

Práctica 4.1: Servicio Mock TipoSalas

Rama git: p_mock

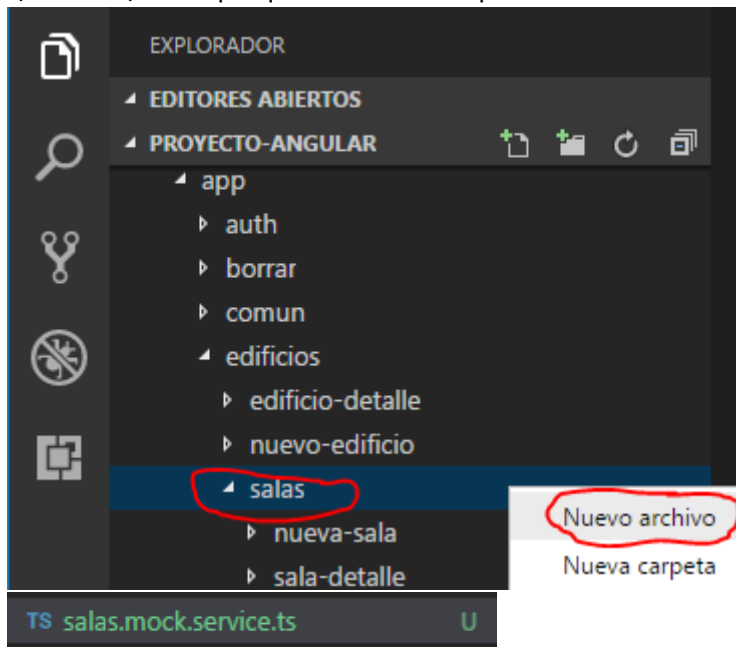
Rama resuelta: p_mock_resuelta

Usar un servicio mock ya creado: **salas.mock.service.ts** que proporciona los tipos de sala (id y descripción)

Injectar el servicio en **nueva-sala.component.ts** para que lo utilice el desplegable de tipo sala

Solución:

1. Ya está hecho: Hemos creado un nuevo servicio, nos interesa crearlo en la carpeta “/edificios/salas” porque es un servicio para salas:



2. Ya está hecho: Creamos lo básico del servicio:

```
5 export class SalasMockService {  
6  
7 }
```

3. Ya está hecho: Hemos importamos el modelo "TipoSala" de api-rest:

```
import { TipoSala } from '../api-rest/';
```

4. Ya está hecho: Hemos definido una propiedad privada "tipos" con la interfaz TipoSala y le hemos asignado el array que queremos devolver en formato json:

```
private tipos: TipoSala[] = [
  {"id":1, "tipo":'Auditorio'},
  {"id":2,"tipo":'Sala de reuniones'},
  {"id":3,"tipo":'Despacho'}
];
```

5. En **app.module.ts** importamos el servicio y lo definimos como "provider":

```
import { SalasMockService } from './edificios/salas/salas.mock.service';
[...]
```

```
providers: [
  SalasMockService,
```

```
72 providers: [
73   SalasMockService,
```

En el **componente** que va a consumir el servicio: **nueva-sala.componente.ts**

6. Importamos el **servicio** y la **interfaz** TipoSala de api-rest models

```
import {SalasMockService} from
'../../../../../edificios/salas/salas.mock.service';
import { TipoSala } from '../../../../../api-rest/';
```

```
10 import { SalasMockService } from '../../../../../edificios/salas/salas.mock.service ;
```

7. Lo instanciamos en el constructor y hacemos una llamada al servicio guardándolo en una propiedad de TipoSala:

```
constructor(private salasMockService: SalasMockService, [...] {
  this.tiposSalas = this.salasMockService.getTiposSalas();
```

```
24 constructor(private salasMockService: SalasMockService, private
25   this.tiposSalas = this.salasMockService.getTiposSalas();
26 }
```

En el componente **nueva-sala.component.html**

8. Ya está hecho: El **<mat-select>** se carga con los valores de **tiposSalas**

nueva-sala.component.html x

```
28 <div formGroupName="tipoSala" fxFlex="15%">
29   <mat-form-field>
30     <mat-select formControlName="id" placeholder="Tipo de Sala" required>
31       <mat-option *ngFor="let tipo of tiposSalas" [value]="tipo.id">
32         {{ tipo.tipo }}
33       </mat-option>
34     </mat-select>
35   </mat-form-field>
36 </div>
```