

# FULL STACK WEB DEVELOPMENT (Assignment) (1)

B. Bhumika  
2111CS020095  
Aiml (Beta)

- (1) Define Differentiate the following (a) Angular and angular js (b) single page Application, and multipage Application.

A:- Angular JS:-

- Architecture: Angular JS follows a modular architecture with controls and directives.
- language: It is written in Javascript.
- Two-way Data Binding: AngularJS uses two-way data binding, which means changes in the UI or the model automatically reflect in the other.
- scope: In AngularJS, \$ scope is used for communication between controllers and views.

Angular:-

- Architecture: Angular adopts a component-based architecture with clear separation of concerns using components, services and modules.
- language: It is primarily written in type-script
- scope: Angular uses a hierarchical dependency injection system instead of \$ scope, making it more efficient and modular.



## ⑤ Singular Page Application (Vs) Multi - Page Application :-

### • Single Page Application :-

\* Loading Approach :- SPAs load a single HTML page and dynamically update the content as the user interacts with the application.

\* Smooth Experience :- SPAs provide a smoother user experience by avoiding full-page reloads, utilizing AJAX for fetching data.

\* Frameworks :- SPAs often leverage front-end frameworks like React, Angular (or) Vue.js.

### • Multi - Page Application :-

\* Traditional Approach :- Traditional websites with separate HTML pages for different content are examples of MPAs.

\* Full Page Reloads :- Each user interaction in an MPA usually results in a full-page reload.

\* Navigation :- MPAs rely on server-side routing for navigation.

\* Loading Approach :- MPAs load entirely new HTML pages from the server when the user navigates to a different section (or) performs an action.



Q Describe the Angular components and its communication in detail?

A:- ① Component Structure:-

- Class:- Represents the component's behavior and properties using TypeScript.
- Decorator:- Provides metadata such as the component's selector, template, styles and more.

② Template:-

- The HTML part of the component that defines the structure of user interface.
- May include Angular bindings, directives and other Angular-specific syntax.

③ Styles:-

- CSS styles applied specifically to the component.

④ Event Binding:-

- Allows the binding of DOM events to methods in component class.

⑤ Input and Output Properties:-

- Input properties:- Allow data to flow into a component.

- Output properties:-

Enable a component to emit custom events that can be captured by parent components.



### ⑥ Services :-

- Components can use services to share data and functionality between them. services are typically singleton objects that can be injected into components.

### ⑦ lifecycle Hooks :-

- methods in the component class that are called at specific points in component's life cycle such as "ngonchanges".

### ⑧ Data Binding :-

- Angular support one-way and two-way binding for communication between the component class and its template.

### ⑨ ViewChild and ContentChild :-

- used to access child component (or) elements within the component's template.

### ⑩ Dependency Injection :-

- Angular uses a hierarchical dependency injection system, allowing components to receive dependencies from parent component to services.