ESTRUCTURA BÁSICA ANGULAR

**Módulo**

Angular se basa en módulos para trabajar. Cargarán los componentes, directivas, servicios, etc, de nuestro proyecto. Se podría definir como la unidad de medida más grande dentro de Angular.  
Siempre hay un módulo raíz que activará el enlazamiento inicial y contendrá todo lo necesario para nuestra aplicación

**Componentes**

Junto a las plantillas o *templates* definen una vista o una pequeña parte de HTML donde la plantilla es la parte visual y el componente se encargará de la parte lógica y funcional.

**Metadatos**

@Component({

selector: 'app-root',

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

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

})

Los metadatos se utilizan para definir la funcionalidad de una clase dentro de nuestro proyecto. Se colocan justo antes de la definición de dicha clase. El ejemplo de arriba sería para definir un componente y el siguiente define un servicio.

@Injectable({

providedIn: 'root',

})

**Directivas**

Se utilizan para transformar el DOM de acuerdo con su lógica. Es decir, podremos modificar el aspecto o duplicar código HTML con ayuda de las directivas.

**Data-Binding**

Se trata de una herramienta muy útil de Angular, donde podremos sincronizar un atributo del componente con su plantilla y ver su valor. Si el valor de este atributo se modifica inmediatamente se verá reflejado en la plantilla.  
También podremos enlazar eventos a métodos del componente y más opciones.

**Servicios e Inyector de Dependencias**

Los servicios son inyectados en un componente a través del Inyector de Dependencias de Angular y proveen funcionalidades específicas a nuestra vista.

PARTES DE UN PROYECTO

## **Estructura raíz**

### Node\_modules

Todos aquellos paquetes instalados para poder trabajar con nuestro proyecto

### Src

Contiene la aplicación de Angular, es donde trabajaremos principalmente desarrollando nuestro proyecto

### Angular.json

Archivo de configuración de angular para nuestro proyecto: ruta donde guardar los archivos tras realizar el ng build, configuración para test, etc.

### Package.json

Archivo de configuración básica de nuestro proyecto: nombre, versión, dependencias de paquetes, etc.

### Tsconfig

Archivos de configuración para la compilación de Typescript.

## **Dentro de src**

### Index.html

Como toda aplicación web, se necesita este archivo para acceder a la aplicación.

### Main.ts

Su función principal consiste en lanzar el módulo raíz para poder ejecutar la aplicación.

### Styles.css

Archivo raíz para los estilos de nuestra aplicación

### Favicon.ico

Icono de nuestra aplicación

### App

Contiene todo lo relacionado con nuestra aplicación: módulos, componentes, servicios, etc.

### Assests

Archivos estáticos utilizados en nuestro proyecto.

### Environments

Variables de entorno

## **Dentro de app**

### App.module.ts

Módulo raíz de nuestra aplicación.

### App.component

Componente principal, el primer componente que se carga en nuestra aplicación.