.NET FULLSTACK

CURRICULUM

Introduction of Dot Net Framework
- What is Dot Net?
- Why Dot Net?
- Advantages of Dot Net
- Component of Dot Net
- What is Framework?
- Versions of Framework
- CLR architecture
- CTS architecture
- CLS architecture
- DLR architecture
- What is JIT?
- Types of JIT?
  - Pre-JIT COMPILER
  - Econo-JIT COMPILER
  - Normal-JIT COMPILER

Getting Started with Visual Studio
- Understanding Project and Solution Explorer
- Creating a Project and Solution
- Understanding Solution Explorer
- Comments in C#
- Debug Project
- Understanding Execution Process

Assemblies in .Net
- What is Assembly?
- Types of Assembly?
- How to make Private Assembly
- How to make Shared Assembly
- What is DLL HALL Problem?
- What is SN Key?
- What is GAC?
- How to Deploy Assembly in Cache
- What is MSIL?
- What is DLL & EXE?
- How make DLL& EXE.
- Use of Ilasm.exe (IL Assembler) & Use of Ildasm.exe (IL Disassembler).

Object Oriented Programming System (OOPS)
- What is OOPS?
- What is Class & Object?
- Difference b/w Class, struct, interface and enum
- Types of Variable

- Static Variable
- Const Variable
- Read Only Variable
- Instance & Static Members
- Inheritance & Types of Inheritance
- What is Interface?
- Encapsulation
- Abstraction
- Polymorphism
- Types of Polymorphism
  - Static Polymorphism
  - Dynamic Polymorphism
  - Concept of Method Overloading
  - Concept of Method Overriding
  - Concept of Early Binding
  - Concept of Late Binding
  - Concept of Up Casting
- Operator Overloading?
- What is Constructor?
- Types of Constructor
  - Default Constructor
  - Parameterized Constructor
  - Copy Constructor
  - Private Constructor
  - Static Constructor
  - Concept of This Keyword in Constructor
- Concept of Base Keyword in Constructor
  - Concept of Base Keyword in Constructor
  - Abstract Class & Abstract Method
  - Static Class & Static Method
  - Sealed Class & Sealed Method
  - Partial Class & Partial Method
- Different types of Parameter
  - Input Type
  - Output Type
  - Ref Type
  - Params Type
  - Dynamic Type

C# LANGUAGE FUNDAMENTALS

Introductions of C#
- What is C#?
- Namespaces in C#?
- Create namespace in C#
- Data types in C#
- Value type
- Reference type
- Null able type
- Heap
- Stack
- Looping in C#
  - For,While,Dowhile
  - Foreach
- Conditional statement
  - Switch case
  - If, if-else, Nested if
  - GoTo
  - Break

- Continue
- Convertion Types
  - Implicit {Boxing}
  - Explicit {UnBoxing}

Array
- What is Array?
- Types of Array
  - Simple Array
  - Multi Array
  - Jagged Array

File Handling
- Introductions of File Handing
- Working with Files and Directories
- Working with Directories
- Working with Files Using the File and FileInfo Classes
- Working with Directories Using the Directories and Directories Info Classes

Exception Handling
- Introductions of Exception Handling
- Role of Exception Handling
- What is Error, bug and exceptions?
- Try, catch & Finally block
- System Level & Application Level Exception
- Handel Multiple Exception
- Throw Exception

Software Engineering Models and Concept (Project Based)
- What is Software Development Life Cycle?
- Phases of a SDLC
- Project initiation and planning
- Feasibility study
- System design
- Coding
- Testing
- Implementation
- Maintenance

Collections
- Introductions of Collections
- Types of collections
- Generic
  - List
  - Dictionary
  - Stack
  - Queue
- Non Generic
  - ArrayList
  - HashTable
  - Stack
  - Queue

Delegate
- Introductions of Delegate
- Types of delegate
- SingleCast
- MultiCast
- Generic delegate

Events
- Introductions of Events
- Understanding Events and Event Handler
- Types of Event
- System Define Event
- User Define Event
- Event Life Cycle
- Binding Events
- Working with Event Handler

Threading
- Introductions of Threading
- Threading Namespace
- Understanding Thread LifeCycle
- Default Threading
- Multi Threading
- Priority in threading Member Info

Introductions of Properties
- set
- get

Indexer
- Define Indexer
- Advantages of Indexer

Business Requirement Specification (Project Based)
- Objective of Project
- DFD
- UML Diagram
- Class Diagram
- ER Diagram

The SQL Server Management
- What is SQL?
- What is Database?
- Create Statement
- Create database
- Create table
- Use Identity
- Select Statement
- Insert statement
- Update statement
- Delete statement
- Drop statement
- Truncate statement
- Difference between delete, drop & truncate
- Alter Query
- Clauses
- Group By
- Order By
- Basic Queries
- Stored Procedure
- Use Input and Output Type parameter in Stored Procedure
- Function
- Types of Functions
- Trigger
- Cursor
- Union & Union All

- Transaction
- Joins
- Indexes
- Views
- Constraints
- Tracing sql query with SqlProfiler

System Requirement Database Structure (Project Based)
- Database Table Structure
- Database Design
- DDL
- DML
- Stored Procedure
- Database Security

ADO .NET 10
- What is ADO.NET?
- Difference between ADO & ADO.NET?
- Architecture of ADO.NET
- Disconnected mode Architecture
  - Using SqlDataAdapter Class
  - Using DataSet Class
  - SqlCommandBuilder Class
- Connected Mode Architecture
  - Using SqlConnection Class
  - Using SqlCommand Class
  - Using SqlDataReader Class
  - Understanding ExecuteNonQuery method
  - Understanding ExecuteScalar method
  - Uunderstanding ExecuteReader method
- Working with DataTable class
- Creating PrimaryKey Column on VirtualTable
- Using Constraints Class
  - ForeignKeyConstraints Class
  - UniqueConstraints Class
- Store table in datatable
- Apply Insert,update,delete operations on VirtualTable
- Show, next,first,last,all by datatable
- Executing Stored Procedure
- Executing Function
- What is Sql Injection
- Applying SqlInjection

ASP.NET 4.7 HTML 5
- Intruduction
- HTML tag
- CSS
- Create landing page
- Table format
- Iframe
- Bootstrap

ASP.NET 4.6 Framework
- Asp.Net Introduction
  - Evolution of Web Development
  - ASP .NET different versions
  - Understanding protocol and domain
  - Understanding static & dynamic page

- Server-side programming
- Client-side programming
- HTML and HTML Forms
- Creating a simple html page
- Developing ASP.NET Applications
  - Creating Websites
  - Creating a New Web Application
  - Page Life Cycle
  - Understanding Events of Page Life Cycle
  - Websites and Web Projects
  - Adding Web Forms
  - Migrating a Website from a Previous Version
  - Difference b/w inline and code behind concept
  - Asp.Net special directory
  - What is Application Directories
  - Different types of Application Directories
  - Understanding application label and page label variable
  - Need of Global class
  - What is Configuration file?
  - Need of Configuration file
  - Difference b/t app.config & web.con file
  - Nested Configuration
- State Management
  - Need of state management technique
  - Types of state management
  - Understanding Server side and Client side state management
  - ViewState
  - HiddenField
  - Cookies
  - Cache
  - QueryString
  - Session
  - Session Tracking
  - Session Configuration
  - Session mode
  - Cookieless
  - Session Timeout
  - Session Mode
  - Application
  - Transferring Information between Pages
  - Transferring Information between Cross Pages
- Web Controls in ASP.Net
  - Types of web controls
  - Server Controls
  - Standard Controls
  - Data Controls
  - Validation Controls
  - Navigation Controls
  - Login Controls
  - Ajax Extensions
  - Ajax ToolKit
- Html Controls
- User Controls
- Data Controls
  - DataList
  - DetailsView
  - FormsView
  - GridView
  - ListView
  - Repeater

- DataGrid
- EntityDataSource
- Object DataSource
- LinqDataSource
- XmlDataSource
- Working with Templates of Data Controls
- Working with events of Data Controls
- CSS
  - Understanding CSS
  - Types of CSS
  - Inline CSS
  - Internal CSS
  - External CSS
  - Selectors in CSS
- Themes & Skins
- Working with Image gallery
- Working with ImageSlide
- Master Page
- Nested Master Page
- Page Tracing
- Mail & Messaging
- Caching Techniques
  - Understanding Caching
  - Types of Caching
  - Output Caching
  - Data Caching
- Web Security
  - Need of security
  - Types of security
  - Forms Authentication
  - Windows Authentication
  - Confidentiality with SSL
  - Password Format
  - Types of password format
  - MD5
  - SHA1
  - Clear
- Performance tuning
- Ajax
- Understanding Gmail,Facebook and Twitter API's
- Deploying ASP.NET Applications
  - ASP.NET Applications and the Web Server
  - How Web Servers Work
  - Understanding Virtual Directory
  - Web Application URLs
  - Web Farms
- Internet Information Services (IIS)
  - Installing IIS
  - Registering the ASP.NET File Mappings
  - Verifying That ASP.NET Is Correctly Installed
- Managing Websites with IIS Manager
  - Creating a Virtual Directory
  - Configuring a Virtual Directory
- Deploying a Simple Site
  - Web Applications and Components
  - Other Configuration Steps
  - Code Compilation
  - Verifying That ASP.NET Is Correctly Installed
- Deploying a Visual Studio IDE
  - Creating a Virtual Directory for a New Project

- Copying a Website
- Publishing a Website
- Verifying That ASP.NET Is Correctly Installed

Technical Design & Development (Project Based)
- Working with Project
- Programming Language: C# (Asp.Net)
- Designing Tools
- HTML
- Using CSS
- Using Ajax
- Using JavaScript

ADVANCE .NET MVC

ASP.NET MVC
- Introduction to ASP.NET MVC
- Brief history of ASP.NET MVC
- Understanding ASP.NET MVC Design Pattern
- Setup & Installation process
- Compare ASP.Net Web Forms with ASP.NET MVC
- Life cycle of ASP.NET MVC
- Key Benefits of ASP.NET MVC
- Understanding ASP.NET MVC project templates

View Engine
- What is View Engine
- Role of View Engine
- Types of View Engine
- Compare ASP View & Razor View Engine
- Understanding Razor syntax
- Accessing model data in Razor views
- Working with Razor Layouts
- Understanding Razor Expressions

Models
- What is MVC
- Understanding scaffolding template
- Scaffolding template with Entity Framework
- Defining a data model
- Understanding Model Binding
- Working with explicit model binding
- Responding to the edit post request

Views
- Purpose of Views
- Understanding View Conventions
- Types of View
- Understanding ViewBag, TempData,ViewData and ViewDataDictionary
- Creating strongly-typed views
- Adding a View
- Using Razor syntax in View
- Code Block in View
- Working with Partial View
- Introduction of Layout/Master page
- Working with Layout

Controllers and Actions
- What is the Controller
- Use of Controller
- What is Action and Action Results
- Action Filters
- Authorize Filters
- Result Filters
- Built In Filters
- Processing Input and Output Action

Membership, Authorization And Security
- Session and application state
- Using the Authorize Attribute to Login
- Securing Controller Action
- Authorize Attribute with Forms Authentication
- Windows Authentication
- Using Role wise Login
- Managing users and roles
- Configuring OAuth And OpenID Providers
- Login via OAuth and OpenID
- Security Implications of External Logins

URL Routing
- Introducing URL Routing
- Routing Approaches
- Defining Attribute and Traditional Routes
- Understanding Route Parameters
- A Details look at URL Generation
- Creating and Registering a Simple Route
- Ambient Route Values
- Using Static URL Segments
- Constraining Routes
- Bypassing the Routing System

Forms And Helper Methods
- Using Form Helper Methods
- The Action and the Methods
- Using GET or POST Method
- Using Template Helper Methods
- Using Model Metadata
- Making Helpers Do your Bidding Inside HTML Helpers
- Setting up the Edit Forms
- Using Strongly Typed Helpers
- Helpers and Model Metadata
- Helpers and Model State
- Html.Hidden
- Html.Password
- Html. RadioButton
- Html.CheckBox
- Html.DropDownList
- Html.Partial and Html.RenderPartial
- Html.Action and Html.RenderAction

Data Annotations And Validation
- Using Validation Annotations
- Custom Error Messages and Localization
- Using Range validator Attribute
- Regular Expression validator Attribute
- Required validator Attribute

- String Length validator Attribute
- Custom Validation and Annotation
- Using display format

Additional Techniques
- View scaffold templates
- Model Validation
- Global filters
- Model binding
- Bundling and Display Modes

Using Ajax, jQuery with ASP.NET MVC
- Overview of JQuery
- Unobtrusive Ajax and the 'this' context
- Using Ajax action links
- Using Ajax Helpers
- Autocomplete with jQuery UI
- JSON & Client-Side Template
- Bootstrap Plug-in

Language Integrated Query (LINQ)
- Introduction to LINQ
  - What is LINQ
  - Need of LINQ
  - How LINQ Works
  - Using Linq via Method based and Query based syntax
- Standard Linq Operators
  - Sorting Data
  - Filtering Data
  - Joining Data
  - Partitioning Data
- Linq Provider
  - LINQ to Object
  - LINQ to XML
  - LINQ to SQL
- Understanding Lazy Loading and Eager Loading
- Difference B/w Lazy and Eager Loading
- Using Stored Procedure in LINQ
- Working with Single table Inheritance
- Using of SQLMetal

Entity Framework
- Overview of the Entity Framework
- Difference b/w Linq to Sql and Entity Framework
- Using Schema First Approach
- Using Model First Aproach
- Using Code First Aproach
- Using Stored Procedure in Code First Approach
- Using Sql Profiler with Stored Procedure
- Creating/Managing one to many and many to one relations
- Using Table, Column and Key attributes
- Associations & Foreign Keys

WindowsCommunicationFoundation (WCF)
- IntroductionToWCF
- What is Distributed Application
- What is WCF
- Evaluation of WCF
- Understanding WCF Architecture

- WCF comparison with WebServices
- WCF comparison with Remoting
- Benefits of WCF Service
- Understanding WCF Templates
  - WCF Service Library
  - WCF Service Application
  - WCF Workflow Service Application

Endpoints in WCF
- What is EndPoint in WCF
- End points seetings in app. Config & web.config file
- ServiceMetaData HTTP_Get enabled
- IMetaDataExchange Contract/Endpoint for Http and Tcp
- What is Address-Binding-Contract (ABC)
- **Address**
  - What is Address
  - Understanding improtance of address
  - Difference between HTTP & HTTPS address
  - Use of TCP and MSMQ address
  - What is Named Pipe address
  - Understanding IIS address
- **Binding**
  - Understanding WCF Bindings
  - Types of bindings
  - Basic Binding
  - Web Binding
  - WS Binding
  - WS Dual Binding
  - TCP Binding
  - IPC Binding
  - MSMQ Binding
- **Contract**
  - What is the of contract
  - Types of Contract
  - Service Contract
  - Operation Contract
  - Data Contract
  - Fault Contract
  - Message Contract

Developing WCF Service Application
- Hosting WCF Service in IIS/ASP.NET Development Server
- Hosting WCF Service in Managed Application(Self Hosting)
- Hosting WCF Service in Window Service
- WCF service durable service
- Use of WCF service Request-Replay
- Use of WCF service One-Way
- Use of WCF Callback service

Angular

Introduction to TypeScript
- Compiling TypeScript
- Why Use TypeScript
- Let and Const
- Lambda/Arrow Functions
- Basic Types
- Type Assertion
- Type Definitions
- Classes and Interfaces
- Constructor

Environment Setup

- Node / NPM
- Angular CLI
- Application File Structure
- Code Editors

Introduction to Angular

- What is Angular?
- Advantages Of Angular
- Disadvantages Of Angular
- Key Features of Angular
- Comparison Between Angular with Other Frameworks.

Modules

- Why use Modules
- NgModule
- Declarations
- Providers
- Imports
- Bootstrapping
- The Core Module
- Shared Modules
- LazyLoading

Components

- Introduction to Components
- Component Architecture Patterns
- Decorator Metadata
- State &Behaviour

Templates

- Inline vs External
- Template Expressions
- Data Bindings
- *ngIf else, *ngFor and *ngSwitch
- Built-in Structural Directives
- Built-in Attribute Directives

Custom Directives

- Create your own Structural Directive
- Create your own Attribute Directive

Pipes

- Built-in Pipes
- Custom Pipes

Services

- Introduction to Services
- Building a Service

Lifecycle Hooks

- Component LifeCycle
- All lifecycle Hooks

Routing

- The Component Router
- Defining Routes
- Navigation
- Route Params
- Child Routes
- Primary Route and Named Route
- Multiple routerOutlet

Advanced Routing

- Route Guard

Template-driven Forms

- Introduction to forms
- Template-driven forms
- Validation

Model-driven Forms

- Introduction to Reactive forms
- FormGroup&FormControl
- Validators
- Custom Validation

Asynchronous Operations

- Introduction to Async
- Observables
- Observables vs Promice
- HTTP Request / Response

Communication Between Component

- @Input()
- @Output()
- @ViewChild()
- @ViewChildren()
- Shared Service

Angular Testing

- Introduction to Testing
- Unit Testing

Technical Design & Development (Project Based)

- Working with Project
- Programming Language: C# (MVC Razor)
- Designing Tools
- CSHTML
- Using CSS
- Using Ajax
- Using JavaScript