FullStack Project

Full-Stack development, *a full-stack developer should know clients as well as Server-side technologies. With the fast pace of technology, a full-stack developer must know the following technologies along with HTML, CSS and Javascript.*

1. ***Client-Side*** *: Front-End*
   - 1.1. *HTML, HTML5,*
   - 1.2. *CSS, CSS 3,*
   - 1.3. *Bootstrap,*
   - 1.4. *Javascript*
   - 1.5. *JQuery*
   - 1.6. *AJAX*
   - 1.7. *Angular*
2. ***Server-side****: Back-End*
   - 2.1. *Java Basics*
   - 2.2. *Java OOPs*
   - 2.3. *Core Concepts*
   - 2.4. *JDBC*
   - 2.5. *jsp/servlet*
3. ***Database****: Back-End*
   - 3.1. *MongoDB*

Front-End Development of Project ***25Hr***

==========

1. HTML and HTML 5 ***1Hr***

- *Introduction of HTML, Tag, Elements and Attributes, Basics syntax,*
- *Table, List, Forms, Structure of HTML4 and HTML5,*
- *Semantic and non-semantic tags, HTML 5 Features, New Input type*

2. CSS (Cascading Style Sheet) and CSS 3 ***1Hr***

- *Attributes (ID, Class, Style, Title), CSS Types (Inline, Internal, External)*
- *Box-model, Display Property (Block, Inline, None), Visibility-Hidden,*
- *Position Property(Static, Relative, Absolute, Fixed), Z-index Property*
- *Combinators (Descendant Selector, Child Selector, Adjacent Sibling Selector*

4. Bootstrap *1Hr*

- *Introduction of Bootstrap (Responsive), Typography, Tables, Images,*
- *Buttons, Grid Structure-Type of columns, Forms, Jumbotron, well,*
- *The panel, Navbar, Nav Tab, Carousel, Responsive Web Page*

5. JavaScript *1Hr*

- *Introduction of JavaScript, Use of JavaScript, Variables, Keywords*
- *Data Type (Primitive, non-primitive), JS Conditions (if, if-else)*
- *Conditional operators & logical operators, Loops (for, while, do-while)*
- *Switch Case, Functions, SetTimeout and set Interval Function*
- *HTML DOM, Use the document object to access and manipulate HTML*

6. jQuery and AJAX *1Hr*

7. Angular

Angular: Introduction and environment set up. *1Hr*

In this module, we will learn what is angular, what are the major differences between AngularJs and Angular.

We will also learn how to set up in the local development environment and a small introduction about typescript.

- *What are Typescript and ECMAScript?*
- *What is Angular?*
- *What is the framework?*
- *The need for MVC, MVVM, MVW and MV* Architecture in Web Application*

- *What is Components-Based Web Development? And the benefits.*
- *AngularJs (vs) Angular*
- *Setup for the local development environment*
- *Angular files and folder structure with JSON configuration*
- *Role of Node JS and NPM in Angular*
- *What is CLI? Angular CLI (Command Line Interface) Commands*
- *Introduction Of Example Project*

Angular: Start First Angular S. P. A. from basic *2Hr*

In this module, we will learn the workflow of an Angular application.

  - *Execution of angular.json and package.json*
  - *Linking between all project files in MVC architecture*
  - *Creating our first element and typescript*
  - *Selector: customize or ignore tag(s)*
  - *Decorators and Metadata*
  - *Import and imports array,*
  - *Component and @Component, NgModule and @NgModule,*
  - *bootstrap, Selector and template,*
  - *Backticks and coding std in the template (ES2015 feature)*
  - *templateUrl, styles array, and the styleUrls array*
  - *Declaration array, class,*
  - *BrowserModule and bootsrapModule etc.*
  - *Launching the application.*
  - *Role of the Module and Components.*
- *Splitting of Module and Component.*

- *Exporting in Angular.*

Angular: Directives ***1Hr***

In this module, we will learn what the built-in directives that we mostly use in angular.

- *Structural directives*
  - *Built-in Directives*
    - *ngIF, ngFor, ngSwitch*
  - *Style and Class Directives*
    - *ngClass, [class.clsName]*
    - *ngStyle, [style.stlName]*
- *Attribute directives*
  - *Customise Directive*
- *Component: Way to Create, Split and reuse it.*
- *Host Listener and Host Binding*

Angular: Data binding ***2Hr***

In this module, we will learn how to do data-binding in Angular

- *Interpolation*
- *Property binding*
- *Event binding*
- *Two-way Binding*
- *Class binding*
- *Style binding*

- *Methods*

Angular: Components *1Hr*

In this module, we will learn one of the major concepts in Angular 4.0 i.e., components. And we can also learn how to

- *create a dynamic component (without a separate component file) using @Component.*
- *What are the components?*
- *Understanding Components lifecycle hooks*
- *Creating a component with CLI*
- *Split an Angular application using components to make Angular application lightweight and high performance.*

Angular: Modules *1Hr*

In this module, we will learn what modules are, and the usage of modules in a different manner.

- *Root App module*
- *Getting more Object-Oriented:*
    - *Create a Model for data (validating data)*
    - *Classes – Properties, Methods, Constructors, Inheritance*
    - *Exporting a model*
    - *Mock data model (as the Angular team prefers)*

Angular: View *2Hr*

In this module, we will learn the angular's view section,

- *Implementing style:*
  - *inline style,*
  - *internal style, and*
  - *external style file*
- *Splitting view files*
- *CSS style Scope*

Angular: Forms ***2Hr***

In this module, we will learn forms-module in angular i.e.

- *Forms in Angular*
- *Template Driven Forms*
- *Reactive*

Angular: Pipes ***1Hr***

In this module, we will learn what pipes are, and how to pass parameters to pipes, and how to create custom pipes.

- *Why are pipes useful?*
- *Built-in pipes*
- *Parameterizing pipes*
- *Custom pipes*

Angular: Services & Dependency injection ***2Hr***

In this module, we will learn what services are and how do they communicate using the HTTP protocol to the server.

- *Creating Service*
- *$http Service*
- *Introduction to Injectors (Dependency Injection)*
- *Providers: use and implementation.*

Angular: Routing

2Hr

In this module, we will learn the introduction for routing in angular and how to navigate between views, how to do

- *parameterized routing.*
- *Introduction*
- *Configuring & Navigating*
- *Parameterized routes*

Angular: Operations Using Http Service ***2Hr***

In this module we will create an application with an end-to-end start from the server to the client, getting response and

- *requests using HTTP service.*
- *Creating Services*
- *Creating Components*
- *Creating Routings*
- *Configuring NgModule*
- *Working with JSON Data file*
- *Run the application*

Angular: Deployment of an optimize app product ***1Hr***

Start with the development build

- *Deploy on FTP web server*
- *Deployment on Google firebase web hosting service*
- *Build an application as a product with a specific location*
- *Build an application as a product in an optimized way*

Back-End Development Content ***25 Hour***

==========

Java Basics: ***3 hours***

- *(Comments, Variables,*
- *Data types,*
- *Keywords,*
- *conditional statement,*
- *loops and methods)*

Java OOPS: ***3 hours***

- *(class, objects,*
- *constructors,*
- *abstraction,*
- *encapsulation,*
- *inheritance,*
- *polymorphism,*
- *Overloading and overriding)*

Core Concepts: ***7 hours***

- *(Arrays, Strings,*
- *Exception Handling,*

- *Threading,*
- *Synchronization,*
- *IO Streams,*
- *Serialization,*
- *Collections and Generics)*

MongoDB: ***3 hours***

- *(Installation,*
- *Database Creation*
- *Create Collection,*
- *Insert, Retrieve,*
- *Update, Delete,*
- *Sorting, Filtering,*
- *Managing database.)*

JDBC: "Java Database connectivity": ***2 hours***

- *(About drivers,*
- *Driver, Connection,*
- *Statement,*
- *PreparedStatement,*
- *ResultSet and Methods)*

JSP/Servlets: ***7 hours***

- *(Web Components,*
- *Request-Response Model,*
- *Server/Container,*
- *Servlet/JSP Request and Response,*
- *Servlet/JSP Collaboration with Html Forms,*
- *Implementation of **JSP MVC**,*
- ***Servlet/JSP** integration with **Front-end**,*
- ***Servlet/JSP** integration **with MongoDB**,*
- ***Deploying Application** on Server(**Apache/JBoss**).*

==========

Real Time Website Development Training : **Duration = 50 hours**