

**Instituto Tecnológico de Oaxaca**

**Materia: Programacion WEB**

# **Trabajo: Consumo de APIs en Angular**

---

*Estudiante:*

*López Reyes Jorge Mauricio*

*Número de Control:*

*19161325*

*Docente:*

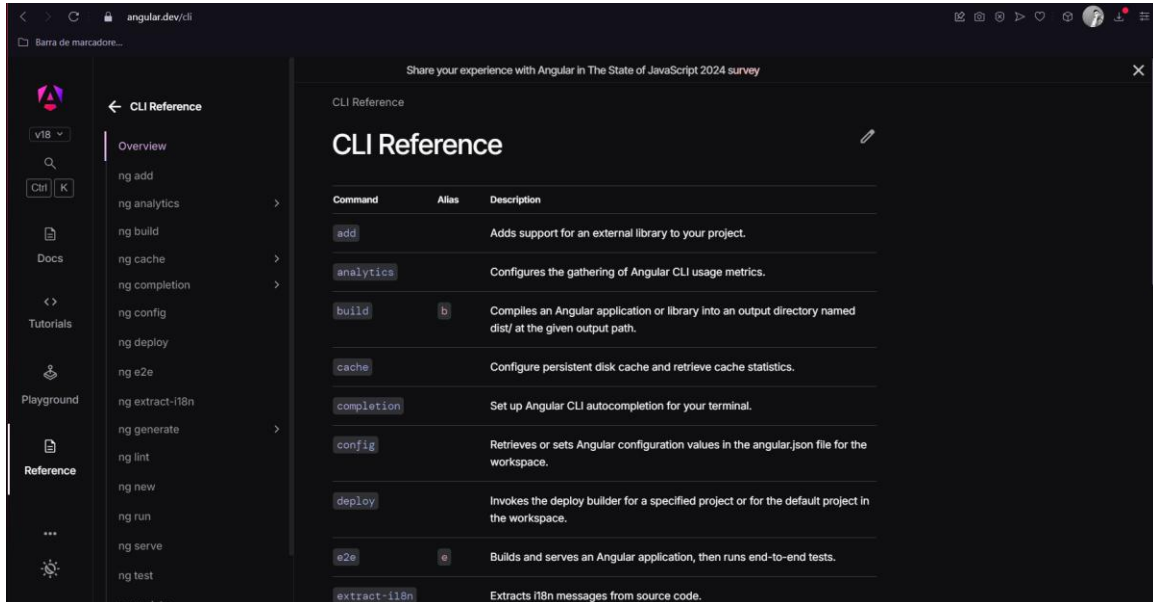
*Martinez Nieto Adelina*

*Año: 2024*

# Consumo de una API en Angular con Cuadrícula y Paginación

## 1. Crear el Proyecto Angular

Usamos Angular CLI para crear un nuevo proyecto Angular. Esto genera la estructura inicial de archivos y carpetas necesarias para nuestra aplicación.



Comando utilizado:  
ng new consumo-api

Este comando crea un proyecto llamado 'consumo-api'. Luego ingresamos a la carpeta del proyecto con:  
cd consumo-api.



## 2. Configurar el Servicio para Consumir la API

Un servicio en Angular es una clase que contiene lógica específica, como conectarse a una API externa. Esto permite mantener nuestro código limpio y organizado.

Comando utilizado:

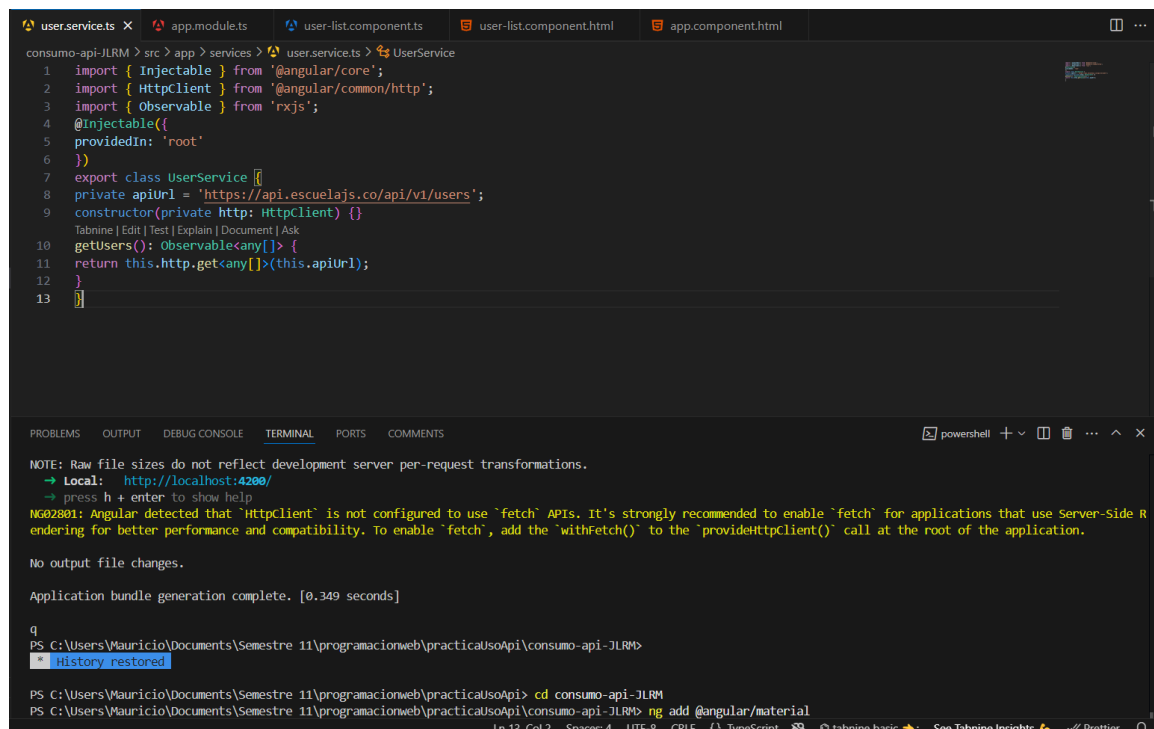
ng generate service services/user

```
PS C:\Users\Mauricio\Documents\Semestre 11\programacionweb\practicaUsaApi> cd consumo-api-JLRM
PS C:\Users\Mauricio\Documents\Semestre 11\programacionweb\practicaUsaApi\consumo-api-JLRM> ng add @angular/material
✔ Determining Package Manager
  > Using package manager: npm
✔ Searching for compatible package version
  > Found compatible package version: @angular/material@18.2.13.
✔ Loading package information from registry
✔ Confirming installation
✔ Installing package
? Choose a prebuilt theme name, or "custom" for a custom theme: Azure/Blue [Preview: https://material.angular.io/?theme=azure-blue]
? Set up global Angular Material typography styles? yes
? Include the Angular animations module? Include and enable animations
UPDATE package.json (1366 bytes)
✔ Packages installed successfully.
UPDATE src/app/app.module.ts (661 bytes)
UPDATE angular.json (3349 bytes)
```

El archivo generado contiene la lógica para obtener datos de la API.

## 3. Configurar HttpClientModule

Angular necesita saber cómo manejar solicitudes HTTP. Para ello, agregamos HttpClientModule en el archivo app.module.ts. Esto asegura que Angular pueda realizar peticiones HTTP a la API.



```
user.service.ts x app.module.ts user-list.component.ts user-list.component.html app.component.html
consumo-api-JLRM > src > app > services > user.service.ts > UserService
1 import { Injectable } from '@angular/core';
2 import { HttpClient } from '@angular/common/http';
3 import { Observable } from 'rxjs';
4 @Injectable({
5   providedIn: 'root'
6 })
7 export class UserService {
8   private apiUrl = 'https://api.escuelajs.co/api/v1/users';
9   constructor(private http: HttpClient) {}
10  getUsers(): Observable<any[]> {
11    return this.http.get<any[]>(this.apiUrl);
12  }
13 }
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
NOTE: Raw file sizes do not reflect development server per-request transformations.
→ Local: http://localhost:4200/
→ press h + enter to show help
NG02801: Angular detected that 'HttpClient' is not configured to use 'fetch' APIs. It's strongly recommended to enable 'fetch' for applications that use Server-Side Rendering for better performance and compatibility. To enable 'fetch', add the 'withFetch()' to the 'provideHttpClient()' call at the root of the application.

No output file changes.

Application bundle generation complete. [0.349 seconds]

q
PS C:\Users\Mauricio\Documents\Semestre 11\programacionweb\practicaUsaApi\consumo-api-JLRM>
History restored
PS C:\Users\Mauricio\Documents\Semestre 11\programacionweb\practicaUsaApi> cd consumo-api-JLRM
PS C:\Users\Mauricio\Documents\Semestre 11\programacionweb\practicaUsaApi\consumo-api-JLRM> ng add @angular/material
```

## 4. Crear el Componente para Mostrar los Usuarios

Un componente en Angular es una pieza de la interfaz de usuario que incluye HTML, CSS y TypeScript. Lo generamos con:

ng generate component components/user-list.

```
8 templateUrl: './user-list.component.html',
9 styleUrls: ['./user-list.component.css']
10 })
11 export class UserListComponent implements OnInit {
12   displayedColumns: string[] = ['id', 'name', 'email', 'role'];
13   dataSource = new MatTableDataSource<{}>();
14
15   @ViewChild(MatPaginator) paginator!: MatPaginator;
16
17   constructor(private userService: UserService) {}
18
19   ngOnInit(): void {
20     this.userService.getUsers().subscribe(data => {
21       this.dataSource.data = data;
22       this.dataSource.paginator = this.paginator;
23     });
24   }
25 }
```

PS C:\Users\mauricio\Documents\Semestre 11\programacionweb\practicas\api\consumo-api-JLRM> ng add @angular/material

Determining Package Manager  
Using package manager: npm  
Searching for compatible package version

El componente se conecta al servicio para obtener los datos de los usuarios y los muestra en la interfaz.

## 5. Mostrar los Datos en una Cuadrícula con Paginación

Se usa Angular Material para implementar una tabla con datos y agregar paginación.

Angular Material se instala con:

ng add @angular/material.

Luego, se configuran MatTableDataSource y MatPaginator para formatear los datos y añadir la funcionalidad de paginación.

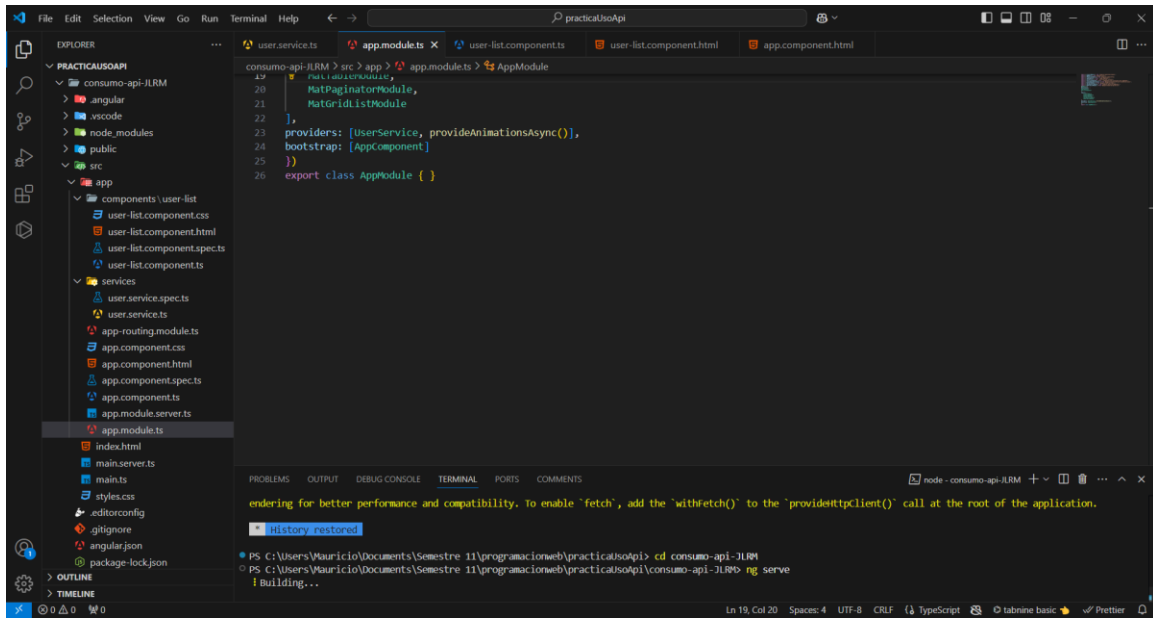
```
1 import { BrowserModule } from '@angular/platform-browser';
2 import { NgModule } from '@angular/core';
3 import { HttpClientModule } from '@angular/common/http';
4 import { AppComponent } from './app.component';
5 import { UserService } from '../services/user.service';
6 import { UserListComponent } from '../components/user-list/user-list.component';
7 import { provideAnimationsAsync } from '@angular/platform-browser/animations/async';
8 import { MatTableModule } from '@angular/material/table';
9 import { MatPaginatorModule } from '@angular/material/paginator';
10 import { MatGridListModule } from '@angular/material/grid-list';
11
12 @NgModule({
13   declarations: [
14     AppComponent,
15     UserListComponent
16   ],
17   imports: [
18     BrowserModule,
19     HttpClientModule,
20     MatTableModule,
21     MatPaginatorModule,
22     MatGridListModule
23   ],
24   providers: [UserService, provideAnimationsAsync()],
25   bootstrap: [AppComponent]
26 })
27 export class AppModule {}
```

PS C:\Users\mauricio\Documents\Semestre 11\programacionweb\practicas\api\consumo-api-JLRM> ng add @angular/material

Determining Package Manager  
Using package manager: npm  
Searching for compatible package version  
Found compatible package version: @angular/material@18.2.13.  
Loading package information from registry  
Confirming installation

## 6. Resultado Final

Ejecutamos el proyecto con `ng serve` y abrimos `http://localhost:4200` en el navegador. Los datos se muestran en una tabla con columnas organizadas y paginación.



localhost:4200

Barra de marcadores...

# User List

ID	Name	Email	Role
1	Jhon	john@mail.com	customer
2	Maria	maria@mail.com	customer
3	Admin	admin@mail.com	admin

Items per page: 5 1 - 3 of 3 |< < > >|