Documentacion para un proyecto en angular:

Tema: Matrices de 3\*3, 4\*4, 5\*5 y 6\*6.

Pasos para la creación:

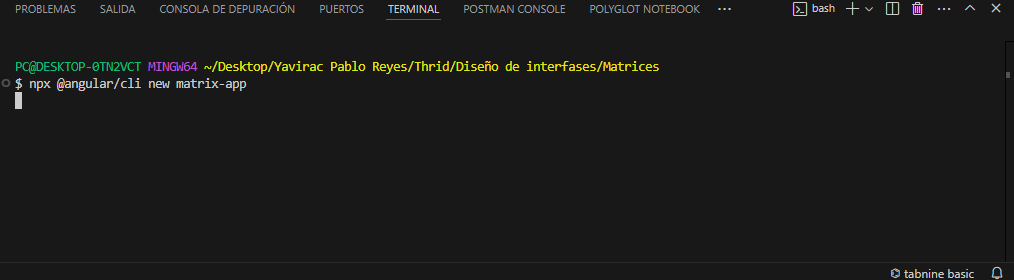
**1. Configuración del proyecto Angular**

Abre Visual Studio Code y sigamos estos pasos:

**a. Crea un nuevo proyecto Angular**

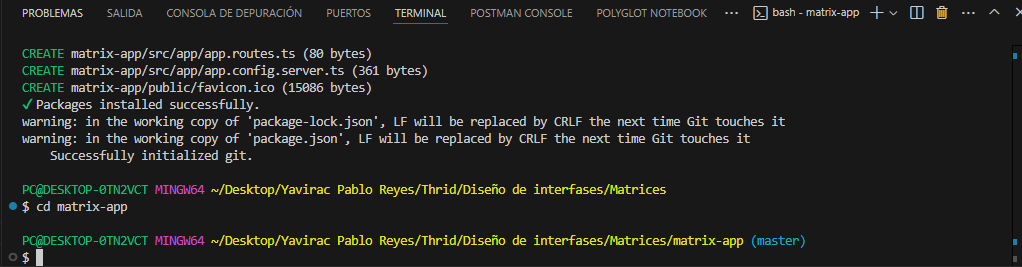
1. Abrimos una terminal integrada (`Ctrl+``) en VS Code.
2. Ejecutamos:

npx @angular/cli new matrix-app



1. Seleccionamos CSS como preprocesador de estilos cuando te lo pida. (Esto facilitará la integración de PrimeNG más adelante).
2. Ingresamos al directorio del proyecto:

cd matrix-app

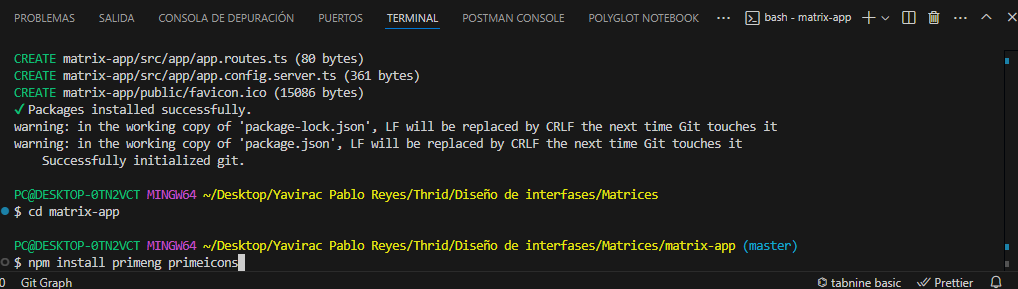


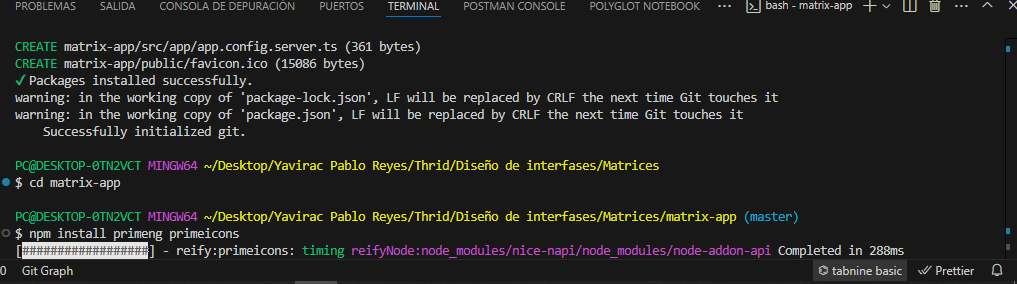
#### b. Instalacion de PrimeNG y sus dependencias:

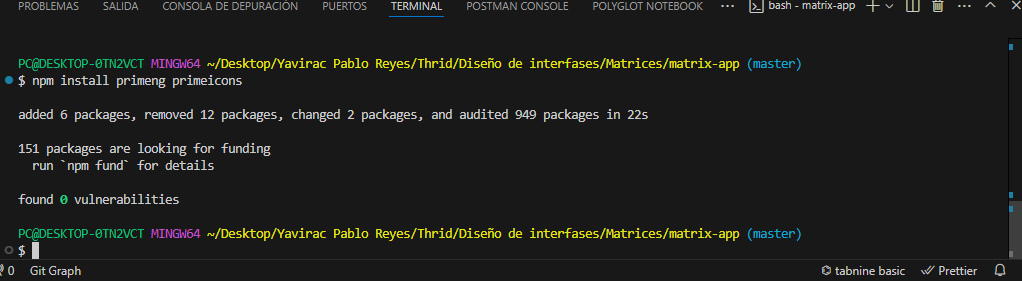
1. Ejecutamos:

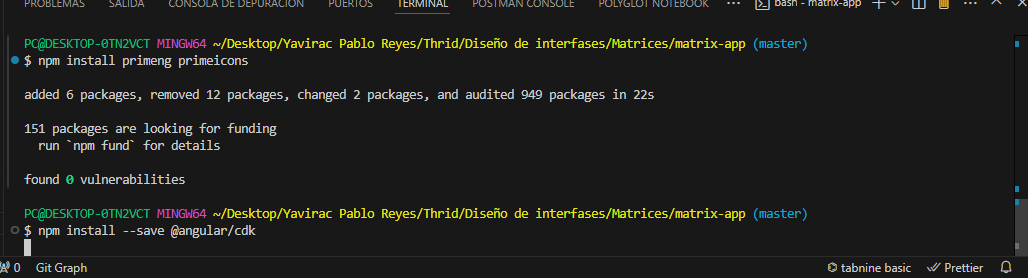
npm install primeng primeicons

npm install --save @angular/cdk

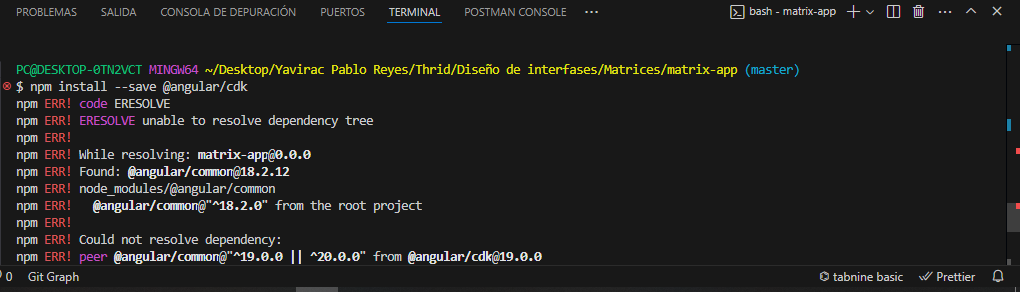






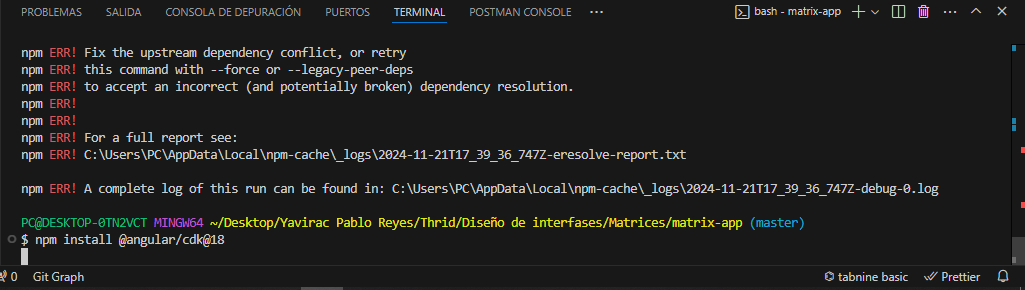
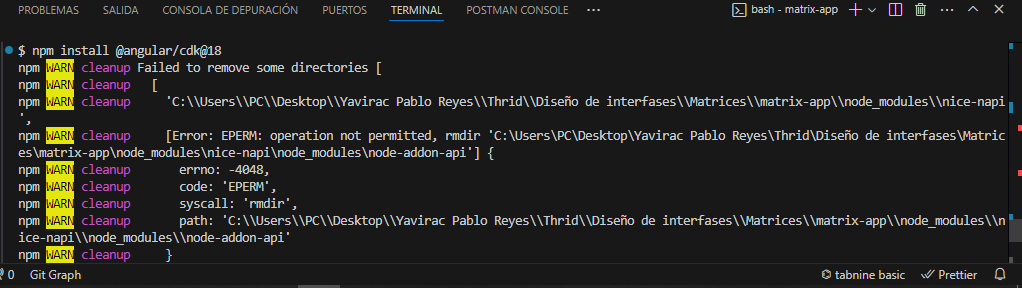


En este comando ocurre un error dependiendo la versión de angular en mi caso es angolar 18 y para importar este comando necesitamos la versión mas reciente:



Entonces vamos a instalar la versión mas reciente:

npm install @angular/cdk@18

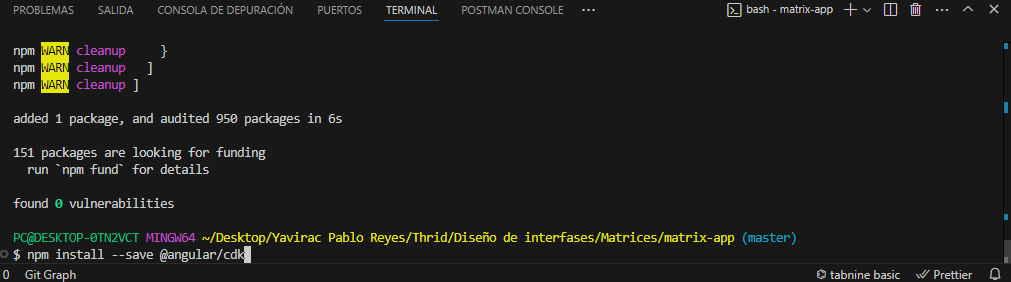
 

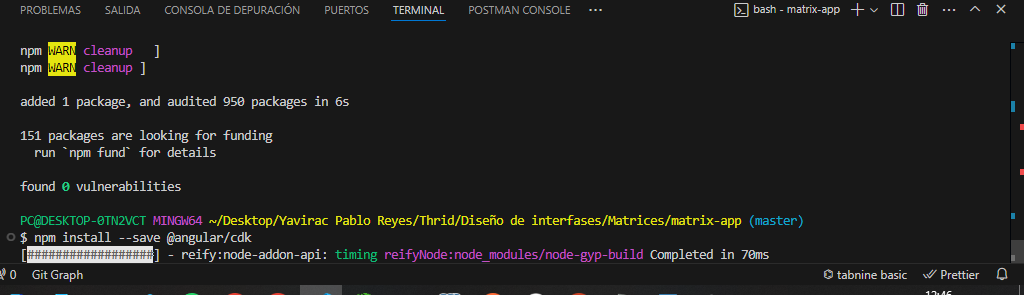
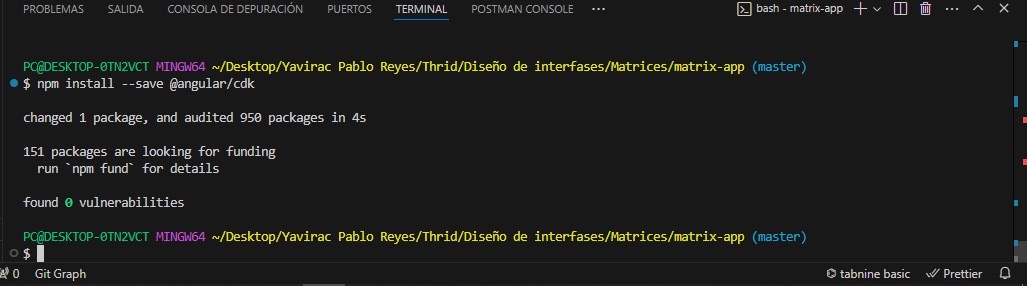
Si queremos podemos actualizar el proyecto a Angular 19 o superior, este problema desaparecerá. Para actualizar:

ng update @angular/cli @angular/core

Pero para propósitos principales del proyecto solo lo haremos en el Angular que acabamos de instalar.

Y ahora si continuamos donde nos quedamos:



#### c. Agregamos PrimeNG en angular.json:

En el archivo angular.json, agregamos las rutas de estilos e iconos:

"styles": [

"src/styles.scss",

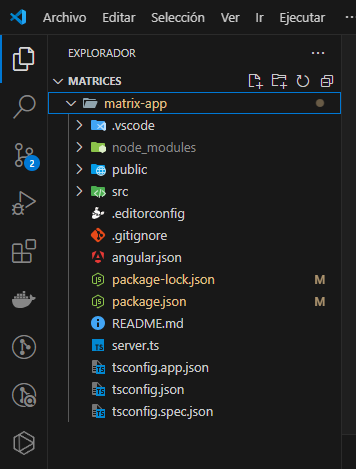
"node\_modules/primeng/resources/themes/saga-blue/theme.css",

"node\_modules/primeng/resources/primeng.min.css",

"node\_modules/primeicons/primeicons.css"

],

"scripts": []

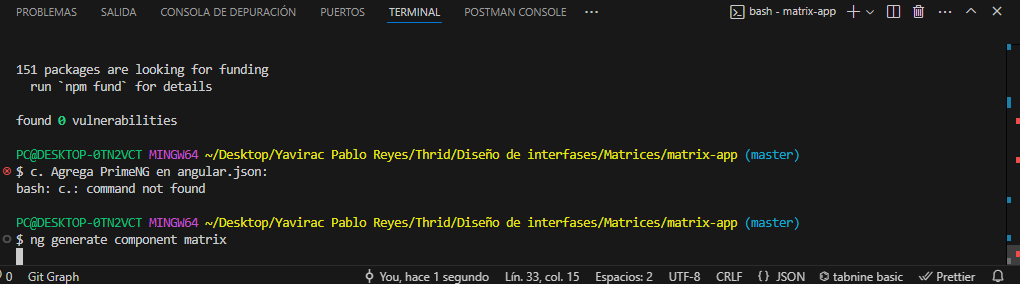
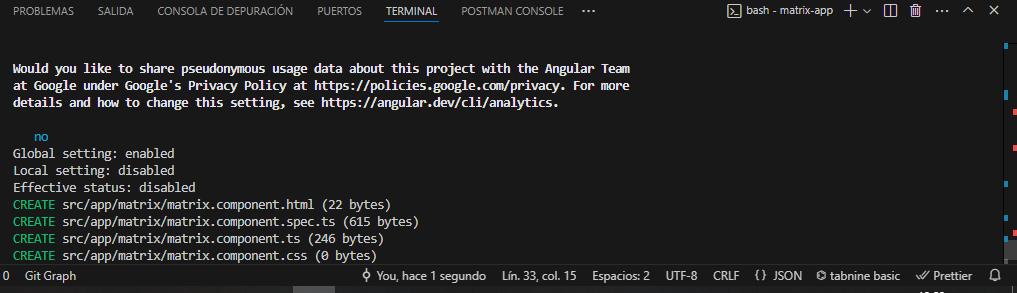




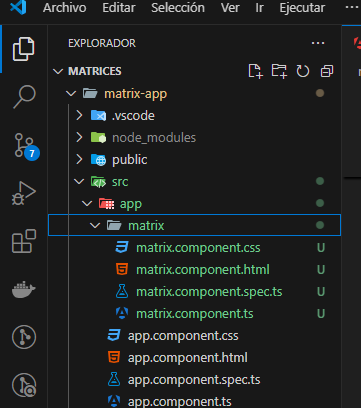
### 2. ****Creación de componentes para matrices.****

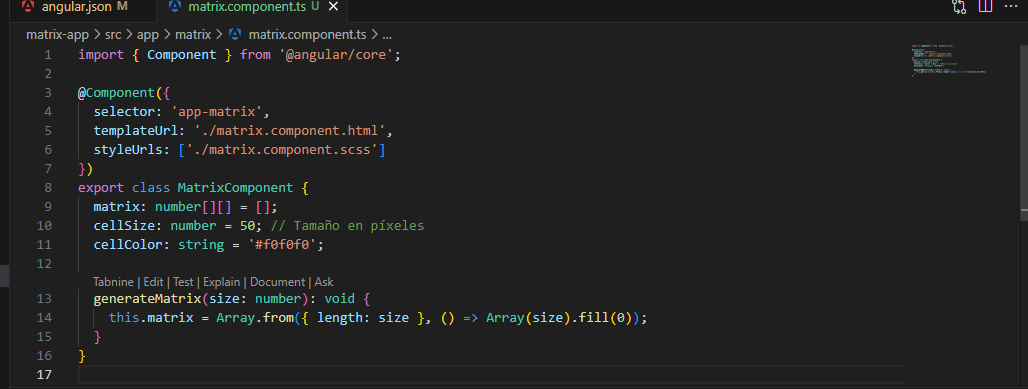
1. Generamos un nuevo componente:

ng generate component matrix

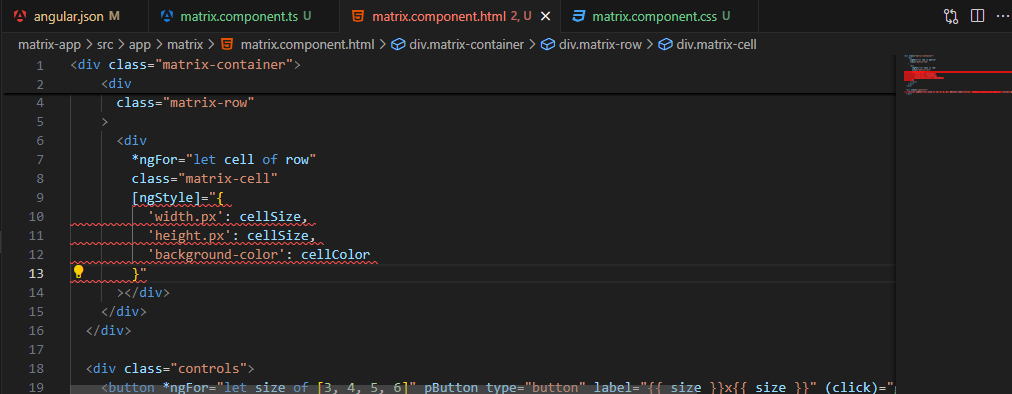
 

1. En el archivo matrix.component.ts, agregamos el siguiente código para generar dinámicamente matrices:





1. En matrix.component.html, utilizamos PrimeNG para mostrar la matriz:



Observamos un problema debido a que Angular no reconoce la directiva ngStyle porque el módulo correspondiente no está importado en la aplicación. La directiva ngStyle forma parte del módulo CommonModule.

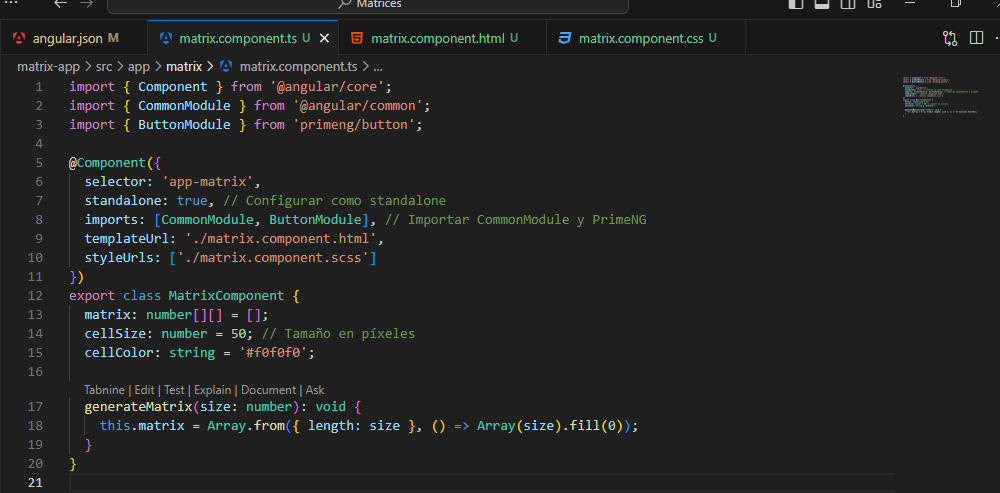
Una solucion seria importar este modulo desde AppModule, pero en nuestro directorio no tenemos el archivo AppModule, eso significa que estamos trabajando con **componentes independientes** , introducidos en Angular 14. En este caso, cada componente debe importar los módulos necesarios por su cuenta.

### Solución para componentes independientes

Editamos el archivo matrix.component.ts para asegurarnos de que importa y utiliza correctamente CommonModule y otros módulos necesarios.

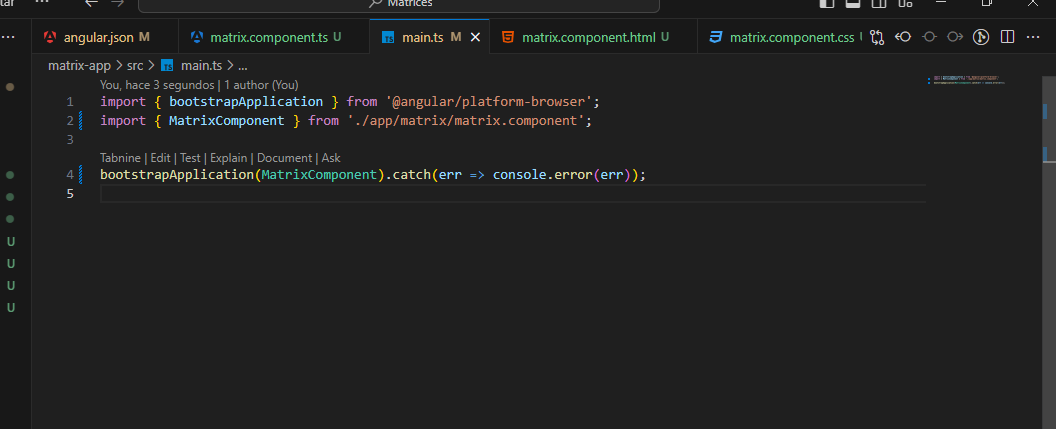
#### Cambios en matrix.component.ts

Nos aseguramos de que su componente se configure como standalone e importamos los módulos que necesita:

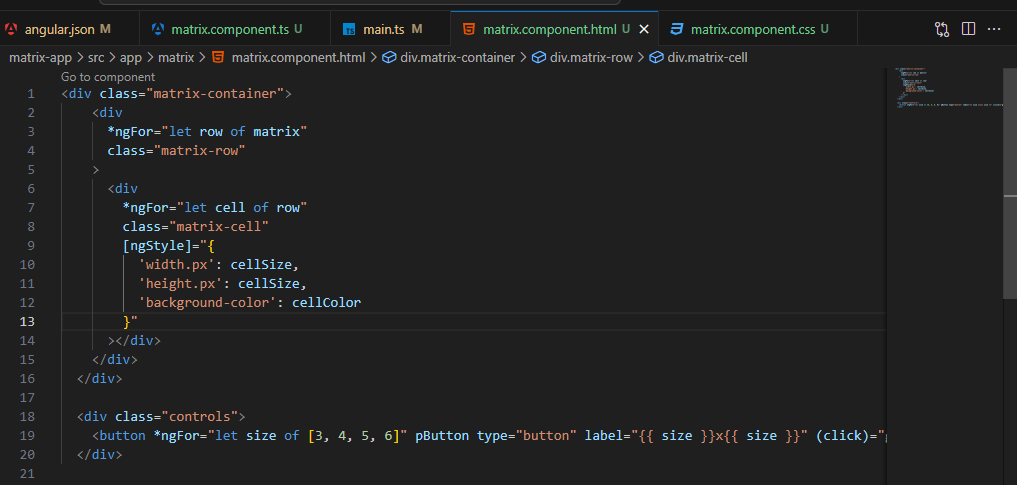


#### Cambios en main.ts

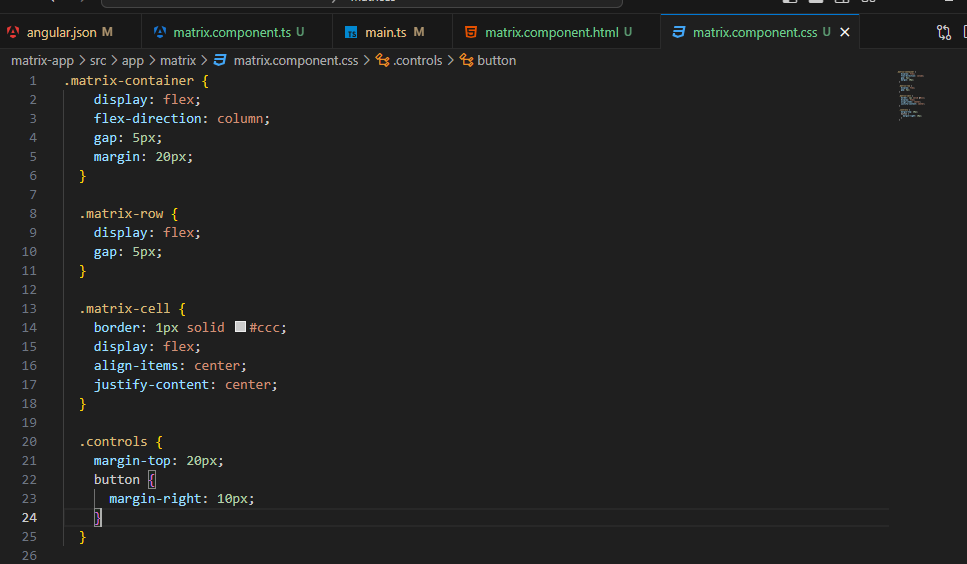
Revisamos el archivo main.ts para asegurarnos de que está cargando su componente correctamente:



Una vez hechos estos cambios ya tenemos arreglado el problema y así podemos continuar:

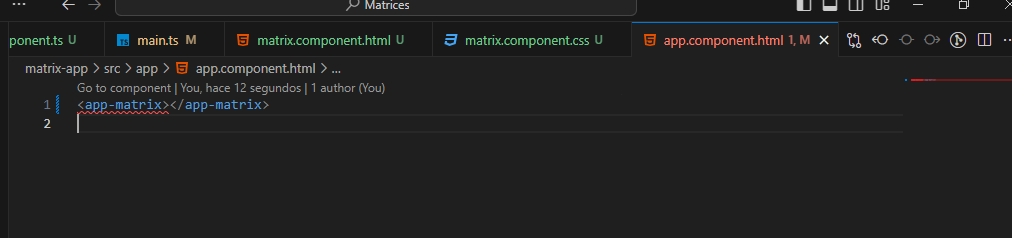


1. En matrix.component.css, se define los estilos:



### 3. ****Integrar el componente al proyecto****

En el archivo app.component.html, agregamos el selector del componente:

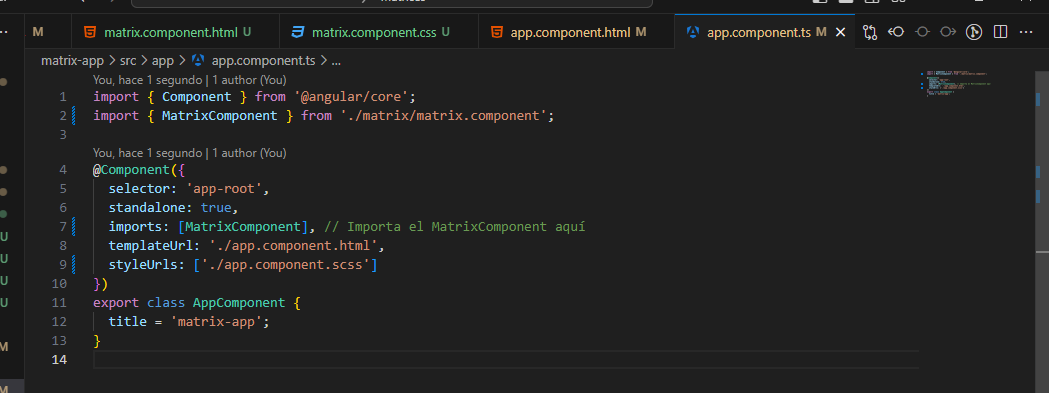


Este error ocurre porque estamos intentando usar el selector <app-matrix> en un componente sin importar específicamente el componente MatrixComponent. Cuando trabajamos con **componentes independientes** en Angular, debemos asegurarnos de importar explícitamente los componentes independientes en cualquier componente que los utilicemos.

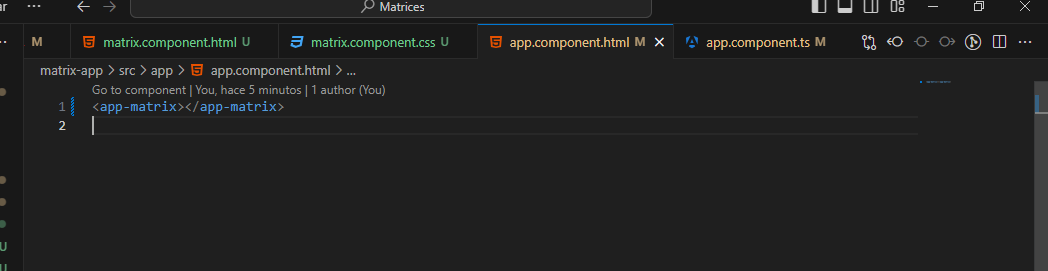
### Solución:

#### Cambios en app.component.ts:

1. Asegúrar de importar el MatrixComponent interior de las importaciones del AppComponent.



Ahora si podemos usar el selector correctamente:



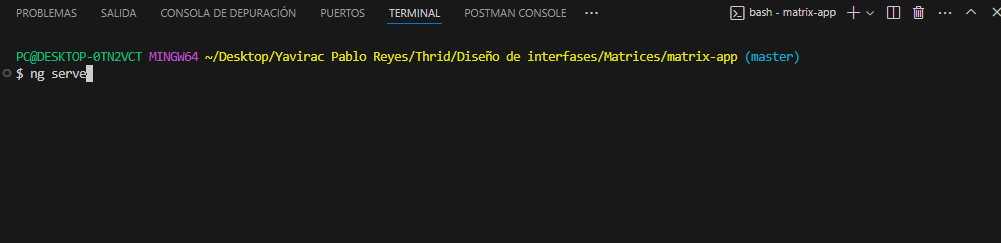
**¡OJO!**

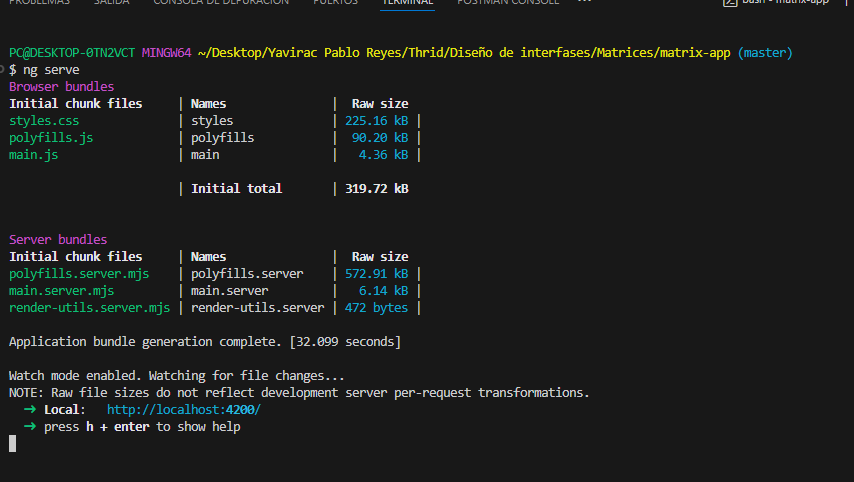
 Componentes **independientes** : Cuando se utilizan componentes independientes, cada componente debe importar explícitamente otros componentes independientes que necesite.

 Importar **MatrixComponent** : Esto asegura que AppComponentreconozca a MatrixComponentcomo un componente válido.

### 4. ****Ejecutamos el proyecto****

1. Iniciamos el servidor de desarrollo:

****

****

1. Abrimos el navegador en http ://localhost :4200 .

### Resultado esperado

Deberíamos ver botones para generar matrices 3x3, 4x4, 5x5, o 6x6. Cada celda tendrá un tamaño fijo y un color uniforme, que podemos personalizar dinámicamente en el código.

