customization

▪ Routing and Navigation: Setting up Routes and Lazy Loading, Router Guards

▪ CRUD Operations: Consuming REST APIs, Creating a CRUD Application with Angular and APIs

Back-End Development

Module-3: C# Programming

▪ Basics: Variables, Data Types, and Operators, Control Structures (If, Switch, Loops), Methods and Functions

▪ Object-Oriented Programming: Classes, Objects, and Constructors, Inheritance, Polymorphism, Abstraction, Interfaces

▪ Advanced C#: Delegates, Events, and Lambda Expressions, Exception Handling, LINQ (Language Integrated Query)

Module-4: .NET Core 8

▪ Introduction to .NET Core: Overview of .NET Core 8, Setting up a Web API Project in .NET Core

▪ RESTful APIs: Creating Controllers and Routes, Handling HTTPMethods (GET, POST, PUT, DELETE), Dependency Injection in .NET Core

▪ Middleware in .NET Core: Custom Middleware, Authentication and Authorization (JWT)

▪ Working with Configuration and Services

▪ Error Handling and Logging

Module-5: Entity Framework Core

▪ Introduction to Entity Framework Core: Code-First and Database-First Approaches, Setting up DbContext

▪ Entity Relationships: One-to-One, One-to-Many, and Many-to-Many, Navigational Properties

▪ Migrations: Creating and Applying Migrations, Updating the Database Schema

▪ CRUD Operations with Entity Framework Core: Adding, Updating, Deleting, and Retrieving Data

Module-6: SQL Server

▪ Introduction to SQL Server: Installing and Configuring SQL Server, SQL Server Management Studio (SSMS)

▪ SQL Basics: DDL, DML, and DQL Statements, Creating and Managing Tables, Writing Queries (SELECT, INSERT, UPDATE, DELETE)

▪ Advanced SQL: Joins (INNER, OUTER, LEFT, RIGHT), Subqueries and Common Table Expressions (CTEs), Stored Procedures and Functions, Triggers and Indexing

▪ Database Design: Normalization and Denormalization, Designing Relational Databases, Creating Relationships with Primary and Foreign Keys

Module-7: Full-Stack Integration

▪ Building a Complete Full-Stack Application: Example (E-Commerce or Task Management System)

▪ Integration: Using Angular to Consume .NET Core APIs, Persisting Data in SQL Server via Entity Framework

▪ Authentication and Authorization: JWT-Based Authentication in .NET Core, Protecting Routes in Angular

▪ Real-Time Updates: SignalR Integration (optional)

Module-8: Deployment

▪ Deploying the Application: Hosting Angular on Firebase/Netlify, Hosting .NET Core APIs on Azure or IIS

▪ Database Management: Backing up and Restoring Databases in SQL Server

Module-9: Capstone Project

▪ Develop a Full-Stack Application integrating Angular,

.NET Core, Entity Framework, and SQL Server.

▪ Example Projects: Online Store, Employee Management System, Learning Management System

Course Duration

Total Duration: 16 to 20 weeks

Front-End: 6-8 weeks

Back-End (C#, .NET Core, Entity Framework): 6-8 weeks

Database (SQL Server): 3-4 weeks

Project Development: 2-4 weeks