UI Developer:

User Interface (**Ui**) **Design** Is The **Process Designers** Use To **Build Interfaces**, **Focusing** On **Looks** Or **Style** Of Your **Client's Web Site** Or **Web Applications**. **Ui** Or **Web Designers** Aim To **Create Interfaces** Which **Users** Find **Easy To Use** Of **Websites** Or **Web-Applications** And **Pleasurable**.

Front-End Web Development Is The **Practice** Of **Converting Data** To A **Graphical Interface**, Through The **Use Of Html, Css, Javascript** And **Applying High-End Functionalities** Either With **Frameworks** Or **Libraries** Of **Javascript**, So That **Users Can View** And **Interact With** That **Data And Get Responses** Or **Results.**

In This Course, We Will Learn How We Can **Create Templates** Or **Designs** Of **Websites** Or **Web-Applications**, And **Front-End Web Development**, Where **Users Can Interact** With Your **Applications** With In **Less Time In Getting Results** Much Faster.

Angular Front End:

Angular is a **TypeScript-based** open-source **web application framework** led by the **Angular Team** at **Google** and by a **community of individuals** and **corporations**. **Angular** is a **complete rewrite** from the **same team** that **built AngularJS**.

AngularJS is a **structural framework** for **dynamic web apps**. It **lets you** use **HTML** as your **template language** and **lets** you extend **HTML's syntax** to **express** your **application's components clearly** and **succinctly**. AngularJS's **data binding** and **dependency injection** eliminate **much of the code** you would otherwise have to write

We Will Discuss, Contents Or Technologies:

Ui Or Web Design:

    Html, Css, Bootstrap

Front-End (Programming Web Development):

    Javascript, Jquery, Ajax & Json

Front-End (Application's Development):

    Typescript, Angularjs

Html

- ❖ **Html Introduction And Its Use In Websites**
- ❖ **Introduction Of Html**
- ❖ **Differences Between Html4 & Html5**
- ❖ **Elements / Tags**
- ❖ **Attributes And With Practical Examples**
- ❖ **Forms**
- ❖ **Tables**
- ❖ **Audio And Video**
- ❖ **Html5 New Elements**

Css

- ❖ **Css Introduction And Its Use In Websites**
- ❖ **Css Implementations In Html**
- ❖ **Animation Properties**
- ❖ **Background Properties**
- ❖ **Text Properties**
- ❖ **Layout Properties**
- ❖ **Selectors**
- ❖ **Elements**
- ❖ **Attributes**
- ❖ **Psuedo Classes**
- ❖ **Animation Effects**

Bootstrap

- ❖ **Bootstrap Basic**
- ❖ **Fixed Layouts**
- ❖ **Responsive Layouts**
- ❖ **Tables**
- ❖ **Images**
- ❖ **Buttons**
- ❖ **Forms**
- ❖ **Glypicons**
- ❖ **Navigations**
- ❖ **Breadcrumbs**
- ❖ **Dropdowns**
- ❖ **Paginations**

- ❖ **Helper Classes**
- ❖ **Modal Dialog Boxes**
- ❖ **Carousels**

Javascript

- ❖ **Introduction**
- ❖ **Basic Concepts**
- ❖ **Conditional Statements**
- ❖ **Loop Statements**
- ❖ **Functions**
- ❖ **Dom And Event Handlers**
- ❖ **Working With Forms**
- ❖ **Object Oriented Programming**
- ❖ **Advanced Concepts – Prototype, Closures, Apply(), Call() Etc.,**

Jquery

- ❖ **Attributes**
- ❖ **Content And Css Manipulations**
- ❖ **Dom**
- ❖ **Events**
- ❖ **Effects**
- ❖ **Animations**
- ❖ **And More!**

Ajax

- ❖ **$.Ajax() Method**
- ❖ **Jqxhr (Jquery Xmlhttprequest) Vs. Xhr (Xmlhttprequest)**
- ❖ **Invoking Jquery Ajax Http Methods**
- ❖ **Synchronous Vs. Asynchronous Communication**
- ❖ **Jquery Ajax Global Event Handlers**
  - o **$.Ajaxsend()**
  - o **$.Ajaxstart()**
  - o **$.Ajaxstop()**
  - o **$.Ajaxsuccess()**
  - o **$.Ajaxerror()**
  - o **$.Ajaxcomplete()**
- ❖ **Using $.Ajaxsetup() To Globalize Parameters**

- ❖ **Using $.Ajaxprefilter() To Filter Ajax Requests**
- ❖ **Other Ajax Powered Functions In Jquery**
    - ○ **$.Get() And $.Post() Functions**
    - ○ **$.Load() Function**
    - ○ **$.Getscript()**

Json

- ❖ **Introduction**
- ❖ **Json Vs Xml**
- ❖ **Json Object**
- ❖ **Json Array**
- ❖ **Ajax Json**

Type Script

What is a SPA? What is Angular?

- ○ **Single Page Apps VS Conventional Web Applications**
- ○ **Where Angular Fits**
- ○ **Angular-9 is arrived (what's new)**

Preparing for Type Script

- ○ **Brief Intro to TypeScript**
- ○ **The TypeScript Module System**
- ○ **Getting Setup**
- ○ **Project Setup**
- ○ **String Templates**
- ○ **Const**
- ○ **Let**
- ○ **Rest & Spread Operators**
- ○ **Arrow Functions**
- ○ **Maps**
- ○ **Sets**
- ○ **Interface**
- ○ **Generics**
- ○ **New Object Features**
- ○ **Classes**
- ○ **Destructing**
- ○ **Typescript decorators**

Building with Components

- o **A Trivial Component**
- o **Nested Components**
- o **Deeper Nesting**

Bootstrap Scaffolding

- o **What Is Bootstrap?**
- o **Bootstrap File Structure**
- o **Basic HTML Template**
- o **Global Styles**
- o **Default Grid System**
- o **Fluid Grid System**
- o **Bootstrap CSS**
- o **Typography**
- o **Code**
- o **Tables**
- o **Forms**
- o **Buttons**
- o **Images**
- o **Icons**

Angularjs

Angular Binding and Events

- o **Templates**
- o **Bindings**
- o **Event Bindings**
- o **Two-way Binding**
- o **Class and Style Bindings**
- o **Built-in Directives**
- o **Component Data Binding**
- o **Component Events**
- o **Ng-model**
- o **Introduction to webpack/systemJS**

Dependency Injection and services

- o **Why Dependency Injection?**
- o **Injection API**

- Component Multi Providers
- Injection Multi Providers
- Services Overview
- Creating a Service
- Injecting a Service

Directives

- Introduction and Examples
- Directives Overview
- Attribute Directives
- Custom Attribute Directive
- Structural Directives
- Structural Directive: ngIf
- Structural Directive: ngFor
- Structural Directive: ngSwitch
- Custom Structural Directive

Pipes

- Using Pipes
- Creating Pipes
- Built-in Pipes
- Chaining Pipes
- Custom Pipes
- Stateful Pipes

Forms

- Controls
- Control groups
- Form Builder
- Template Driven Approach
- Data Driven Approach

Routing

- Introduction and Use
- Creating and configuring Routes
- Imperative Routing
- Routing Parameters
- Child Routes

- Routing Lifecycle Hooks
- Setting up Firebase
- HTTP POST Request
- HTTP GET Request

HTTP, Promises, and Observables

- Promises – still available, still useful
- Calling an API via HTTP
- Introduction to Observables
- Observable Bindings
- Observables Operator

Testing

- Unit Testing
- End to End Testing
- Angular-9 new features (Add-on)
- ..Else Template Conditions
- Animation Module
- Typescript's StrictNullChecks
- Angular Universal
- FESM
- Angular2 to Angular9 migrartion
- Router ParamMap
- TypeScript 2.1/2.2