

Introducción a Componentes



Resumen

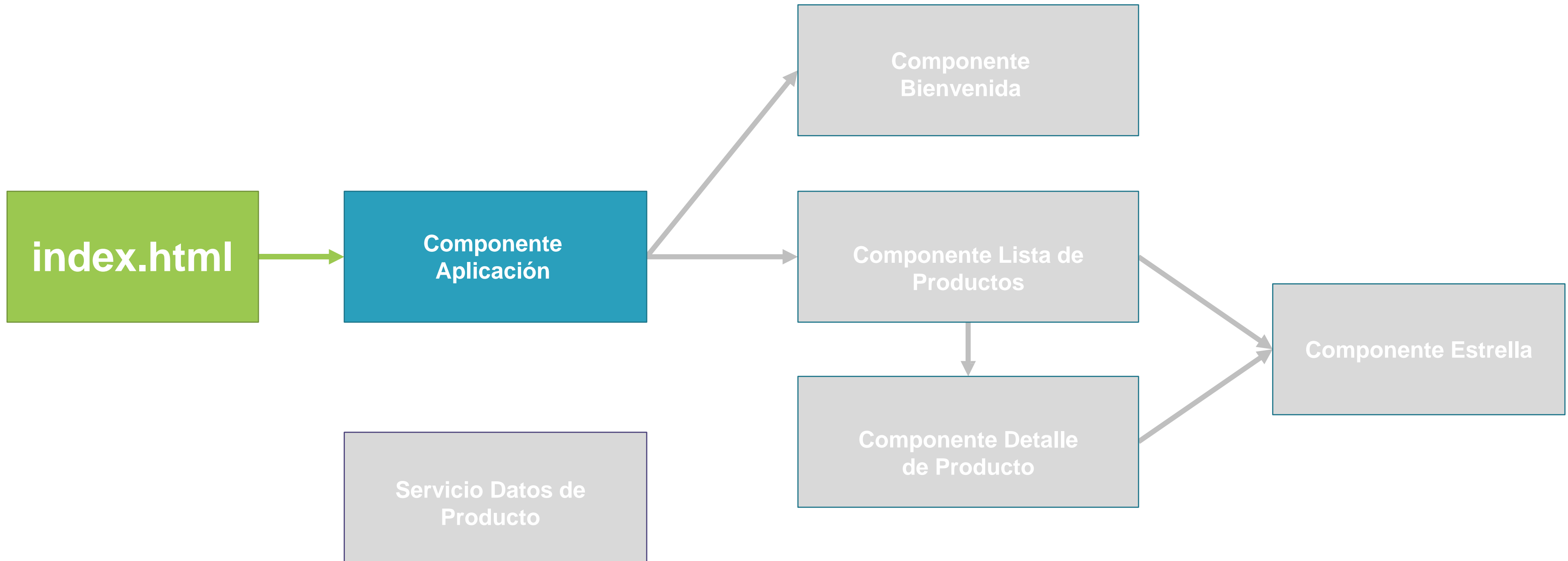


What is a component?

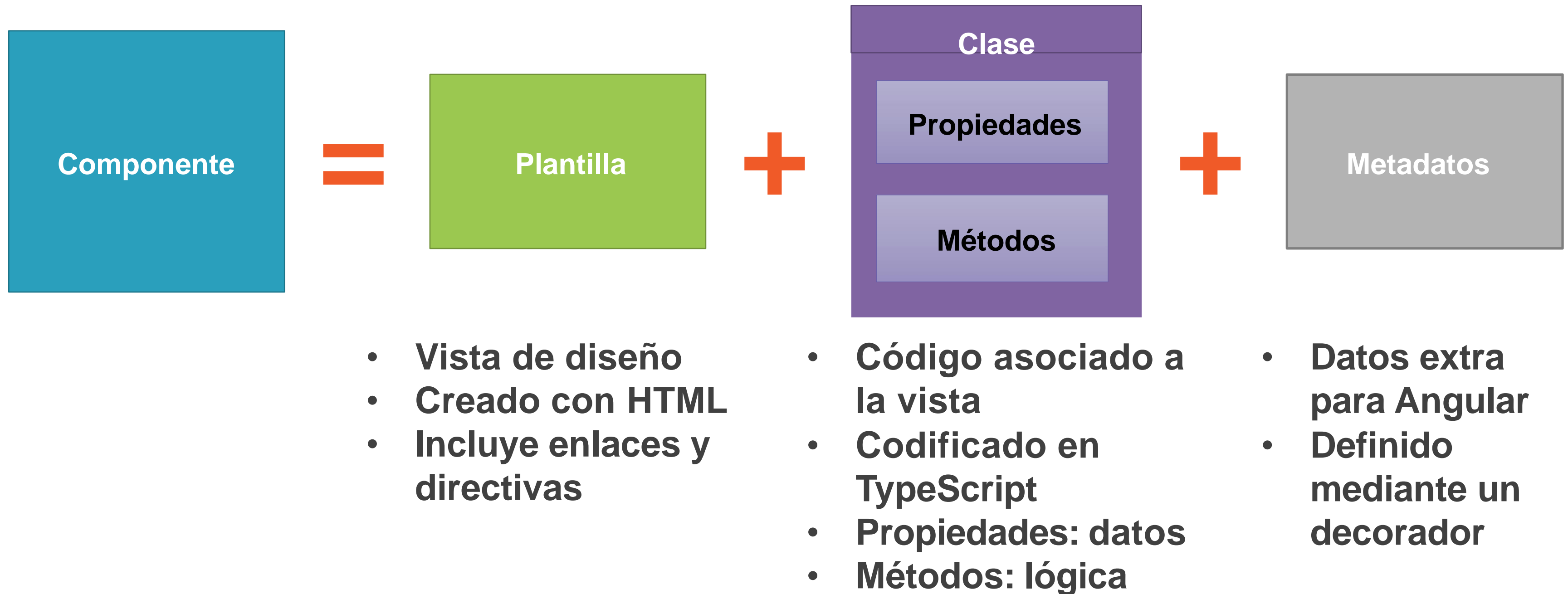
Creating the component class

- Qué es un componente
- Crear la clase del componente
- Definir los metadatos mediante un decorador
- Importar lo necesario
- Preparar el componente de aplicación
- Algo ha ido mal?

Arquitectura de la aplicación



¿ Qué es un componente ?



Componente

app.component.ts

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

Import

```
@Component({  
  selector: 'pm-root',  
  template: `  
    <div><h1>{{pageTitle}}</h1>  
      <div>My First Component</div>  
    </div>  
  `,  
})
```

**Metadatos y
Plantilla**

```
export class AppComponent {  
  pageTitle: string = 'Acme Product Management';  
}
```

Clase

Crear la clase del Componente

app.component.ts

```
export class AppComponent {  
  pageTitle: string = 'Acme Product Management';  
}
```

Palabra clave
class

Nombre de
clase

Palabra clave
export

Nombre del
componente cuando se
utiliza en el código

Crear la clase del componente

app.component.ts

```
export class AppComponent {  
  pageTitle: string = 'Acme Product Management';  
}
```

Nombre de
propiedad

Tipo de datos

Valor
predeterminado

Métodos

Definiendo los metadatos

app.component.ts

```
@Component ({
  selector: 'pm-root',
  template: `
    <div><h1>{{pageTitle}}</h1>
      <div>My First Component</div>
    </div>
  `
})
export class AppComponent {
  pageTitle: string = 'Acme Product Management';
}
```

¿ Qué es un Decorador ?

- Una función que añade **metadatos** a una clase, a sus miembros o a los argumentos de un método.
- Precedido con una **@**
- Angular proporciona decoradores integrados
 - `@Component ()`

Definiendo los metadatos

app.component.ts

```
@Component({  
  selector: 'pm-root',  
  template: `  
    <div><h1>{{pageTitle}}</h1>  
      <div>My First Component</div>  
    </div>  
  `,  
})  
export class AppComponent {  
  pageTitle: string = 'Acme Product Management';  
}
```

Decorador del componente

**Directiva de nombre de
componente en HTML**

Plantilla de la vista

Enlace (Binding)

Importando únicamente lo necesario



- **Antes de utilizar una función o clase externa, definimos dónde encontrarla**
- **Mediante la sentencia import**
- **La sentencia import nos permite utilizar miembros exportados de**
 - **Otros archivos en nuestra aplicación**
 - **Framework Angular**
 - **Librerías externas de JavaScript**

Importando únicamente lo necesario

app.component.ts

```
@Component ({
  selector: 'pm-root',
  template: `
    <div><h1>{{pageTitle}}</h1>
      <div>My First Component</div>
    </div>
  `
})
export class AppComponent {
  pageTitle: string = 'Acme Product Management';
}
```

Importando únicamente lo necesario

app.component.ts

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

```
@Component({  
  selector: 'pm-root',  
  template: `  
    <div><h1>{{pageTitle}}</h1>  
      <div>My First Component</div>  
    </div>  
  `,  
})
```

```
export class AppComponent {  
  pageTitle: string = 'Acme Product Management';  
}
```

Palabra clave import

**Nombre de biblioteca
Angular**

Nombre de miembro

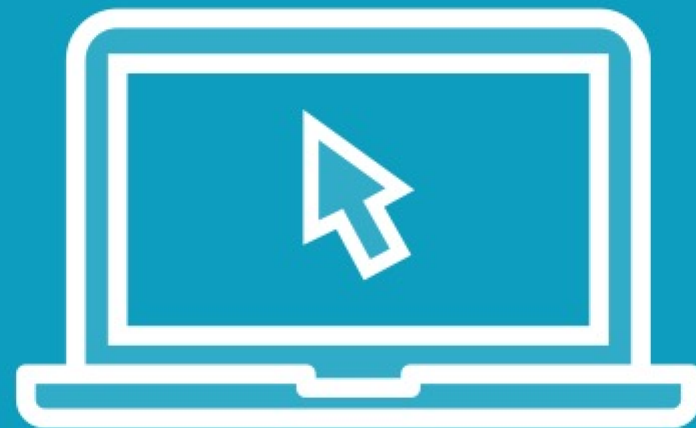
Componente completado

app.component.ts

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

@Component({
  selector: 'pm-root',
  template: `
    <div><h1>{{pageTitle}}</h1>
      <div>My First Component</div>
    </div>
  `
})
export class AppComponent {
  pageTitle: string = 'Acme Product Management';
}
```

Demo



Crear el componente de aplicación

Cómo poner en marcha nuestro componente de aplicación



- Alojar la aplicación
- Definir el módulo Angular



Navegador

Servidor
Web

Petición URL
(www.mysite.com)

Respuesta

index.html

JavaScript

Single Page Application (SPA)



- `index.html` **contiene la página principal de la aplicación**
- **A menudo es la única página web en la aplicación**
- **Por lo tanto, una aplicación Angular a menudo se denomina Aplicación de una Sola Página (Single Page Application)**

Alojando la aplicación (Hosting)

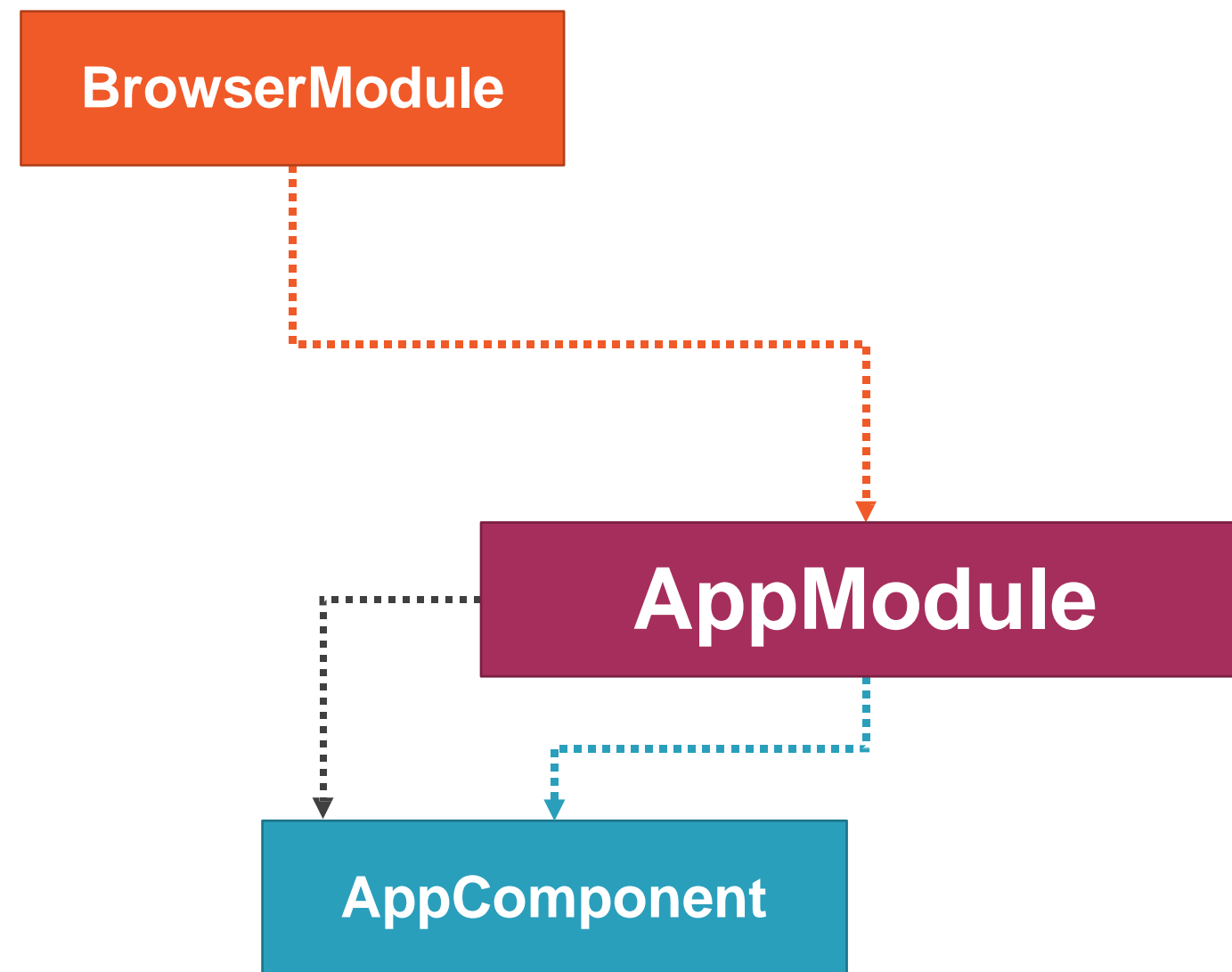
index.html

```
<body>
  <pm-root></pm-root>
</body>
```

app.component.ts

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

@Component({
  selector: 'pm-root',
  template: `
    <div><h1>{{pageTitle}}</h1>
      <div>My First Component</div>
    </div>
  `
})
export class AppComponent {
  pageTitle: string = 'Acme Product Management';
}
```



- Límites de la organización
- Entorno de localización de plantillas

— Imports

— Exports

— Declaraciones

— Arranque

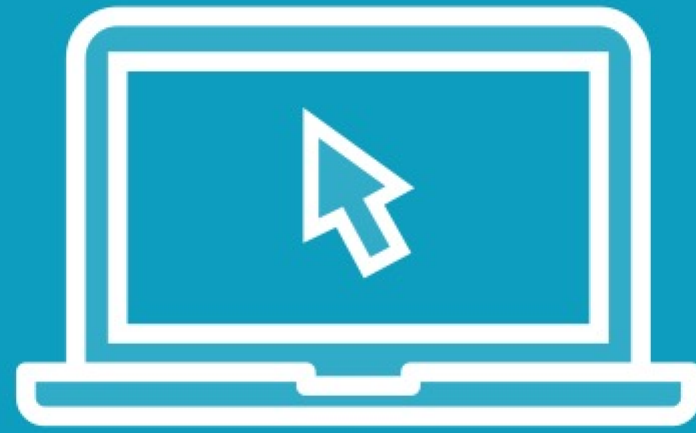
Definiendo el módulo Angular

app.module.ts

```
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';
import { AppComponent } from './app.component';

@NgModule({
  imports: [ BrowserModule ],
  declarations: [ AppComponent ],
  bootstrap: [ AppComponent ]
})
export class AppModule { }
```


Demo



**Poniendo en marcha nuestro
componente de Aplicación**

Angular compila
nuestras plantillas
HTML y componentes
TypeScript a JavaScript

¿Algo ha ido mal ?



The screenshot shows a terminal window with the following text:

```
PROBLEMS 1 TERMINAL ... 1: node
Build at: 2021-02-13T17:19:23.766Z - Hash: 037fb9aff1a10daf940d - Time: 152ms
Error: src/app/app.component.ts:12:14 - error TS2304: Cannot find name 'strin'.
12  pageTitle: strin = 'Acme Product Management';
    ~~~~~
```

**El compilador
muestra errores
de sintaxis**

```
export class AppComponent {  
  pageTitle = 'Acme PM';  
  PageTitle = 'Something else';  
}
```

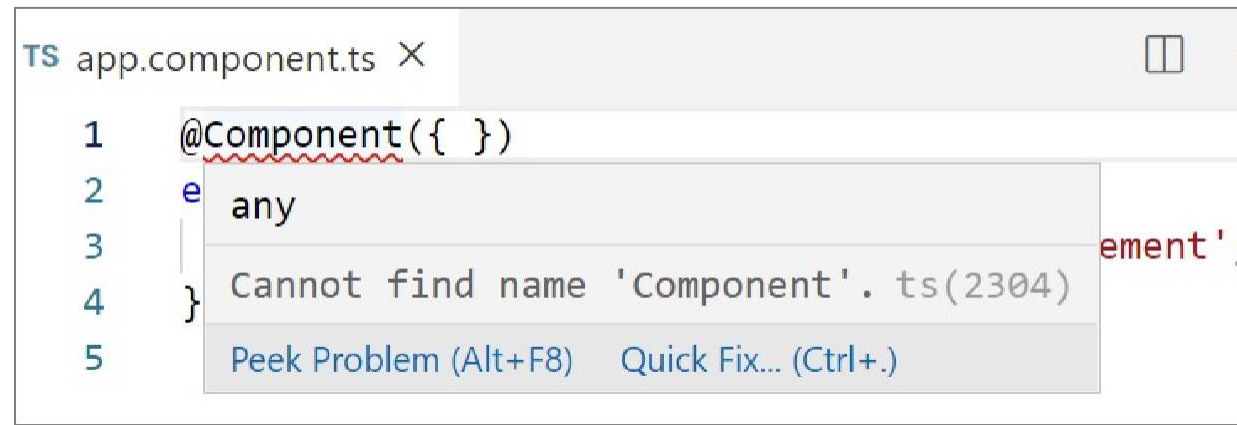
**Es importante la
diferencia entre
mayúsculas y
minúsculas**

```
@NgModule({  
  declarations: [  
    AppComponent,  
    ProductListComponent  
  ],  
  imports: [...],  
  bootstrap: [...]  
})  
export class AppModule { }
```

**Los componentes
deben definirse en
un módulo de
Angular**

Empezando con el editor de código

Comprueba si
hay líneas en
forma de
garabato



```
TS app.component.ts X
1 @Component({ })
2 e any
3 }
4 }
5
```

Cannot find name 'Component'. ts(2304)

Peek Problem (Alt+F8) Quick Fix... (Ctrl+.)

Abrir el
terminal

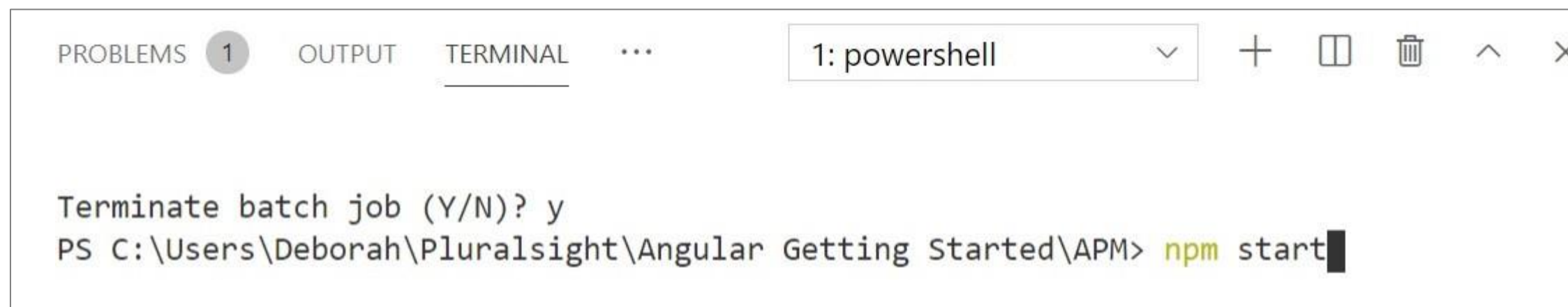


```
PROBLEMS 1 TERMINAL ... 1: node
```

Error: src/app/app.component.ts:1:2 - error TS2304: Cannot find name 'Component'.

```
1 @Component({ })
   ~~~~~
```

Parar
(Ctrl+C) y
reiniciar

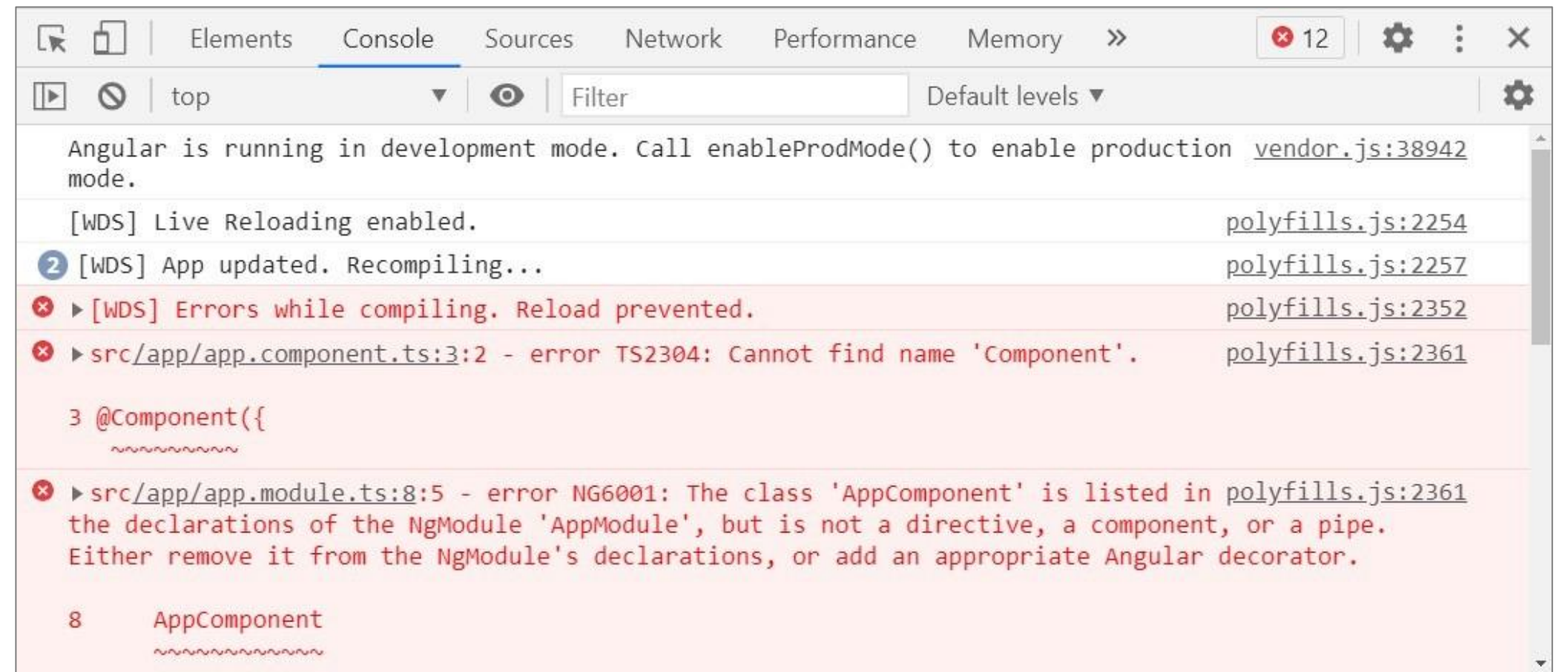


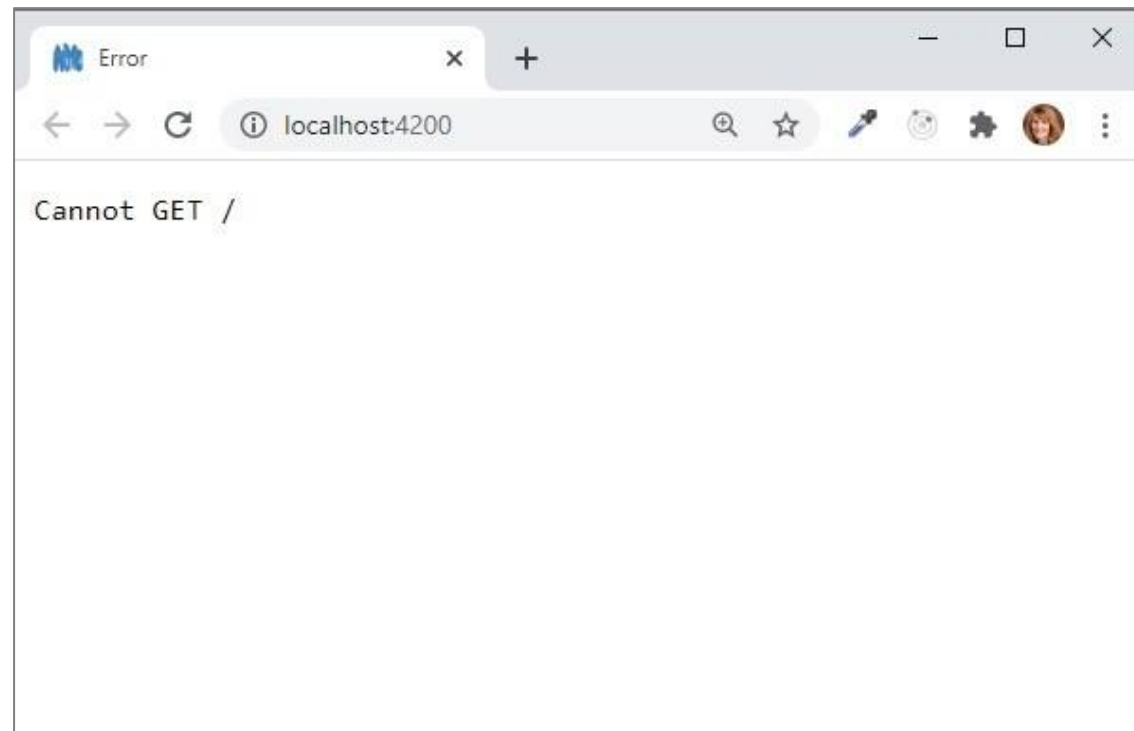
```
PROBLEMS 1 OUTPUT TERMINAL ... 1: powershell
```

Terminate batch job (Y/N)? y

PS C:\Users\Deborah\Pluralsight\Angular Getting Started\APM> npm start

Utiliza las
herramientas de
desarrollador del
navegador





- **A menudo significa que un error impidió que el compilador completara su trabajo**
- **Utiliza el terminal de Visual Studio Code para ver los errores**

Vuelve a comprobar tu código



- **HTML**
 - Etiquetas de cierre
 - Las directivas de Angular distinguen entre mayúsculas y minúsculas
- **TypeScript**
 - Llaves de cierre
 - TypeScript distingue entre mayúsculas y minúsculas

Lista de comprobación de componentes



- Clase -> Código
- Decorador -> Plantilla y metadatos
- Importar lo necesario

Lista de comprobación de componentes: clase



- **Nombre claro**
 - Utiliza PascalCasing
 - Agrega "Component" al nombre

Palabra clave `export`

```
export class AppComponent {  
  pageTitle: string = 'Acme Product Management';  
}
```

Lista de comprobación de componentes: miembros

- **Datos en propiedades**
 - Tipos de datos adecuados
 - Valor por defecto adecuado
 - Utiliza camelCase con la primera letra en minúscula
- **Lógica en los métodos**
 - Utiliza camelCase con la primera letra en minúscula



```
export class AppComponent {  
  showImage: boolean = false;    // property  
  
  toggleImage(): void {          // method  
    this.showImage = true;  
  }  
}
```


Lista de comprobación de componentes: metadatos



- **Decorador del componente**
 - **Prefijo con @; Sufijo ()**
- `selector`: **Nombre del componente en HTML**
 - **Utiliza prefijo para mayor claridad**
- `template`: **HTML de la vista**
 - **Utiliza sintaxis HTML de forma adecuada**

```
@Component ({  
  selector: 'pm-root',  
  template: `  
    <div><h1>{{pageTitle}}</h1></div>  
  `,  
})
```

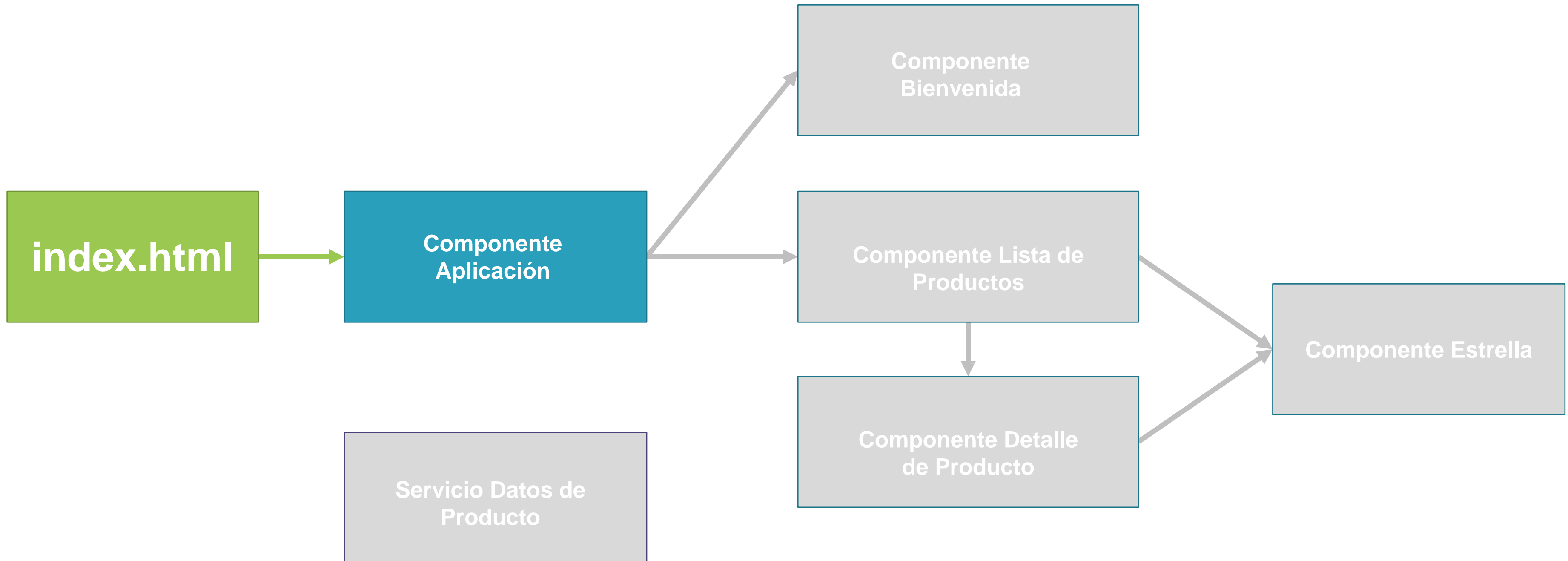
Lista de comprobación de componentes: Sentencia Import



- Define dónde encontrar los miembros que necesitará este componente
- **palabra clave** `import`
- **Nombre de miembro**
 - Especificar correctamente su nombre
- **Ruta**
 - Envolverlo entre comillas simples
 - Especificar correctamente la ruta

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

Arquitectura de la aplicación





A continuación ...

**Plantillas, Interpolación y
directivas**