

## Lab - Notificación con SweetAlert2 en Angular 12

Este tutorial te explica cómo mostrar una alerta o notificación emergente en Angular 12 con el paquete npm SweetAlert2. Se usará **sweetalert2** en angular por completo, y aprenderemos a mostrar los cuadros de alerta con información centrada en el usuario. Seguiremos todo el proceso de forma gradual y constante para instalar SweetAlert2 en angular.

Si quieres proporcionar información a tu usuario desde el punto de vista de UX, entonces este lab de SweetAlert2 en angular será útil para el proyecto.

SweetAlert2 es una poderosa biblioteca, y ofrece un hermoso, sensible, personalizable, accesible, como reemplazo de las cajas emergentes de JavaScript con absolutamente cero dependencias necesarias.

### Prerequisite

- Node
- npm
- SweetAlert2 Pacakage
- Angular CLI
- Latest Angular version
- Code Editor or IDE

El orden de precedencia comienza con la instalación de Node.js y NPM en su máquina de desarrollo local. Si no tienes las herramientas mencionadas instaladas, entonces revisa este artículo para comprender el proceso de instalación de Node y NPM:

Ejecuta el siguiente comando, responde a algunas preguntas que te hace el CLI de angular.

```
ng new angular-sweetalert-demo
```

```
? Would you like to add Angular routing? No
```

```
? Which stylesheet format would you like to use?
```

```
CSS
```

```
> SCSS      [ https://sass-lang.com/documentation/syntax#scss ]
```

```
Sass      [ https://sass-lang.com/documentation/syntax#the-indented-syntax ]
```

```
Less      [ http://lesscss.org ]
```

```
Stylus [ http://stylus-lang.com
```

Puedes optar por cualquier formato de estilo como SCSS, SASS o LESS para tu proyecto.

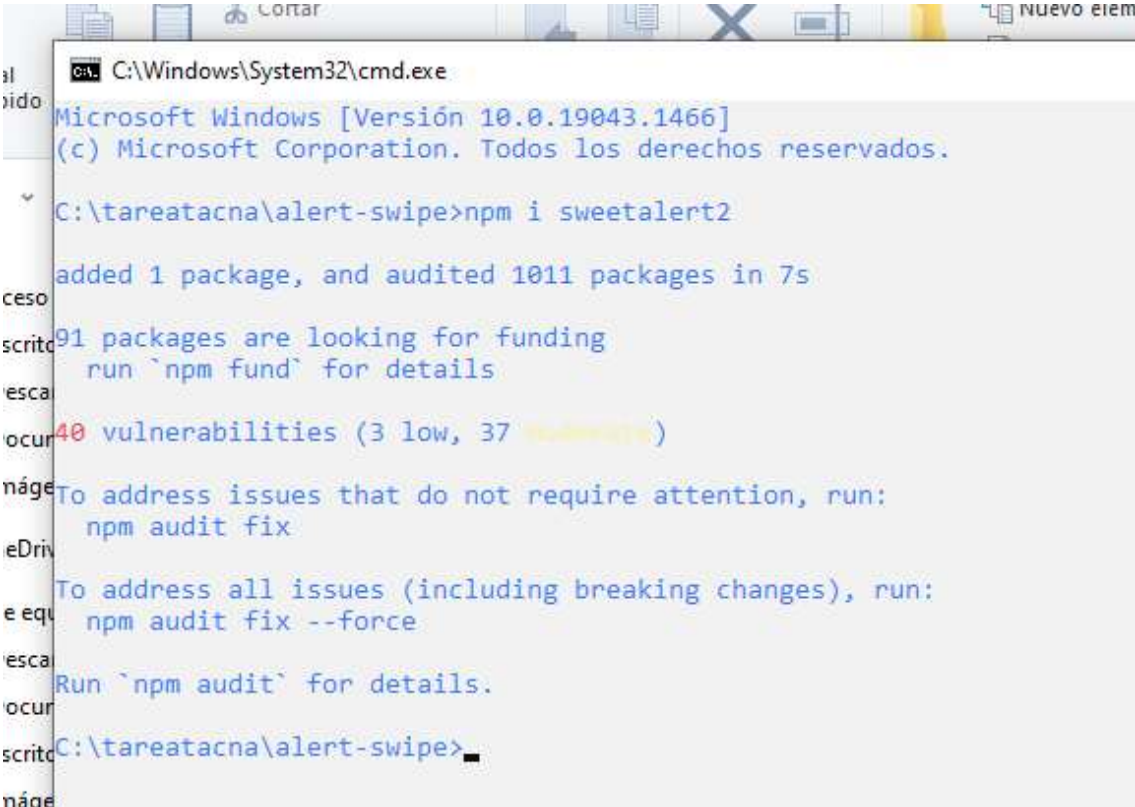
Finalmente, la instalación se ha completado. Entra en la carpeta del proyecto..

```
cd ng new angular-sweetalert-demo
```

## Instalar SweetAlert2 en Angular 12

Instala el paquete npm/sweetalert2 para mostrar hermosas notificaciones y alertas en angular 12. Ejecute el comando en su terminal:

```
npm i/sweetalert2
```

A screenshot of a Windows Command Prompt window. The title bar shows 'C:\Windows\System32\cmd.exe'. The window content displays the following text: 'Microsoft Windows [Versión 10.0.19043.1466]', '(c) Microsoft Corporation. Todos los derechos reservados.', 'C:\tareatacna\alert-swipe>npm i/sweetalert2', 'added 1 package, and audited 1011 packages in 7s', '91 packages are looking for funding', 'run `npm fund` for details', '40 vulnerabilities (3 low, 37 moderate)', 'To address issues that do not require attention, run:', 'npm audit fix', 'To address all issues (including breaking changes), run:', 'npm audit fix --force', 'Run `npm audit` for details.', and 'C:\tareatacna\alert-swipe>'. On the left side of the image, there are some vertical text labels: 'al', 'ido', 'ceso', 'scrito', 'esca', 'ocur', 'nág', 'eDriv', 'e equ', 'esca', 'ocur', 'scrito', 'nág'.

Para mostrar notificaciones atractivas a los usuarios, primero hay que incorporar la ruta CSS de/sweetalert2 en el archivo angular.json.

```
"styles": [  
  "src/styles.css",  
  "node_modules/sweetalert2/src/sweetalert2.scss"  
],  
....
```

```

    "assets": [
      "src/favicon.ico",
      "src/assets"
    ],
    "styles": [
      "src/styles.css",
      "node_modules/sweetalert2/src/sweetalert2.scss"
    ],
    "scripts": []
  },
  "configurations": {

```

### Register SweetAlert in App Component

En este paso, registraremos los imperativos necesarios (servicios del paquete) para invocar los mensajes de notificación a través del paquete sweetalert2.

Aquí están los métodos que necesitamos definir para mantener el orden de precedencia para abrir la alerta SweetAlert en angular en el evento de clic.

- tinyAlert()
- successNotification()
- alertConfirmation()

Coloque el siguiente código en el archivo **app/src/app.component.ts**

```

import { Component, OnInit } from '@angular/core';
import Swal from 'sweetalert2/dist/sweetalert2.js';

```

```

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})

```

```

export class AppComponent implements OnInit {

```

```

  title = 'alert-swipe';

```

```
ngOnInit(): void {
    console.log('Ciclo de vida con respuesta espontanea.');
```

```
}

tinyAlert(){
    Swal.fire('Sigamos!');
}

successNotification(){
    Swal.fire('Que tal', 'Han sido informados!', 'success')
}

alertConfirmation(){
    Swal.fire({
        title: 'Esta seguro?',
        text: 'Este proceso es irreversible.',
        icon: 'warning',
        showCancelButton: true,
        confirmButtonText: 'Si, adelante.',
        cancelButtonText: 'No, Lo pensare'
    }).then((result: { value: any; dismiss: any; }) => {
        if (result.value) {
            Swal.fire(
                'Eliminado!',
                'Producto eliminado correctado.',
                'success'
            )
        } else if (result.dismiss === Swal.DismissReason.cancel) {
            Swal.fire(
                'Cancelado',
```

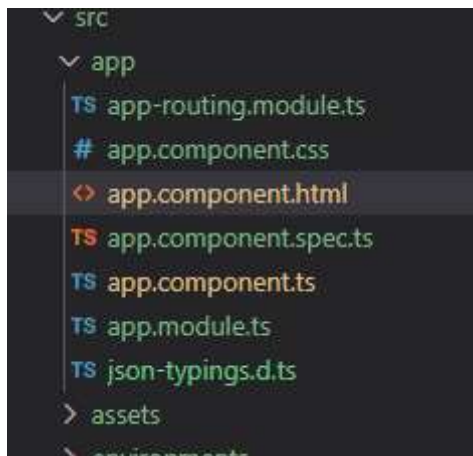
```

        'Producto aún en nuestra base de datos.),'
        'error'
    )
}
}))
}
}

```

### Nota

Es casi seguro que la inclusión que el archivo que representa a la librería saldrá error, debido a que solo un archivo js, por esta razón el procedimiento para resolver este problema será crear el archivo **json-typing.d.ts** en la carpeta ts.



Luego de ello, se colocara el siguiente código:

```
declare module 'sweetalert2/dist/sweetalert2.js';
```

```
declare module 'sweetalert2/dist/sweetalert2.js';
```

grabando este código, se resolverá este problema.

### Mostrando Notificación

En este último paso, vamos a mostrar de forma espontánea Alertas y Notificaciones Popup en Angular con la librería SweetAlert y los métodos personalizados que hemos desarrollado en el paso anterior.

Colocar el código en el archivo **app/src/app.component.html**.

```
<button (click)="tinyAlert()">Notificacion Simple</button>
```

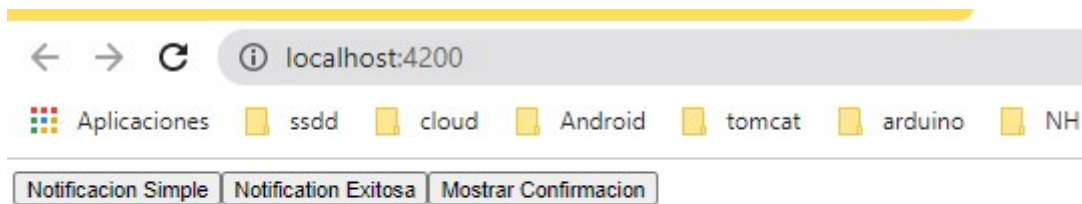
```
<button (click)="successNotification()">Notification
Exitosa</button>
```

```
<button (click)="alertConfirmation()">Mostrar  
Confirmacion</button>
```

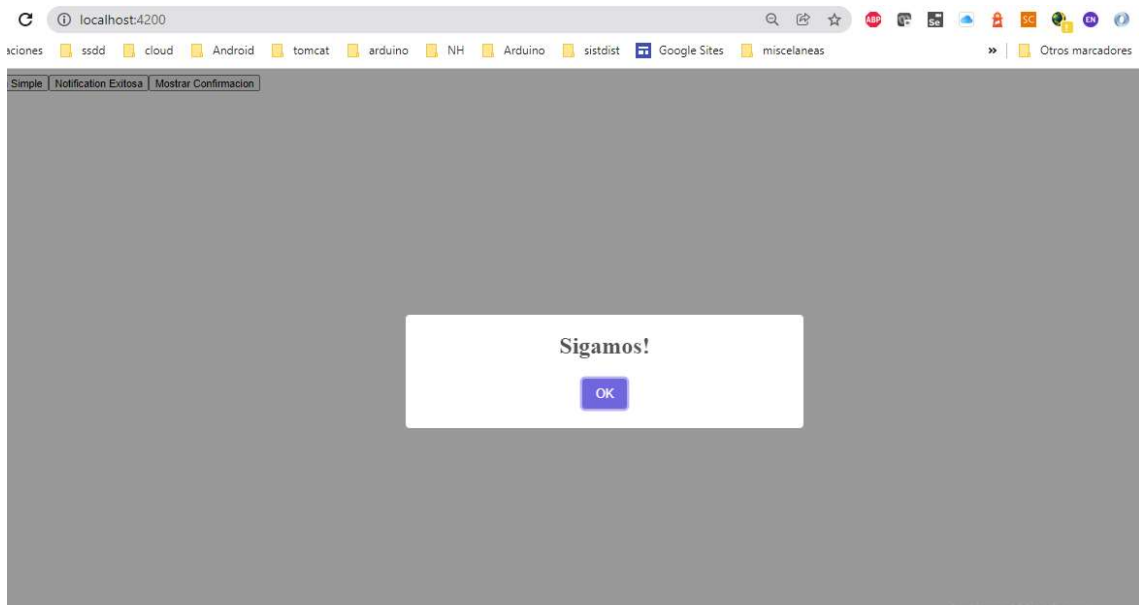
Utilice el siguiente comando para iniciar la aplicación angular:

```
ng serve --open
```

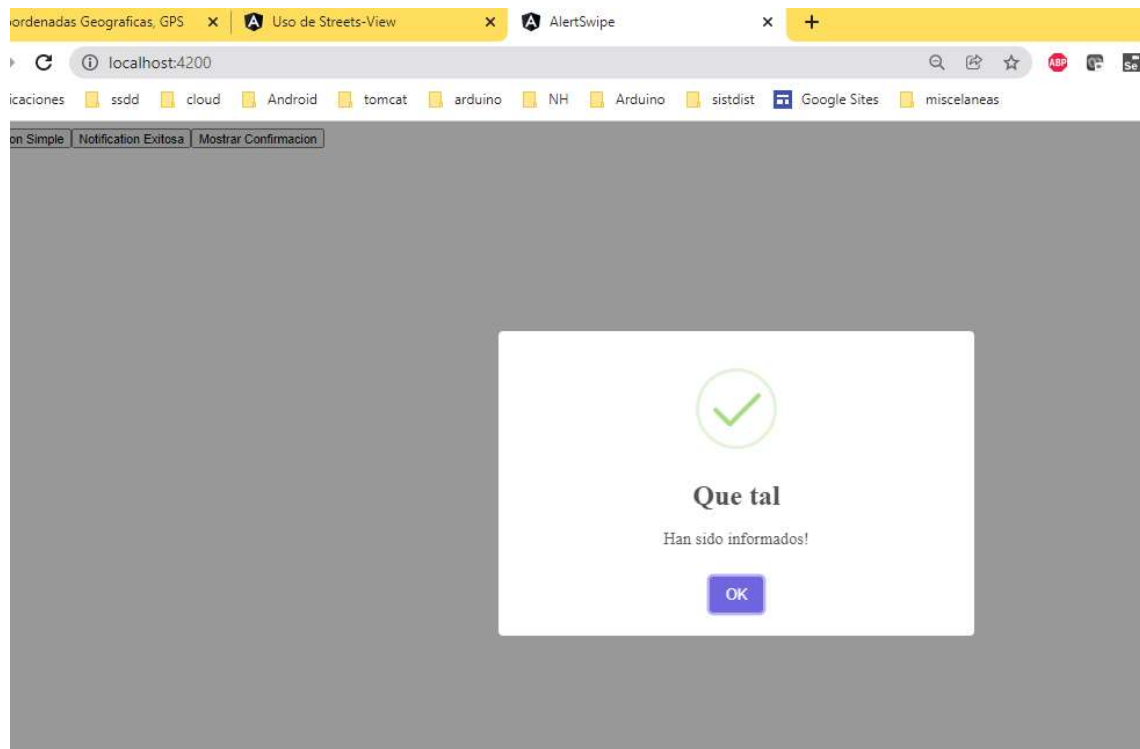
El comando anterior abre su aplicación automáticamente en el navegador, y luego la siguiente salida en la pantalla del navegador después de hacer clic en cualquiera de los botones.



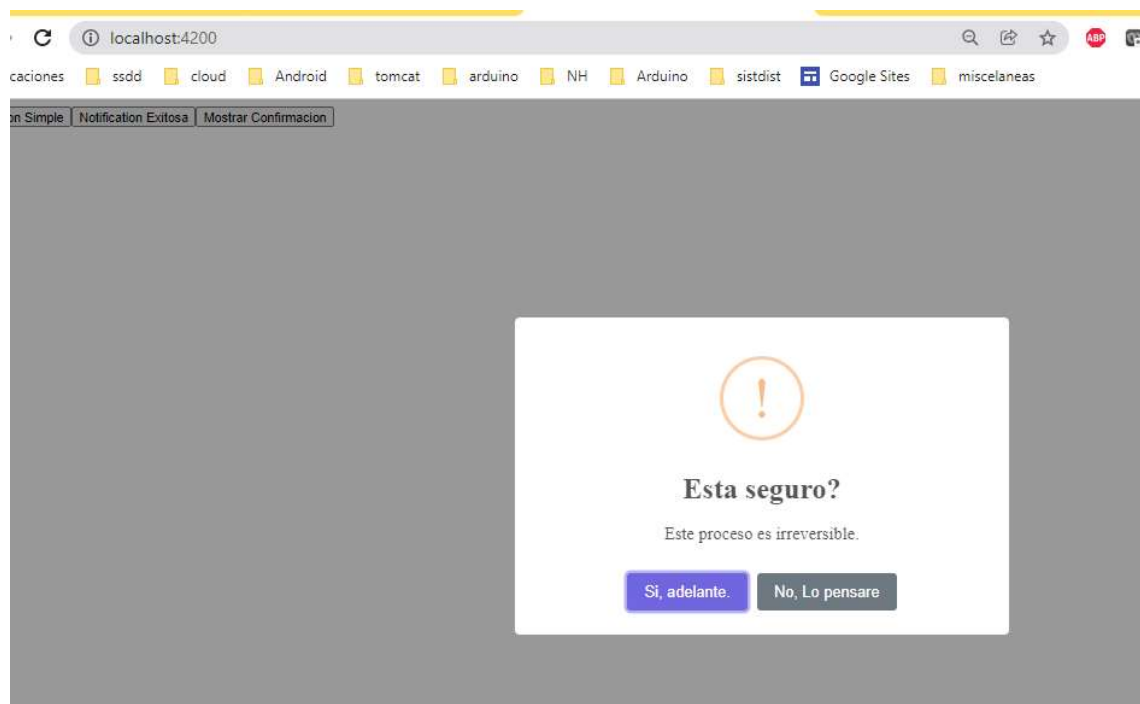
Al presionar el primer botón:



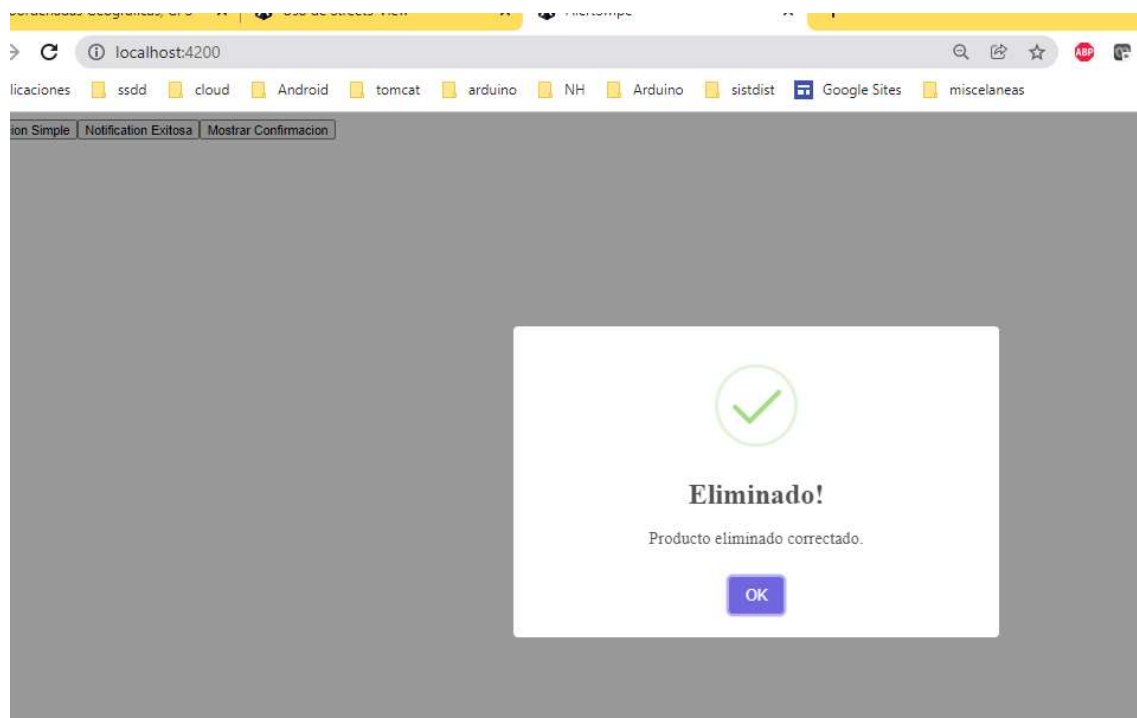
Al presionar el segundo botón..



Al presionar el tercer botón:



Si se presiona si:



Si se presiona no:

