

node\_modules: Todo proyecto de Javascript posee este directorio donde se almacenan las librerías y dependencias que se descarguen con NPM.

src: Directorio principal del proyecto donde encontramos:

app: Directorio donde guardaremos todo el código fuente de Angular.

assets: Directorio para imágenes y otros recursos que la app necesita.

environments: Directorio de ambientes, por defecto viene con desarrollo y producción.

favicon.ico: Ícono por defecto que tendrá la pestaña del navegador.

index.html: Archivo HTML principal desde donde se construye toda la aplicación.

main.ts: Archivo principal para la configuración de Angular.

polyfills.ts: Librería para que Angular funcione en navegadores viejos y que la aplicación sea retro compatible.

styles.scss: Archivo principal de estilos.

test.ts: Archivo principal para lanzar el ambiente de pruebas de Angular.

.browserslistrc: Lista de navegadores y sus versiones que permite configurar la compatibilidad de la aplicación con cada uno.

.editorconfig: Permite autoformatear los archivos, espacios, indentación, etc. Hay que tener instalado la extensión en el editor.

.gitignore: Indicarle a GIT qué archivos/directorios ignorar.

angular.json: Archivo principal con toda la configuración del proyecto Angular.

karma.conf.js: Archivo de configuración de Karma. Karma es un task runner para correr pruebas unitarias.

package-lock.json: Describe el las dependencias exactas que se generaron en la instalación del proyecto.

package.json: Archivo para el manejo de dependencias, scripts y metadatos relevantes para el proyecto.

README.md: Archivo markdown para la documentación del proyecto.

tsconfig.app.json: Archivo principal para la configuración de TypeScript.

tsconfig.json: Extensión con más configuraciones de TypeScript.

tsconfig.spec.json: Configuración de TypeScript pero para el ambiente de pruebas.  
Crear proyecto

```
ng new "nombre"
```

Entrar al proyecto

```
cd "nombre del proyecto"
```

El comando ng serve se usa para levantar un servidor de desarrollo en nuestro entorno local y probar nuestra aplicación

```
ng serve
```

```
ng serve -o
```

```
<h1>Hola Mundo!</h1>
```

Un método de devolución de llamada que se invoca después del detector de cambios

```
export class AppComponent implements OnInit {
```

```
  ngOnInit(): void {
```

```
  }
```

```
}
```

```
</div>
```

```
{{2+2}}
```

```
  edad: number;
```

```
  constructor(){
```

```
    this.edad = 18;
```

```
  }
```

```
<h2>Tu edad es: {{edad}}</h2>
```

```
<div *ngIf="edad<18">
```

```
  Eres menor de edad
```

```
</div>
```

```
<div *ngIf="edad>18">
```

```
  Eres mayor de edad
```

```
</div>
```

```
<h3 [style.background] ="edad">=18 ? 'green' : 'red' "[style.color]="white"> Tu edad es: {{edad}}</h3>
```

```
<button (click)="aumentarEdad()">Aumentar</button>
```

```
<button (click)="disminuirEdad()">disminuir</button>
```

```
  aumentarEdad(){
```

```
    this.edad=this.edad+1;
```

```
    console.log("aumentar")
```

```
  }
```

```
  disminuirEdad(){
```

```
    this.edad=this.edad-1;
```

```
    console.log("disminuir")
```

```
  }
```

```
<h3>Lista de Nombres</h3>
```

```
{{nombres[0]}}
```

```
nombres:Array<string>;
```

```
constructor(){
```

```
  this.nombres=['Victor','Paco']
```

```
}
```

```
<ul><li *ngFor="let nombre of nombres; let indice = index">{{indice+1} + '. '+nombre}</li></ul>
```