

JavaScript Avanzado

Sesión 8



Universidad
Tecnológica
del Perú

¿Tienen alguna consulta o duda sobre la clase previa?

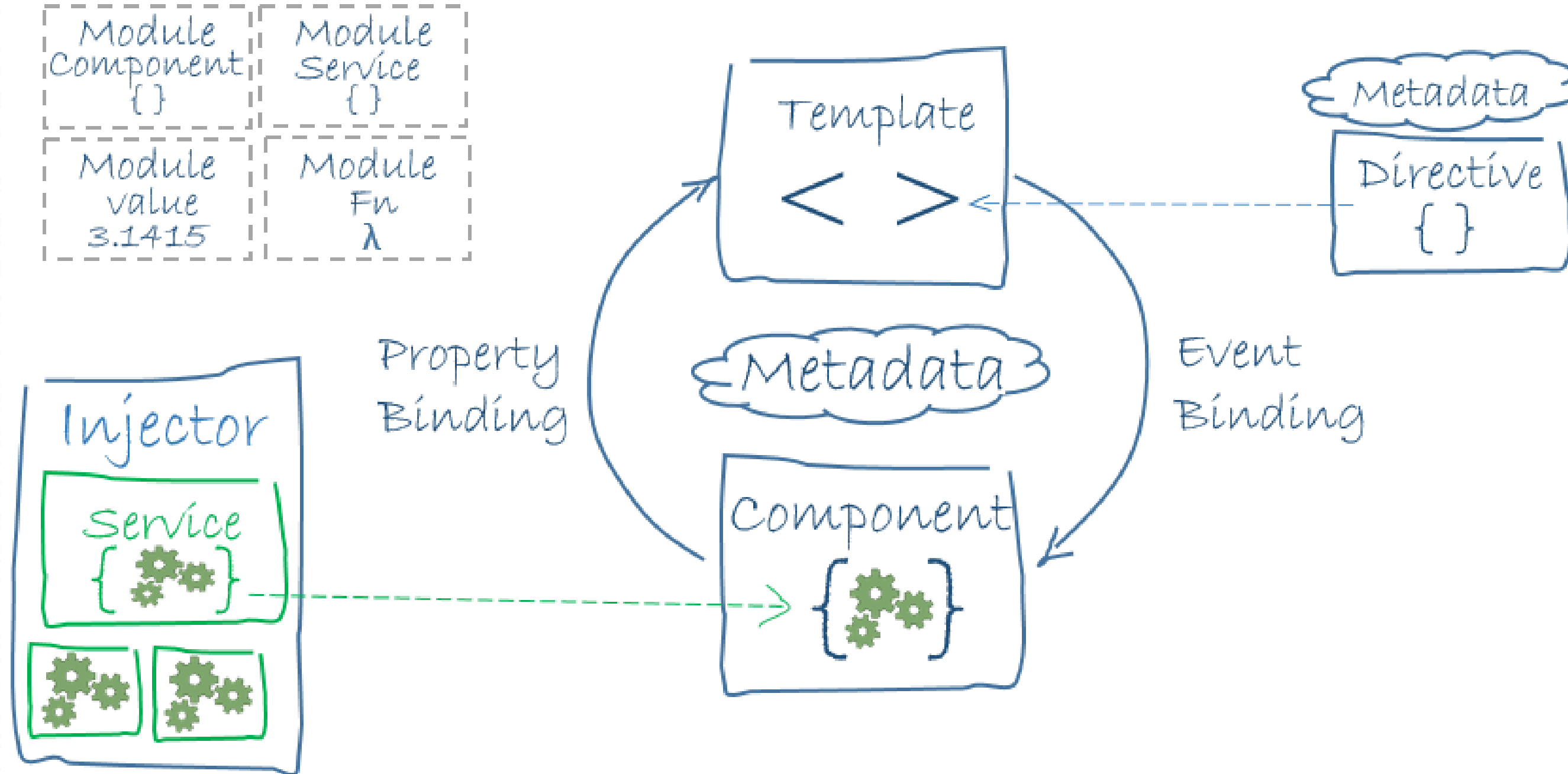


Logro de la sesión

Al finalizar la sesión, el estudiante crea aplicaciones web con el Framework Angular integrando las funcionalidades de navegación y procesamiento formularios.

¿Cómo se compone Angular? ¿Cuáles son sus características?





¿Cuál es la importancia de la arquitectura de Angular?



Contenido

Elementos de Angular

- Componentes
- Navegación
- Pipes
- Formularios
- Implementación de un proyecto web empleando navegación, formateo de elementos y formularios.



Componentes en Angular

- Los componentes son una parte fundamental de Angular y representan los bloques de construcción básicos de la interfaz de usuario (UI) de una aplicación.
- Una aplicación Angular está compuesta por un árbol de componentes, donde **cada componente puede contener otros componentes**, formando una estructura jerárquica.

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

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'example';
}
```

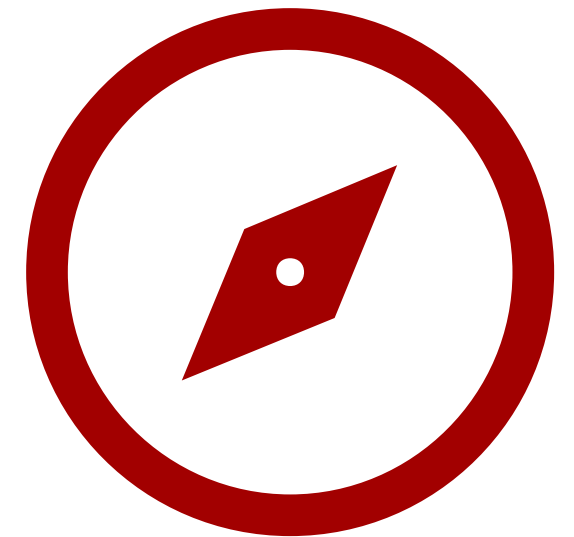
Componentes en Angular

Cada componente en Angular consta de tres partes principales:

- **Plantilla **HTML****: Define lo que se muestra en la página.
- **Clase **TypeScript****: Contiene la lógica y el comportamiento del componente.
- **Selector **CSS****: Permite usar el componente en una plantilla mediante una etiqueta personalizada.

Navegación en Angular

- En una aplicación de una sola página, se cambia lo que ve el usuario mostrando u ocultando partes de la pantalla que corresponden a componentes concretos, en lugar de ir al servidor para obtener una nueva página.
- A medida que los usuarios realizan tareas de la aplicación, deben moverse entre las diferentes vistas que ha definido.



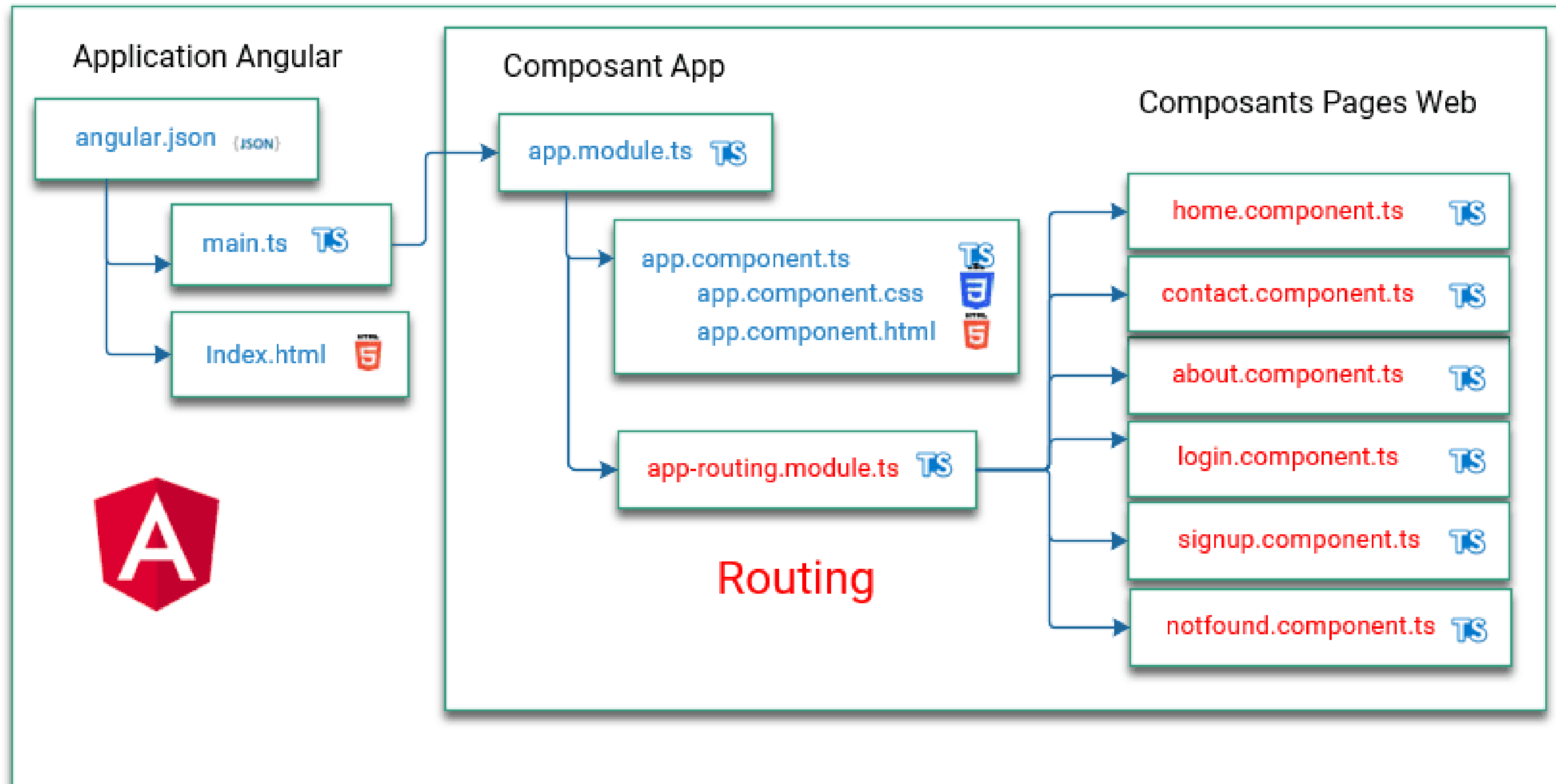
Navegación en Angular

- Para manejar la navegación de una vista a la siguiente, utiliza el enrutador angular.
- El “router” permite la navegación interpretando la URL de un navegador como una instrucción para cambiar la vista.

```
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';

const routes: Routes = [];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule {}
```

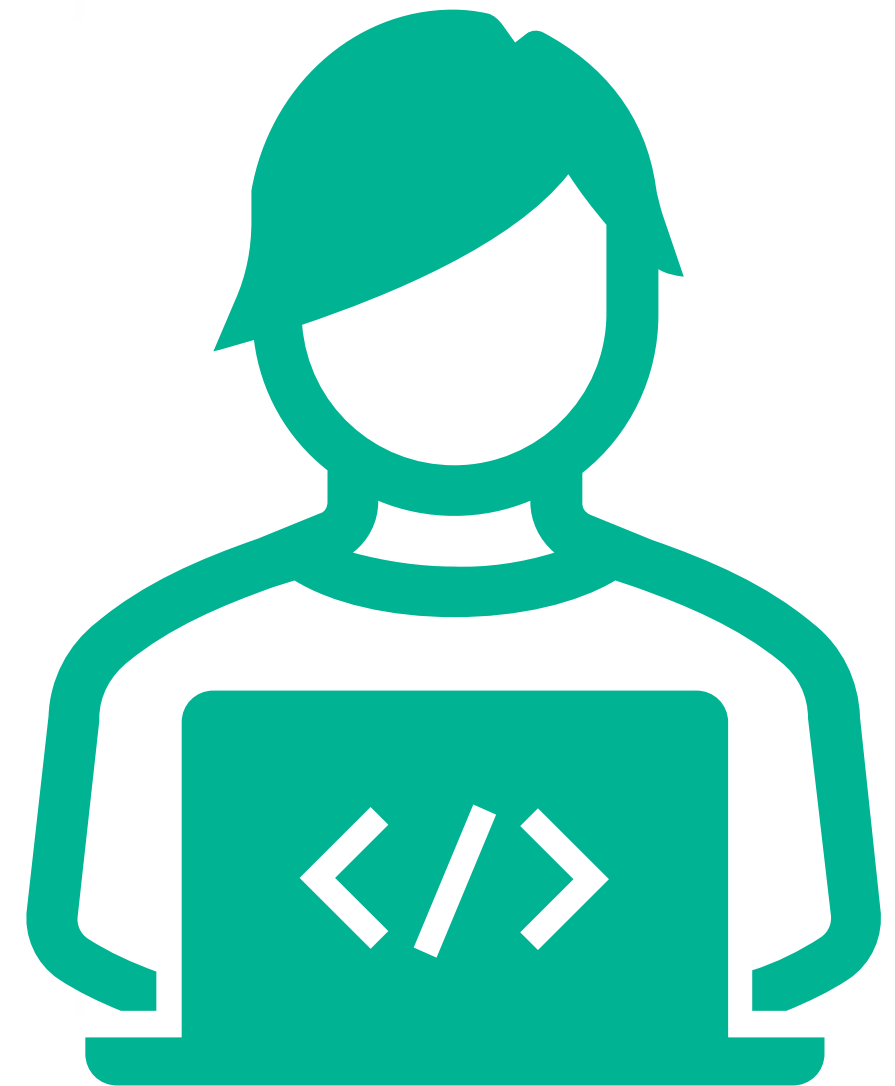


Pipes

- Son herramientas que permiten **transformar visualmente la información** en las plantillas sin alterar los datos subyacentes.
- **Por ejemplo**, puedes usar pipes para cambiar un texto a mayúsculas o minúsculas, formatear fechas y horas, o convertir números a diferentes formatos.
- Se emplea el **símbolo de barra vertical** para indicar un pipe (|)

Programando

- Empleamos **pipes** para de formatos de **moneda**, **fecha** y **cadenas**.



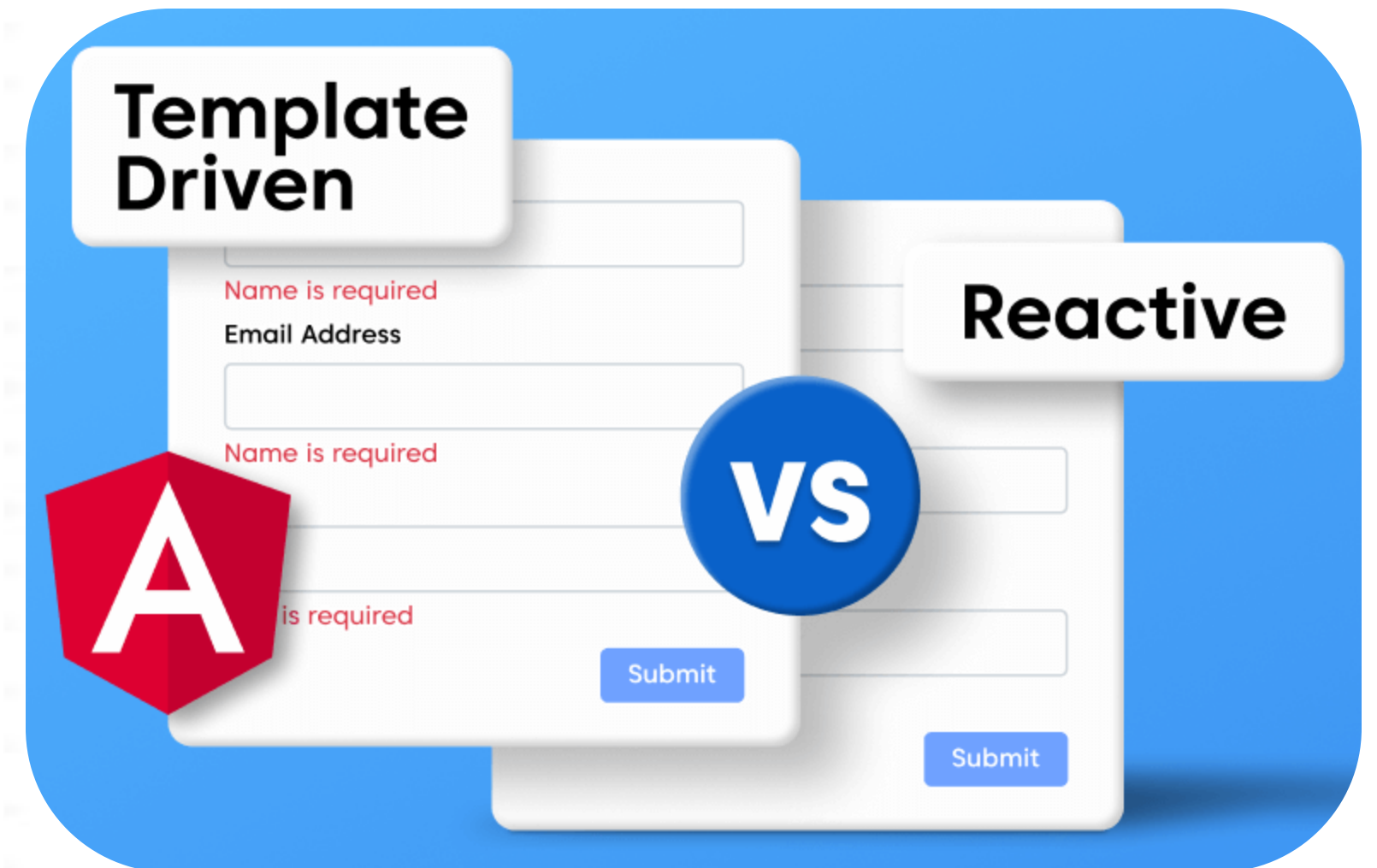
Formularios

- Angular proporciona dos enfoques diferentes para manejar la entrada del usuario a través de formularios: **reactivo** y **basado en plantillas**.
- Los formularios basados en plantillas son una excelente opción para formularios pequeños o simples, mientras que los formularios reactivos son más escalables y adecuados para formularios complejos.



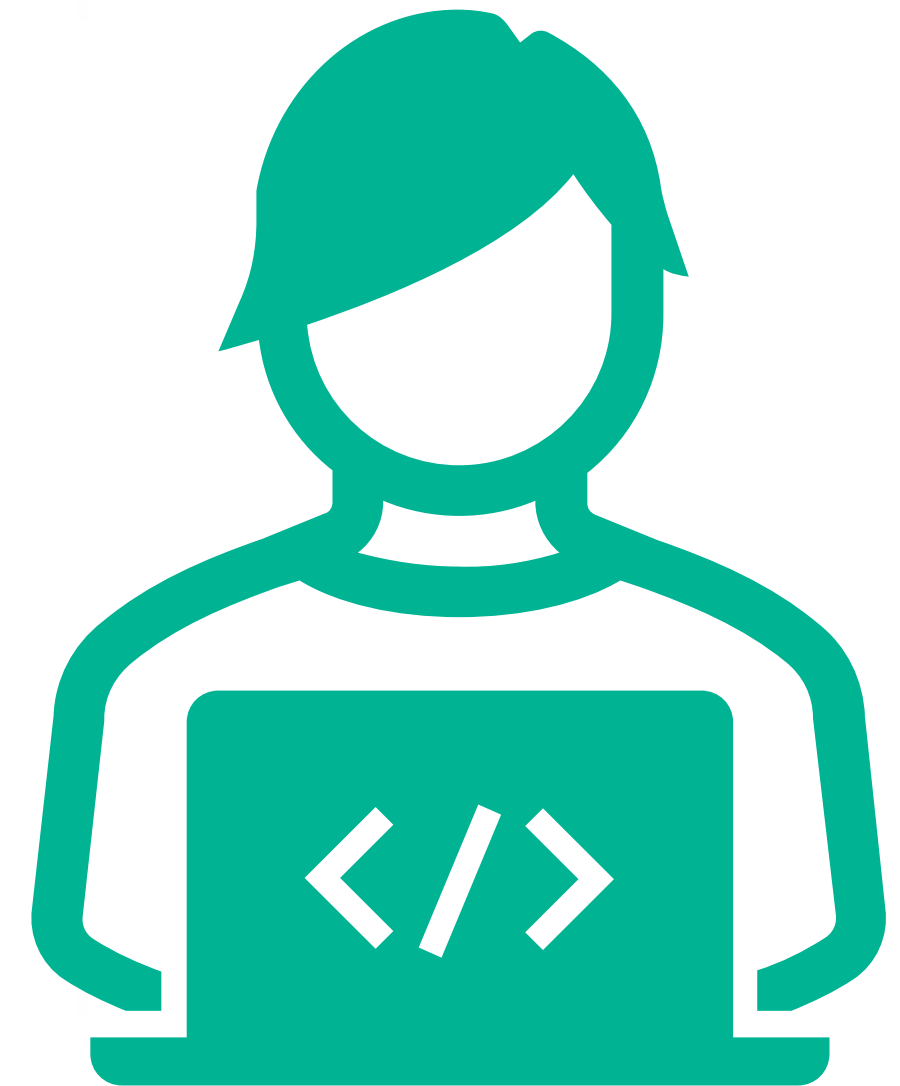
Formularios

- Ambos capturan eventos de entrada de usuario de la vista, validan la entrada del usuario, crean un modelo de formulario y un modelo de datos para actualizar, y proporcionan una forma de realizar un seguimiento de los cambios.



Programando

- Empleamos **formularios**, **navegación** y **pipes** en la creación de un sitio web con Angular.



¿Tienen alguna consulta o duda?



Actividad



Resolver la actividad planteada en la plataforma.

Cierre

¿Qué hemos aprendido hoy?



Elaboramos nuestras conclusiones sobre el tema tratado



**Universidad
Tecnológica
del Perú**