Taller Tutorial de Ionic

Francy Daniela Olmos Vaca

55222017

Ing. Johan Mauricio Fonseca Molano

Universidad de Boyacá

Facultad de Ciencias e Ingeniería

Tunja, Boyacá

2025

**Temas de Angular necesarios para el taller**

1. **Componentes básicos**
   * Qué es un componente y su estructura (decorador @Component, selector, template, style).

Un componente es un bloque, que contiene un template, contiene estilos, contiene lógica.

 **Una estructura o plantilla (el archivo .html):** Piensa en esto como la forma física de la pieza de Lego. Es lo que el usuario ve en la pantalla, como un botón, una tarjeta de producto o un menú de navegación.

 **Un estilo (el archivo .css):** Esto es la pintura y los detalles de la pieza. Le da color, tamaño y la hace ver bien.

 **Una lógica o comportamiento (el archivo .ts):** Esta es la "inteligencia" de la pieza. Le dice qué hacer, por ejemplo, qué pasa cuando alguien hace clic en el botón o cómo se actualiza la información que se muestra.

* + Diferencia entre un componente de página (vista completa) y uno reutilizable (ej. ProductCard).

El componente de página es el **contenedor principal** que representa una vista completa de tu aplicación, como la página de inicio, el perfil de usuario o el carrito de compras.

El componente reutilizable, en cambio, es una pieza **independiente y autónoma** con una funcionalidad muy específica.

1. **Standalone Components (Angular 15+)**
   * Qué significa que no hay módulos (NgModule).

Que una aplicación "no tenga módulos" significa que no usa el **sistema de módulos tradicional** para organizar y agrupar sus componentes, directivas y servicios. En lugar de eso, utiliza un enfoque más moderno y directo.

* + Uso de la propiedad standalone: true.

Este deja de ser una pieza que depende de un módulo para existir. En cambio, se vuelve una unidad completa que puede manejar sus propias dependencias.

* + Importación de otros componentes/módulos dentro de imports.

Cuando un módulo necesita acceder a los componentes o servicios proporcionados por otro módulo, importa ese módulo utilizando la matriz de importaciones en el decorador @NgModule. ayudame a parafrasearlo.

1. **Data Binding**
   * **Interpolación** ({{ }})

Es el mecanismo para sustituir una expresión por un valor de tipo string en la plantilla (template).

* + **Property Binding** ([prop]="value")

Una forma específica de enlace de datos unidireccional que se utiliza para establecer el valor de la propiedad de un elemento DOM o la propiedad de entrada de una directiva en un valor del componente.

* + **Event Binding** ((event)="handler()")

La vinculación a eventos le permite escuchar y responder a las acciones del usuario, como pulsaciones de teclas, movimientos del mouse, clics y toques.

1. **Directivas estructurales**
   * \*ngFor
   * \*ngIf.
2. **Decoradores @Input y @Output**
   * @Input()
   * @Output()

**Referencias**

*Angular*. (s/f). Angular.Io. Recuperado el 16 de septiembre de 2025, de https://v17.angular.io/guide/event-binding

Del Castillo, R. T. (2024, febrero 8). *Angular 17*. Medium. https://medium.com/@t0k1dev/angular-17-standalone-components-d0ba91488841

*How do you import a module into another module Angular?* (2024, abril 29). GeeksforGeeks. https://www.geeksforgeeks.org/angular-js/how-do-you-import-a-module-into-another-module-angular/

Landajuela, I. (s/f). *Interpolacion - programación con angular*. Gitlab.Io. Recuperado el 16 de septiembre de 2025, de https://soka.gitlab.io/angular/conceptos/data-binding/interpolacion/interpolacion/

*Que son los componentes en angular*. (2017, noviembre 15). https://platzi.com; Platzi. https://platzi.com/blog/que-son-los-componentes-en-angular/