



## Manual de Configuración y Usuario

### Configuración y Desarrollo Angular con Plantilla de Proyecto SFGProv

Dirección de Tecnologías e Información

**Elaborado por:**

Mgs. Norma Arcos  
Mgs. Javier Chillogallo

Versión 1.0  
Fecha de emisión: 8 de mayo de 2024  
Quito – Ecuador

## Contenido

1.	INSTALACION Y DESPLIEGUE DE ANGULAR.....	3
1.1.	Requisitos previos: .....	3
1.2.	Pasos para instalar Angular:.....	3
1.3.	Publicación de un proyecto Angular.....	6
1.3.1	Preparación del proyecto: .....	6
1.3.2	Configuración del servidor Apache: .....	6
1.3.3	Copia de archivos al servidor Apache.....	7
1.3.4	Prueba tu aplicación:.....	7
1.3.5	Verificación y pruebas: .....	7
1.4.	DESPLIEGUE DE LA PLANTILLA DE TRABAJO PARA EL PROYECTO SFGPROV.....	7
2.	FIRMAS.....	12

## 1. INSTALACION Y DESPLIEGUE DE ANGULAR

Angular es un popular framework de desarrollo de aplicaciones web de código abierto desarrollado por Google. Es una plataforma que permite a los desarrolladores crear aplicaciones web dinámicas y de una sola página (SPA, por sus siglas en inglés) de manera eficiente. Angular utiliza TypeScript como lenguaje de programación y sigue el patrón de diseño MVC (Modelo-Vista-Controlador) para organizar y desarrollar aplicaciones web de manera modular y escalable.

Algunas de las características clave de Angular incluyen enlace de datos bidireccional, inyección de dependencias, enrutamiento, manejo de formularios, pruebas unitarias integradas y una estructura modular basada en componentes. Con su potente conjunto de herramientas y su amplia comunidad de desarrolladores, Angular se ha convertido en una opción popular para la creación de aplicaciones web modernas y robustas.

### 1.1. Requisitos previos:

Angular requiere Node.js, que incluye npm (Node Package Manager). Puede descargar e instalar Node.js desde su sitio web oficial.

<https://nodejs.org/en>

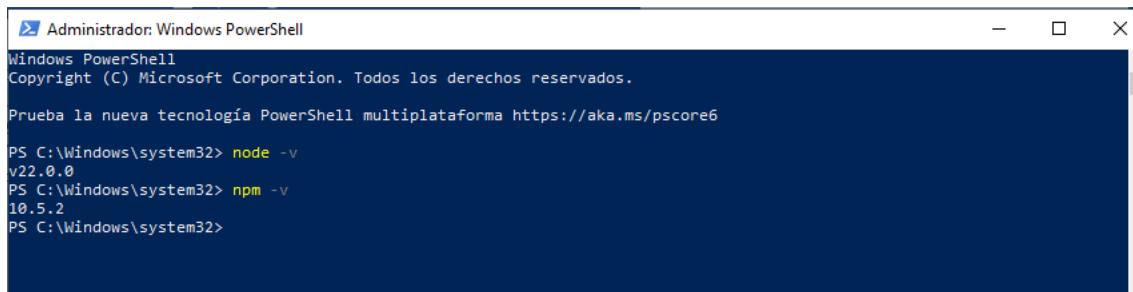
### 1.2. Pasos para instalar Angular:

#### Verificar la instalación de Node.js y npm

Abre tu terminal o línea de comandos y ejecuta los siguientes comandos para verificar que Node.js y npm estén instalados correctamente:

```
node -v  
npm -v
```

Estos comandos mostrarán la versión de Node.js y npm instalada en tu sistema. Si ves las versiones, significa que Node.js y npm están instalados correctamente.



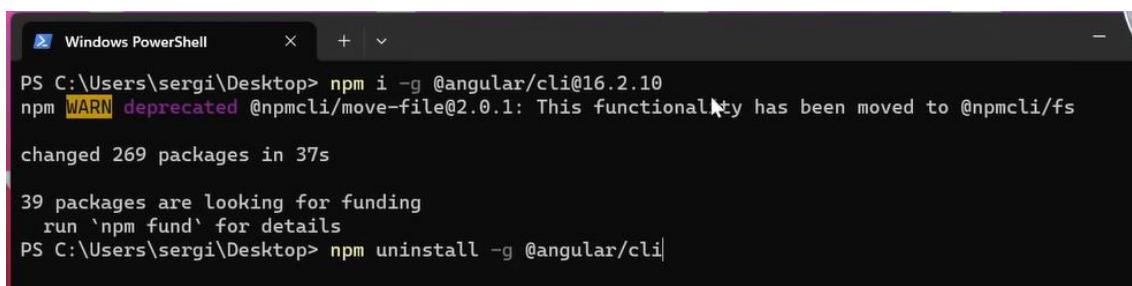
```
Administrator: Windows PowerShell  
Windows PowerShell  
Copyright (C) Microsoft Corporation. Todos los derechos reservados.  
Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6  
PS C:\Windows\system32> node -v  
v22.0.0  
PS C:\Windows\system32> npm -v  
10.5.2  
PS C:\Windows\system32>
```

## Instalar Angular CLI

Una vez que hayas verificado la instalación de Node.js y npm, puedes instalar Angular CLI utilizando el siguiente comando:

```
PS C:\Users\narcos> npm install -g @angular/cli
added 237 packages in 9s
44 packages are looking for funding
  run `npm fund` for details
```

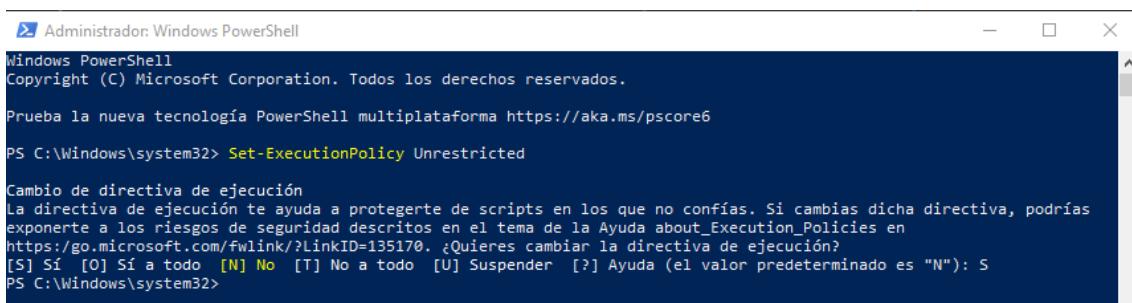
En caso de necesitarse una versión particular del editor CLI se debe ejecutar el siguiente comando:



```
Windows PowerShell
PS C:\Users\sergi\Desktop> npm i -g @angular/cli@16.2.10
npm WARN deprecated @npmcli/move-file@2.0.1: This functionality has been moved to @npmcli/fs
changed 269 packages in 37s
39 packages are looking for funding
  run `npm fund` for details
PS C:\Users\sergi\Desktop> npm uninstall -g @angular/cli
```

Este comando instalará Angular CLI globalmente en el sistema.

Es necesario configurar una política sin restricciones para el cual se debe ejecutar el siguiente comando:



```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Windows\system32> Set-ExecutionPolicy Unrestricted

Cambio de directiva de ejecución
La directiva de ejecución te ayuda a protegerte de scripts en los que no confías. Si cambias dicha directiva, podrías
exponerse a los riesgos de seguridad descritos en el tema de la Ayuda about_Execution_Policies en
https://go.microsoft.com/fwlink/?LinkId=135170. ¿Quieres cambiar la directiva de ejecución?
[S] Sí [O] Sí a todo [N] No [T] No a todo [U] Suspender [?] Ayuda (el valor predeterminado es "N"): S
PS C:\Windows\system32>
```

## Crear un nuevo proyecto Angular

Después de instalar Angular CLI, puedes crear un nuevo proyecto Angular ejecutando el siguiente comando en tu terminal:

```
ng new nombre-del-proyecto
```

Reemplaza "nombre-del-proyecto" con el nombre que deseas para tu proyecto. Este comando generará una estructura de directorios y archivos para tu nueva aplicación Angular.

### Navegar al directorio del proyecto

Una vez que se haya creado el proyecto, navega al directorio del proyecto recién creado utilizando el siguiente comando:

```
cd nombre-del-proyecto
```

Se debe mencionar que al crear componentes en CLI se debe considerar lo siguiente:

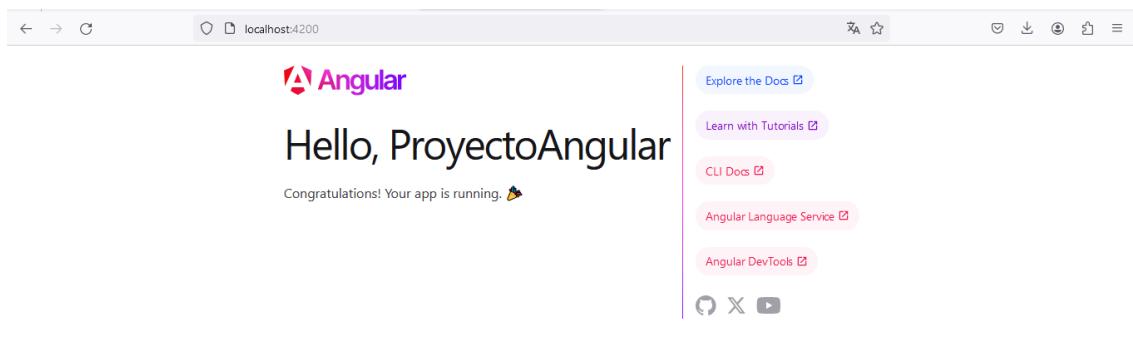


### Ejecutar la aplicación Angular

Para ejecutar tu aplicación Angular localmente, utiliza el siguiente comando:

```
ng serve ó ng s
```

Este comando compilará tu aplicación y la servirá en un servidor local. Por defecto, la aplicación estará disponible en <http://localhost:4200>.



### 1.3. Publicación de un proyecto Angular

#### 1.3.1 Preparación del proyecto:

Antes de publicar tu aplicación Angular en Apache, asegúrate de haber realizado las siguientes tareas:

##### Compila la aplicación

Ejecuta el siguiente comando en tu terminal para compilar tu aplicación Angular:

```
ng build --prod
```

Este comando compilará tu aplicación para producción y generará los archivos optimizados en el directorio dist/.

Revisa la configuración de producción: Verifica que la configuración de producción en angular.json esté correctamente configurada para tus necesidades específicas.

#### 1.3.2 Configuración del servidor Apache:

##### Habilita el módulo de reescritura

Si aún no está habilitado, asegúrate de activar el módulo de reescritura de Apache. Puedes hacerlo ejecutando el siguiente comando en tu terminal:

```
sudo a2enmod rewrite
```

##### Configura el archivo .htaccess

Crea un archivo .htaccess en el directorio raíz de tu proyecto Angular (junto con el archivo index.html generado en la carpeta dist/) y agrega las siguientes reglas de reescritura:

```
RewriteEngine On
RewriteBase /
RewriteRule ^index\.html$ - [L]
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule . /index.html [L]
```

Estas reglas de reescritura asegurarán que todas las solicitudes se enruten correctamente a tu aplicación Angular.

### 1.3.3 Copia de archivos al servidor Apache

#### Copia los archivos generados

Copia los archivos generados en el directorio dist/ de tu proyecto Angular al directorio raíz de tu servidor Apache.

#### Configuración adicional (opcional):

**Configuración de virtual hosts:** Si deseas alojar tu aplicación Angular en un dominio específico, puedes configurar un virtual host en Apache para ello. Consulta la documentación de Apache para obtener más información sobre cómo configurar virtual hosts.

### 1.3.4 Prueba tu aplicación:

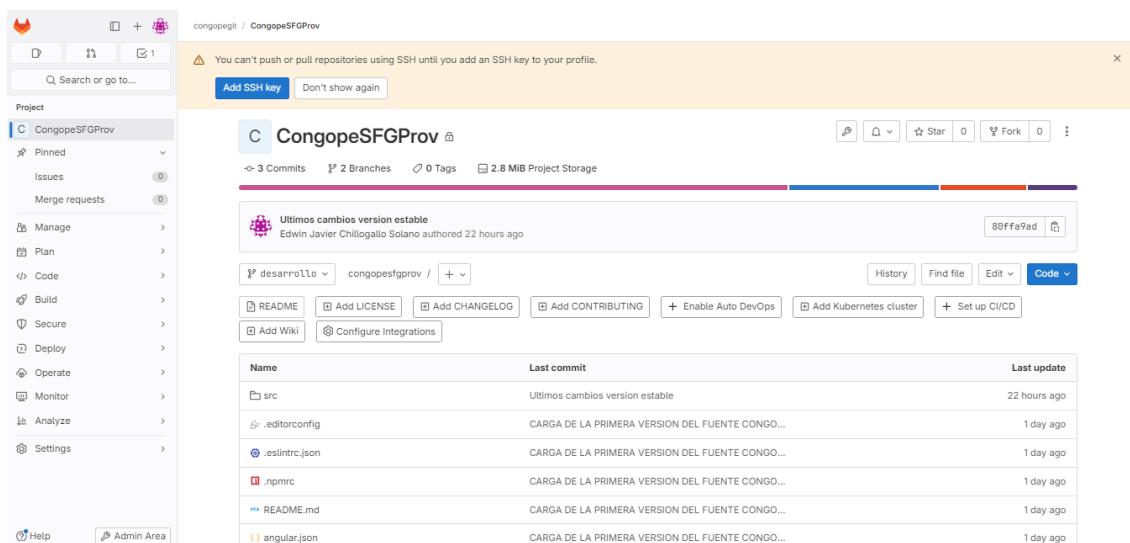
Una vez que hayas configurado Apache y copiado los archivos de tu aplicación Angular, puedes acceder a tu aplicación en un navegador web navegando a la dirección URL de tu servidor Apache.

### 1.3.5 Verificación y pruebas:

Después de publicar tu aplicación, asegúrate de realizar pruebas exhaustivas para verificar que todo funcione correctamente en el entorno de producción. Abre la aplicación en diferentes dispositivos y navegadores para garantizar una experiencia uniforme para tus usuarios.

## 1.4. DESPLIEGUE DE LA PLANTILLA DE TRABAJO PARA EL PROYECTO SFGPROV

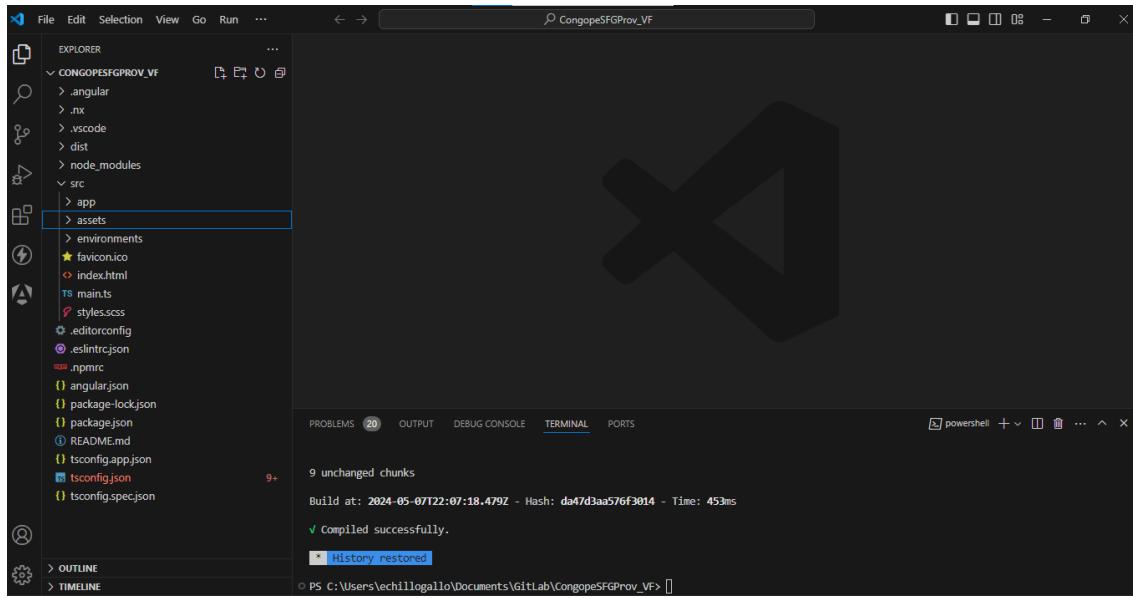
En el repositorio git, provisto para el Congope se habilitó el código fuente para el uso del equipo de desarrollo:



The screenshot shows the GitHub repository 'CongopeSFGProv'. The repository has 3 commits, 2 branches, and 0 tags. It contains files like README, .editorconfig, .eslintrc.json, .npmrc, README.md, and angular.json. A message at the top says: 'You can't push or pull repositories using SSH until you add an SSH key to your profile.' There are buttons to 'Add SSH key' and 'Don't show again'.

Para el desarrollo se recomienda utilizar el editor de código fuente Visual Studio Code.

En el proyecto se abre la carpeta raíz del proyecto



Una vez en el proyecto se abre un terminal y se ejecuta el siguiente comando:

```
npm install
```

Este comando habilitará las librerías necesarias para que el proyecto funcione adecuadamente.

Estructura base del proyecto:

```
main/
├── .angular
├── node_modules
└── src/
    ├── app/
    │   ├── authentication/
    │   ├── config/
    │   ├── core/
    │   ├── layout/
    │   ├── shared/
    │   ├── app-routing.module.ts
    │   ├── app.component.html
    │   └── app.componentscss
```

```
|   ├── app.component.spec.ts
|   ├── app.component.ts
|   └── app.module.ts
|   ├── assets/
|   |   └── .gitkeep
|   ├── environments/
|   |   ├── environment.development.ts
|   |   └── environment.ts
|   ├── index.html
|   ├── main.ts
|   └── styles.scss
├── .eslintrc.json
└── angular.json
```

package.json  
tsconfig.app.json  
tsconfig.json  
tsconfig.spec.json

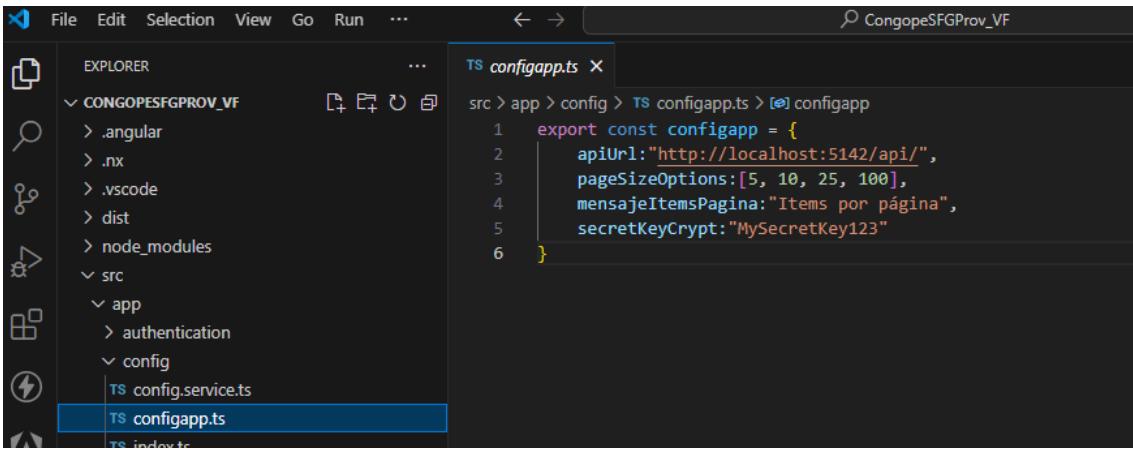
## Descripción de la estructura base

Carpeta	Descripción
authentication	Todas las páginas de autenticación se colocan aquí, como inicio de sesión, registro, etc.
config	El archivo de configuraciones del tema de plantilla config.service.ts en esta carpeta. Puede cambiar la configuración del tema como la plantilla clara u oscura desde aquí.
core	El módulo central es un módulo que creamos para definir servicios comunes, modelos, interceptor y guardia. Los servicios definidos en el módulo principal se instancian una vez.
layout	Aquí puede encontrar módulos relacionados con el diseño, como el encabezado, la barra lateral, la barra lateral de configuración, etc.
shared	Creado para definir componentes comunes y otros módulos que necesitamos importar dentro de todos los demás módulos de nuestra aplicación, por lo que no es necesario importarlos una y otra vez (como los componentes de material). Solo necesitamos

Carpeta	Descripción
	agregar el módulo Shared a cada módulo para acceder a los componentes comunes.

## Funcionamiento de la plantilla

La plantilla tiene una configuración básica en la que se configura aspectos básicos de la aplicación

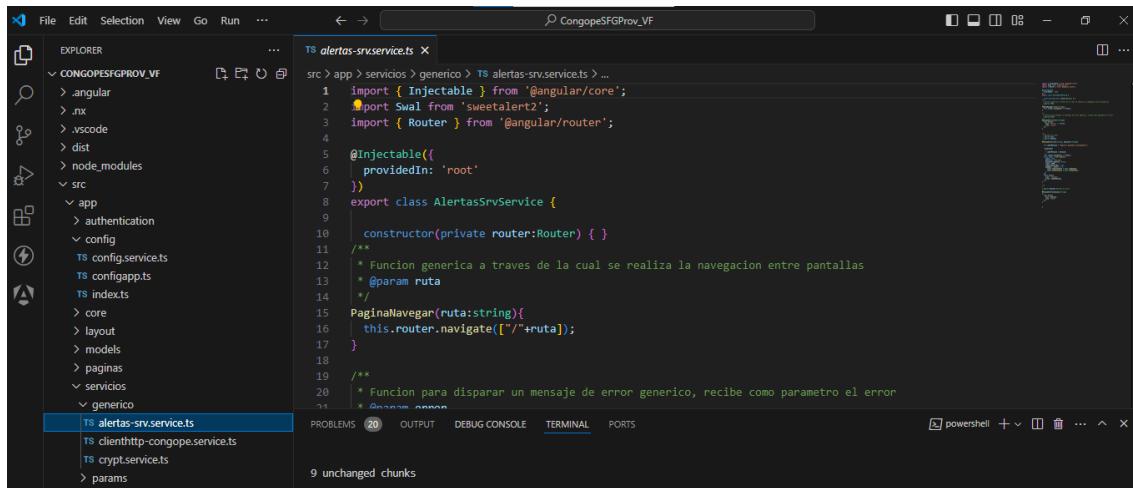


```
src > app > config > configapp.ts > configapp
1  export const configapp = {
2    apiUrl:"http://localhost:5142/api/",
3    pageSizeOptions:[5, 10, 25, 100],
4    mensajeItemsPagina:"Items por página",
5    secretKeyCrypt:"MySecretKey123"
6 }
```

En este archivo se realizan las configuraciones básicas de la aplicación como la Url, básica para el api y el código secreto para la encriptación y desencriptación del código.

## Carpeta servicios

Esta carpeta mantiene los archivos genéricos para el funcionamiento de la aplicación.



```

src > app > servicios > generico > alertas-srvservice.ts ...
1 import { Injectable } from '@angular/core';
2 import Swal from 'sweetalert2';
3 import { Router } from '@angular/router';
4
5 @Injectable({
6   providedIn: 'root'
7 })
8 export class AlertasSrvService {
9
10   constructor(private router:Router) { }
11 /**
12  * Funcion generica a traves de la cual se realiza la navegacion entre pantallas
13  * @param ruta
14  */
15 PaginaNavegar(ruta:string){
16   this.router.navigate(['/'+ruta]);
17 }
18
19 /**
20  * Funcion para disparar un mensaje de error generico, recibe como parametro el error
21  * @param error

```

Si se requiere agregar mas configuraciones genéricas las realizaremos en esta carpeta.

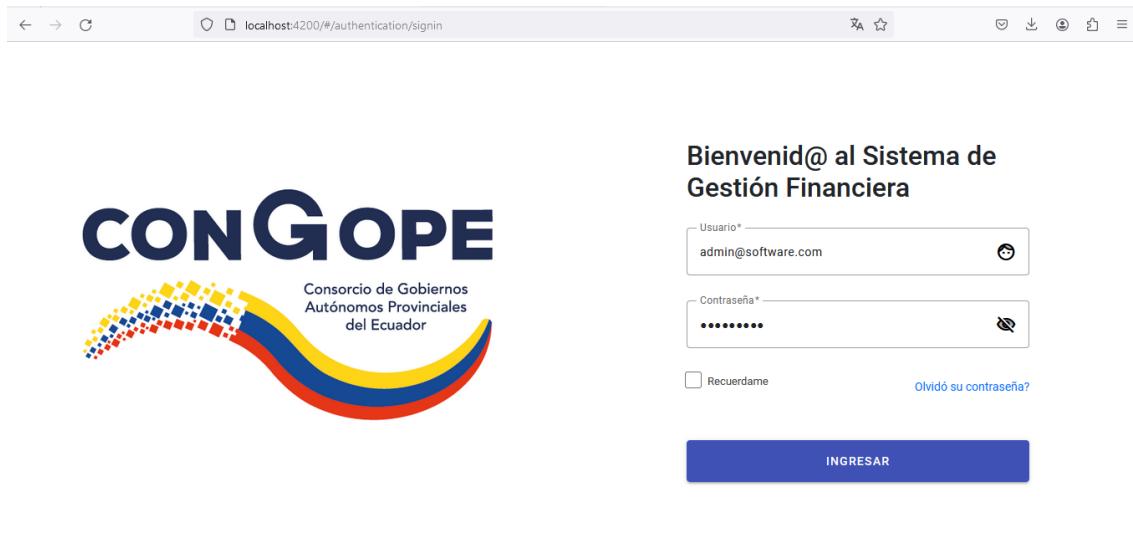
## Carpeta paginas

En esta carpeta se ubican los archivos para la visualización de las pantallas, el ejemplo básico es la pantalla tipo comprobante, el cual contiene un ejemplo CRUD completa.

## Funcionamiento de la plantilla

A continuación, se muestra el funcionamiento del proyecto base de Angular.

### LOGIN



Bienvenid@ al Sistema de Gestión Financiera

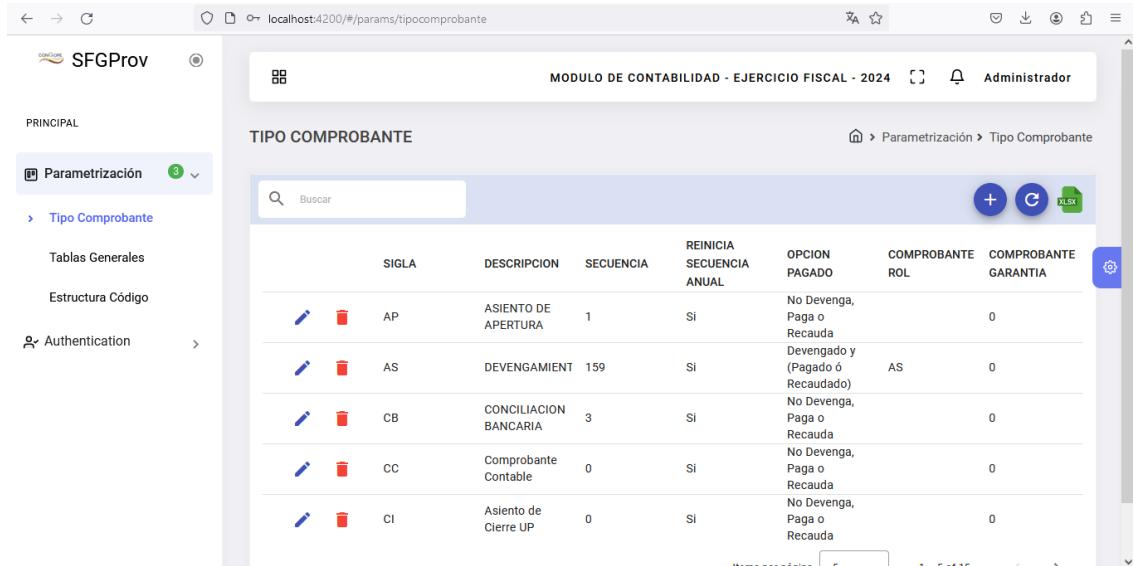
Usuario\*  
admin@software.com

Contraseña\*  
••••••••

Recuerdame [Olvidó su contraseña?](#)

**INGRESAR**

## PANTALLA BASE TIPO COMPROBANTE



SIGLA	DESCRIPCION	SECUENCIA	REINICIA SECUENCIA ANUAL	OPCION PAGADO	COMPROBANTE ROL	COMPROBANTE GARANTIA
AP	ASIENTO DE APERTURA	1	Si	No Devenga, Paga o Recauda		0
AS	DEVENGAMIENT	159	Si	Devengado y (Pagado ó Recaudado)	AS	0
CB	CONCILIACION BANCARIA	3	Si	No Devenga, Paga o Recauda		0
CC	Comprobante Contable	0	Si	No Devenga, Paga o Recauda		0
CI	Asiento de Cierre UP	0	Si	No Devenga, Paga o Recauda		0

El proyecto se encuentra disponible en el repositorio GitLab para su personalización mas a detalle por parte del personal del Congope.

## 2. FIRMAS

	CARGO	FIRMA	FECHA
<b>ELABORADO POR</b>	Mgs. Javier Chillogallo <b>Profesional Contratado</b>		08-05-2024
<b>ELABORADO POR</b>	Mgs. Norma Arcos <b>Profesional Contratado</b>		08-05-2024