



FUTURA LABS

ENABLE • TRANSFORM • ACCELERATE

ASP.NET-REACT

INTERNSHIP SYLLABUS

Time Period : 6 Months

About Us

At the heart of Futura Labs is a commitment to bridge the gap between ambition and expertise. In an ever-evolving technological ecosystem, staying ahead requires not just theoretical knowledge but practical, hands-on experience. We strive to be the catalyst that transforms ambitious individuals into industry-ready professionals.

Why Choose Futura Labs?

Expert Faculty: Our team of seasoned professionals and industry experts serve as mentors, guiding students through the latest trends and best practices.

Cutting-Edge Curriculum: Our meticulously crafted curriculum reflects the current industry demands, ensuring that our students graduate with skills that are relevant and in high demand.

State-of-the-Art Facilities: Futura Labs boasts state-of-the-art training facilities equipped with the latest tools and technologies, providing an immersive learning experience.

Hands-On Projects: Practical application is key to mastery. We prioritize hands-on projects that simulate real-world scenarios, allowing students to build a robust portfolio before entering the job market.

Placement Assistance: Our commitment doesn't end with the completion of the course. We provide dedicated placement assistance, connecting our graduates with top-tier companies. Whether you're a budding developer, a creative designer, or a digital marketing enthusiast, Futura Labs is your gateway to a future-proof career. We invite you to embark on a transformative learning journey with us, where innovation meets education, and potential meets opportunity.

Futura Labs - Shaping Tomorrow's Technological Leaders.

1.Introduction to programming

- Algorithm
- Programming concepts
- Object oriented programming and concepts

2.Git -Version Controlling

3. Introduction to HTML

- Introduction and structure of HTML
- HTML Elements, Tags and Attributes
- HTML Image
- HTML Link
- HTML Div and Span
- HTML List
- HTML Table
- HTML Form
- HTML Media
- Iframe
- HTML5 Semantic Elements

4. Introduction to CSS

- Introduction of CSS
- CSS Syntax
- CSS Selectors
- Styling methods in CSS
- CSS Properties
- Marquee
- Media Query

5. Advanced CSS

- CSS Gradients
- CSS Flex Property
- CSS Display
- Transitions, Transformation, Animations
- Box Sizing
- Grid

6.Bootstrap

- Introduction to Bootstrap
- Bootstrap Containers and Components
- Tables, Images, Colors
- Alerts, Buttons
- Spinners, Cards, Pagination, Carousel, Navbar and Modal
- Bootstrap grid systems
- Bootstrap grid system-Advanced

7.JavaScript

- Introduction to JavaScript
- Creating Variables in JavaScript
- Creating Functions in JavaScript
- Returning Data from Functions
- Working with Conditions
- Looping in JavaScript
- Working with Strings and Arrays

7.1 Manipulating DOM Elements with JavaScript

- Accessing DOM element using “classes”
- Accessing Elements using “id”
- Manipulating DOM Elements
- Updating DOM styling with JavaScript
- Template Literals
- Query Selector

7.2 OOPs Concept in JavaScript

- Class Implementation in JS
- Encapsulation in JS
- Polymorphism in JS
- Inheritance in JS

7.3 Working with Modern JavaScript

- Working with ES6
- Arrow Functions
- Comparing Arrow Function with Normal Function
- Callback Functions
- Array Map
- Rest and Spread Operators
- Promises
- Working with Async and Await
- Form Validation with JavaScript
- Concept of Json
- Closure and Function Currying
- Map and Set
- Error Handling
- Hoisting
- Memoization and Recursion

8. Working with React Basics

8.1 Introduction to React

- Introduction To React
- Comparing React with other Frameworks
- Working with Virtual DOMs

- Explanation with component-based Architecture
- Concept of Single Page Application
- Features and Limitations of React JS
- Setting Up React Development Environment

8.2 Working with Functional Components

- Defining a Component
- Working with Child Components
- Nested usages of Components
- Comparing Functional component with Class components
- Component Rendering
- Understanding JSX

8.3 Styling React Component

- Inline Internal and External styling
- React Bootstrap concept and implementation
- Array Mapping

8.4 Introduction to Hook

- Understanding concept of Hook
- Adding State to the Component
- Updating Component State

8.5 Working with Child Components

- Adding Child Component to Parent Components
- Passing Data to Child Components
- Updating Data in Child Components

8.6 Introduction to useContext Hook

- Concept of Props Drilling
- Introduction to useContext Hook
- Passing Data to Multiple Components
- Updating the Parent Data from Child Component

8.7 Working with Data in Functional Component

- Invoking Functions in React Component
- Raising Event with React Components
- Working With Spread and Rest Operators
- Working with Conditional Operators
- Concept of Destructuring in React JS

8.8 Introduction to useEffect Hook

- Concept of useEffect Hook
- How to handle side effects in React JS
- API Integration using use Effect hook

8.9 Working with Routing in React

- Understanding the dynamics of Single Page Architecture
- Creating Single Page architecture with React Routers
- Defining React Router Dom installation
- Concept of Browser Router, Routes and Route
- Nested Routing in React
- Creating Custom Routes and Links
- Concept of useParams and useNavigate
- React Router V6 and its features

8.10 Working with Forms

- Capturing Events from Forms
- Updating Forms Data
- Working with TextArea, Inputs, etc
- Introduction to Form Management libraries

8.11 Working with Redux

- What is Redux
- Installing Redux
- Why use Redux for state management
- Redux principles: Actions, Reducers, Store
- Data flow in Redux
- Creating selectors to retrieve data from the Redux store

9. Create a react project using your concepts and themes

.NET Framework

- CLR, CLS, CTS
- Compilation Process in Dot net
- Assemblies and Versioning

C#

- Introduction to C#
- Identifiers
- Keywords
- Operators
- Data Types
- Access Modifiers
- Functions, Property
- Collection Classes
- Math Function
- String Functions
- lambda Functions

MSSQL

- Introduction and Installation SSMS
- Database
- CREATE,ALTER,DELETE,DROP tables
- INSERT,UPDATE,DELETE table data
- WHERE Condition
- Aggregate Functions
- AND, OR, IN ,BETWEEN, LIKE, DISTINCT,ORDER BY, GROUP BY, UNION
- Sub-Queries
- LEFT JOIN, RIGHT JOIN, INNER JOIN

ADO.NET

- Overview of ADO.NET
- Connected vs Disconnected Architecture
- Data Connection Object
- Data Command Object
- Data Adapter Object
- Data Readers
- Data sets and Data Adapters
- Structure of Dataset
- Execute Non-query
- Execute Reader
- Execute Scaler

Session

- Session- Introduction

Uploading Files

- Using file upload control

Forms

- Introduction to ASP .NET Web Forms,ASP .NET Controls
- Master Pages and Themes
- Simple master page and Nested master page
- Configuring master page Creating themes Applying themes

Advanced Programming with Layered Architecture

- Two Tier Architecture
- Three Tier Architecture
- SQL Stored Procedure(Sample Application)

PROJECT

ASP.NET Core MVC

- Introduction to ASP.NET Core MVC Framework
- How to Set up MVC in ASP.NET Core
- AddController vs Add Mvc vs AddControllers With Views vs Add Razor Pages
- Models
- Controllers
- Views
- ASP.NET Core Dependency Injection
- Creating ASP.NET Core Application using MVC Template
- ViewData
- ViewBag
- Strongly Typed View
- ViewModel
- Routing
- How to Use Bootstrap in ASP.NET Core MVC
- Tag Helpers

Entity Framework Core

- Introduction
- How to Install Entity Framework Core
- DbContext in Entity Framework Core
- Migration in Entity Framework Core
- Seed Data
- Validation
- Authorization and Authentication
- Scaffolding
- File Upload
- Identity login
- Bootswatch Theme

API

- Introduction to web API
- Swagger

React and .Net Integration

PROJECT

Thank You

Get in touch with Us

Calicut:

+91 7994420045

+91 7994420046

Kochi:

+91 9778043002

+91 9946326888