**Paso 6 — Consultar sobre**

**Temas de Angular necesarios para el taller**

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

En Angular, un componente es una unidad básica de la aplicación que controla una vista. Su estructura se define mediante el decorador @Component, que incluye:

* + - selector: etiqueta HTML para incluir el componente.
    - template: el contenido HTML que renderiza.
    - style: estilos del componente.
  + Diferencia entre un componente de página (vista completa) y uno reutilizable (ej. ProductCard).

Los componentes pueden ser páginas completas (como vistas principales) o componentes reutilizables, como una tarjeta de producto ( ProductCard) (Tutoriales Programación YA, sf; Angular, 2011).

1. **Standalone Components (Angular 15+)**
   * Qué significa que no hay módulos (NgModule).
   * Uso de la propiedad standalone: true.
   * Importación de otros componentes/módulos dentro de imports.

Desde Angular 15, los componentes pueden definirse como standalone, es decir, que no requieren estar declarados en un NgModule. Se determina con la propiedad standalone: true dentro del decorador @Component. Esto permite importar otros componentes o módulos directamente en la propiedad imports sin necesidad de módulos (Maestro, 2023; CertiDev, 2024).

1. **Data Binding**
   * **Interpolación** ({{ }})
   * **Property Binding** ([prop]="value")
   * **Event Binding** ((event)="handler()")

Angular soporta varios tipos de enlace de datos:

* Interpolación ({{ }}): para insertar valores en la vista.
* Property Binding ([prop]="value"): para enlazar propiedades de elementos.
* Event Binding ((event)="handler()"): para manejar eventos (Olayeancarh, 2025; Angular Docs, 2011).

1. **Directivas estructurales**
   * \*ngFor
   * \*ngIf.

Las directivas estructurales modifican la estructura del DOM:

* ngFor: para iterar listas y renderizar múltiples elementos.
* ngIf: para mostrar u ocultar elementos en función de una condición (Imagina Formación, 2025; Tutoriales Programación YA, s.f.).

1. **Decoradores @Input y @Output**
   * @Input()
   * @Output()

**Para la comunicación padre-hijo en componentes:**

* **@Input():** sirve para recibir datos del componente padre.
* **@Output():** permite emitir eventos para que el componente padre escuche y reaccione (Dominicode, 2023; Tutoriales Programación YA, s.f.).

**Referencias**

**Angular. (2011). Anatomy of components.** [**https://angular.dev/guide/components**](https://angular.dev/guide/components)

**Tutoriales Programación YA. (s.f.). TypeScript y Angular: decorador @Component.** [**https://www.tutorialesprogramacionya.com/angulardevya/detalleconcepto.php?punto=60&codigo=60&inicio=40**](https://www.tutorialesprogramacionya.com/angulardevya/detalleconcepto.php?punto=60&codigo=60&inicio=40)

**Tutoriales Programación YA. (s.f.). \*Directivas \*ngIf y ngFor.** [**https://www.tutorialesprogramacionya.com/angularya/detalleconcepto.php?punto=5&codigo=5&inicio=0**](https://www.tutorialesprogramacionya.com/angularya/detalleconcepto.php?punto=5&codigo=5&inicio=0)

**Tutoriales Programación YA. (s.f.). Decoradores @Input y @Output.** [**https://www.tutorialesprogramacionya.com/angulardevya/detalleconcepto.php?punto=61&codigo=61&inicio=60**](https://www.tutorialesprogramacionya.com/angulardevya/detalleconcepto.php?punto=61&codigo=61&inicio=60)

**Dominicode. (2023, enero 5). ¿Qué es un decorador en Angular? @Input y @Output.** [**https://dominicode.com/ques-es-un-decorador-en-angular-input-y-output/**](https://dominicode.com/ques-es-un-decorador-en-angular-input-y-output/)

**Maestro, I. (2023, enero 15). Implementing standalone components in Angular 15.** [**https://dev.to/ismaestro/implementing-standalone-components-in-angular-15-3c2p**](https://dev.to/ismaestro/implementing-standalone-components-in-angular-15-3c2p)

**CertiDev. (2024, agosto 26). Angular componentes standalone: creación y uso.** [**https://certidevs.com/tutorial-angular-componentes-standalone**](https://certidevs.com/tutorial-angular-componentes-standalone)

**Olayeancarh. (2025, agosto 30). A deep dive into interpolation, property, event, and two-way binding.** [**https://dev.to/olayeancarh/unlocking-angulars-data-binding-a-deep-dive-into-interpolation-property-event-and-two-way-4j8h**](https://dev.to/olayeancarh/unlocking-angulars-data-binding-a-deep-dive-into-interpolation-property-event-and-two-way-4j8h)

**Imagina Formación. (2025, septiembre 4). Directivas ngIf y ngFor de Angular.** [**https://imaginaformacion.com/tutoriales/guia-de-las-directivas-ngif-y-ngfor-de-angular**](https://imaginaformacion.com/tutoriales/guia-de-las-directivas-ngif-y-ngfor-de-angular)