**Angular Installation and Importance**:

**SW HW requirement:**

* VS code (Latest)
* node js (Latest)
* npm (Latest) - by cmd
* Angular (14) - by cmd

1. What & Why Angular - latest & stable version
2. Building blocks of angular
3. Architecture of angular
4. Features of angular
5. Flow of angular application
6. Files and folder structure of angular

**Angular official website - https://angular.io/**

**Angular Installation steps:**

1. <https://nodejs.org/en/download>
2. <https://code.visualstudio.com/download>
3. - Install Extension and restart - Angular, Code runner, Html CSS IntelliSense
4. NPM (Node Package Manager), npm i npm@latest -g

(if path not set for npm - ng not recognized)

1. Set Environment Variable append this path to the end of the existing path = %AppData%\npm; C:\Program Files\NodeJS\;
2. Check the version - node -v v19.4.0npm -v 9.2.0restart vs code

**Angular CLI installation/update**:

1. Update npm to Latest version - **npm i npm@latest -g** restart vs code
2. Note: To install a specific version, you can use **npm install -g @angular/cli@14.0.0** restart vs code
3. Check if old version of Angular is already installed on your computer, if exist then remove it **npm uninstall -g @angular/cli**
4. npm cache clean **npm install -g @angular/cli**
5. **What is Angular?**

* It is a Framework to develop Component based web application
* It is Based on Typescript - which is an OOP language - based on JS **.ts will transpile in .js**
* A collection of well-integrated libraries that cover a wide variety of features, including routing, forms management, client-server communication, and more**.**
* A suite of developer tools to help you develop, build, test, and update your code
* (what is a framework? - semi complete, reusable application , to reduce boilerplate code )

1. **Why Angular?**

* powerful framework
* powerful CLI
* DevTools
* SPA Single Page Application (rendering the data in the same page without reloading)
* **MVC** - Model (Customer) View (CustomerRegisterComponent) Controller (Flow of request & Response)
* **MVVM** – MV VM modelView ViewModel
* **DI** - Dependency Injection - we must get the dependent Object, rather than creating it.

Whenever we need the dependent Object, creation and management of the required object will be taken care by the framework. Angular support constructure DI.

* angular support constructure DI only

TravelComponent (Vehicle V1) {}

1. **Building blocks of Angular?**

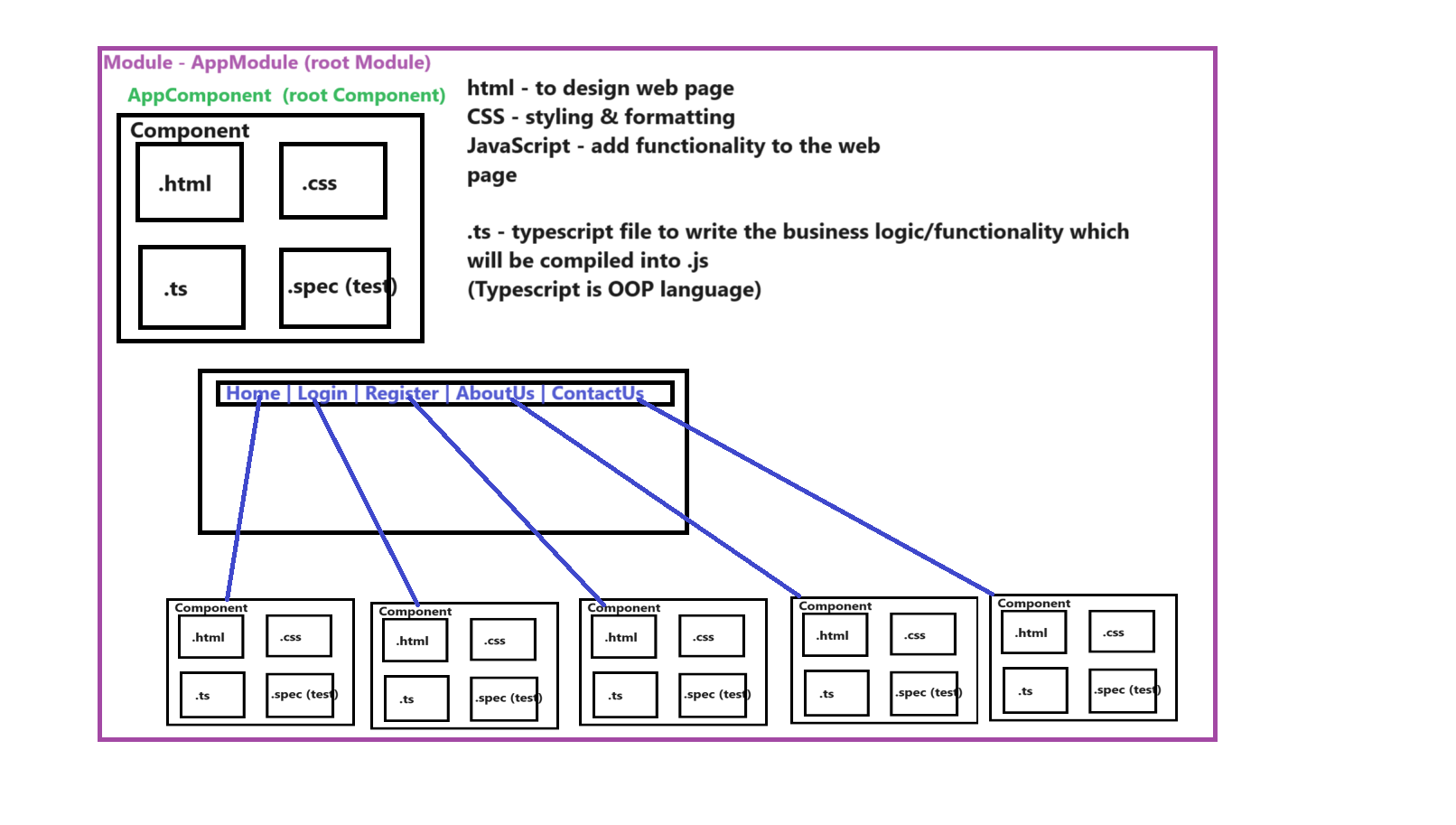
* **Module** - Act as a container for the angular components & services
* **Component** - It’s a part of view
* Template (html)
* **Metadata** - additional information about the specific building block
* **data binding** - the data display is bind with the application backend data or logic
* one-way-binding (String interpolation)
* property binding
* event binding
* two-way-binding (Form & Model)
* **Directives** - which change the structure or behaviour of the web components
* Component Directive
* Structural
* Attribute

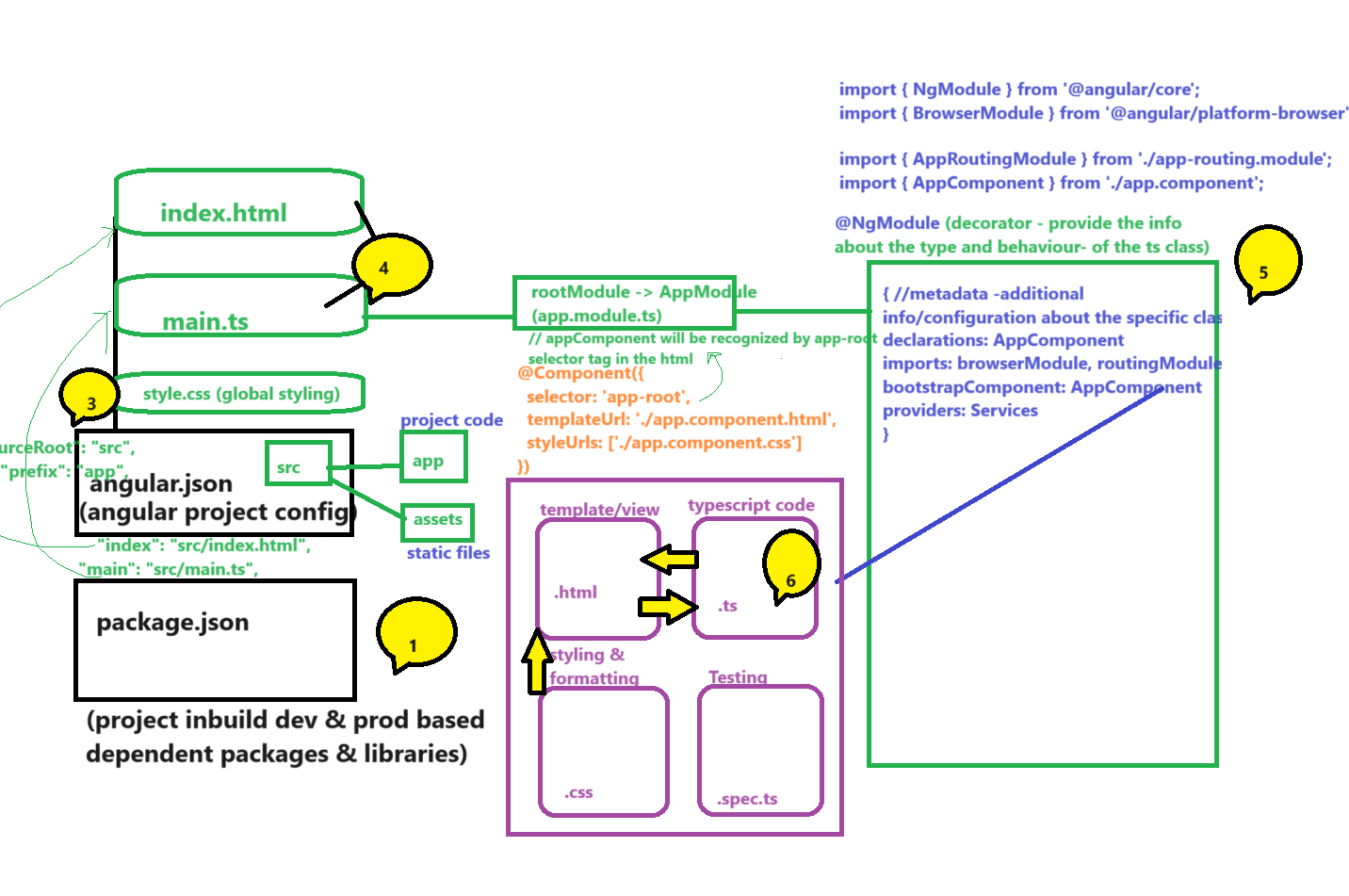
1. **Services** **-** To provide the interaction with the backend API or Database
2. **DI** **(dependency Injection)** - It is a design pattern which allow you to get the dependent object by the framework as and when required - dependent Object, creation and management of the required object will be taken care by the framework. Angular support constructure DI.
3. **Pipes**: Pipes are used for data transformation only at the UI level (in the html page only)

Ex: salary 10000**,** currency | salary, $10000

1. **Routing**: navigation from one component to another component

**Angular Component and modules workflow:**



**Angular Application flow -Files & Folder structure:**

1. **Create new project**

* ng new my-first-angular-app
* skip to install node modules while creating project.
* ng new my-first-angular-app –skip-install
* npm install [run this command separately to load all dependencies]
* ? Would you like to add Angular routing? Yes
* ? Which stylesheet format would you like to use? CSS
* Run your application to get started **cd my-first-angular-app** and use command **ng serve**
* Open in the browser <http://localhost:4200>
* Make the changes in the app-component.ts file (just replace the below code)

import {Component } from '@angular/core';

@Component ({

selector: 'app-root',

templateUrl: './app.component.html',

styleUrls: ['./app.component.css']

})

export class AppComponent {

title = 'my-first-angular-app';

uname='Smita';

msg='Welcome to Angular World!!';

}

make the changes in the app-component.html file (just replace the below code)

<h1> {{title}} app is running! </h1><hr>

<! -- String Interpolation or One way data binding!! -->

<h2>Hello, {{uname}}!!! <br> {{msg}} </h2>

<router-outlet></router-outlet>

* **Life cycle hook methods for the angular**  
  <https://nalawadeshivani98.medium.com/angular-lifecycle-d34c57bf147>
* The best new features in angular 17  
  [The best new features in Angular 17: A kinder, faster Angular | InfoWorld](https://www.infoworld.com/article/3711373/the-best-new-features-in-angular-17-a-kinder-faster-angular.html)
* What’s new in angular 17  
  [What’s new in Angular 17?. Angular 17 Features | by mallikharjuna | Dec, 2023 | Medium](https://medium.com/@sangumallikharjuna/whats-new-in-angular-17-a4590a405a48)