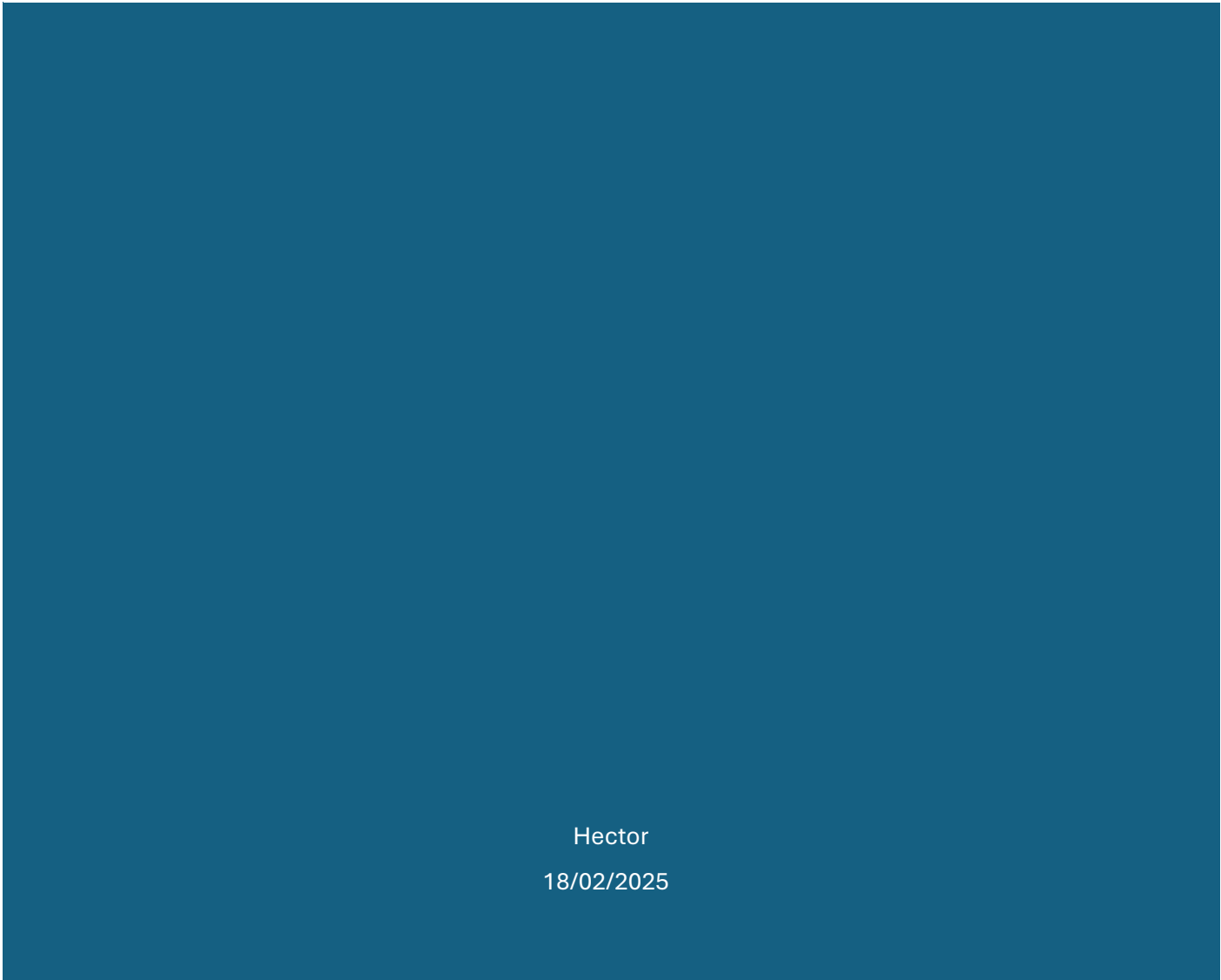




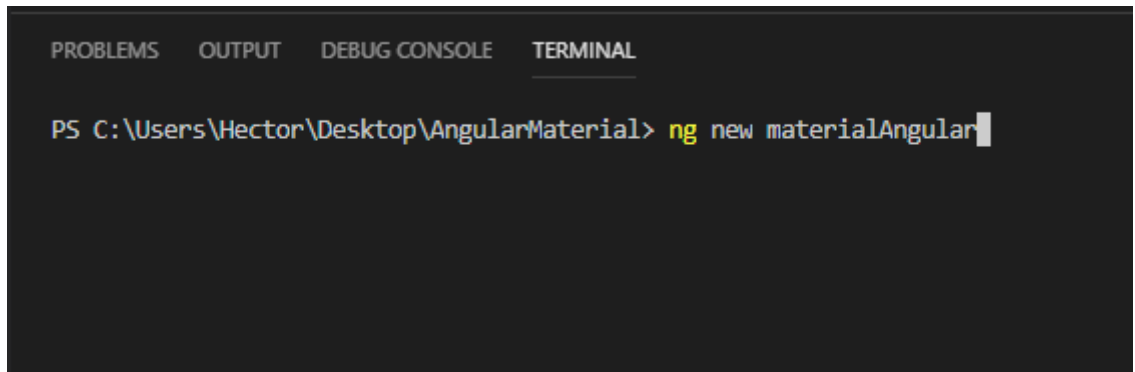
MATERIAL



Hector
18/02/2025

Para la instalacion de material en un proyecto de angular se hace con el siguiente comando:

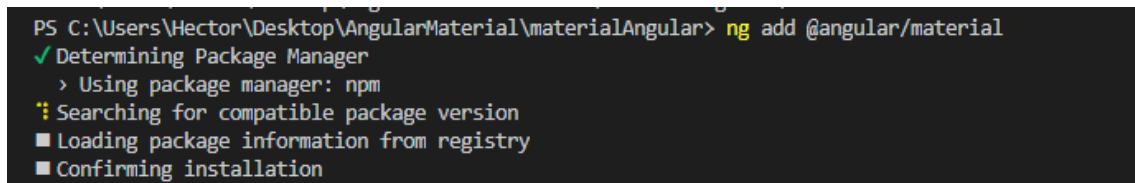
- a. Primero creamos un proyecto de angular:



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

PS C:\Users\Hector\Desktop\AngularMaterial> ng new materialAngular
```

- b. Implementamos material en el proyecto angular:



```
PS C:\Users\Hector\Desktop\AngularMaterial\materialAngular> ng add @angular/material
✓ Determining Package Manager
  > Using package manager: npm
  ⚠ Searching for compatible package version
  ■ Loading package information from registry
  ■ Confirming installation
```

Tras haber implementado Material en el proyecto de angular creamos el componente de navegacion, material contiene unos comandos para crear los componentes directamente con el programado:

- a. Creamos un componente navigation:



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  1 mode

PS C:\Users\Hector\Desktop\AngularMaterial\materialAngular> ng generate @angular/material:navигation navigation
```

b. Aquí tenemos el código HTML del componente navigation:

```
navigation.component.html ×
materialAngular > src > app > navigation > navigation.component.html > mat-sidenav-container.sidenav-container > mat-sidenav.sidenav
1 <mat-sidenav-container class="sidenav-container">
2   <mat-sidenav #drawer class="sidenav" fixedInViewport
3     [attr.role]="(isHandset$ | async) ? 'dialog' : 'navigation'"
4     [mode]="(isHandset$ | async) ? 'over' : 'side'"
5     [opened]="(isHandset$ | async) === false">
6     <mat-toolbar>Menu</mat-toolbar>
7     <mat-nav-list>
8       <a mat-list-item href="#">Link 1</a>
9       <a mat-list-item href="#">Link 2</a>
10      <a mat-list-item href="#">Link 3</a>
11    </mat-nav-list>
12  </mat-sidenav>
13  <mat-sidenav-content>
14    <mat-toolbar color="primary">
15      @if (isHandset$ | async) {
16        <button
17          type="button"
18          aria-label="Toggle sidenav"
19          mat-icon-button
20          (click)="drawer.toggle()">
21          <mat-icon aria-label="Side nav toggle icon">menu</mat-icon>
22        </button>
23      }
24      <span>materialAngular</span>
25    </mat-toolbar>
26    <!-- Add Content Here -->
27  </mat-sidenav-content>
28 </mat-sidenav-container>
29
```

c. Aquí el código de CSS:

```
# navigation.component.css ×
materialAngular > src > app > navigation > # navigation.component.css >
1 .sidenav-container {
2   height: 100%;
3 }
4
5 .sidenav {
6   width: 200px;
7 }
8
9 .sidenav .mat-toolbar {
10  background: inherit;
11 }
12
13 .mat-toolbar.mat-primary {
14   position: sticky;
15   top: 0;
16   z-index: 1;
17 }
18
```

d. Aquí el código de TypeScript:

```
TS navigation.component.ts X
materialAngular > src > app > navigation > TS navigation.component.ts > ...
1 import { Component, inject } from '@angular/core';
2 import { BreakpointObserver, Breakpoints } from '@angular/cdk/layout';
3 import { AsyncPipe } from '@angular/common';
4 import { MatToolbarModule } from '@angular/material/toolbar';
5 import { MatButtonModule } from '@angular/material/button';
6 import { MatSidenavModule } from '@angular/material/sidenav';
7 import { MatListModule } from '@angular/material/list';
8 import { MatIconModule } from '@angular/material/icon';
9 import { Observable } from 'rxjs';
10 import { map, shareReplay } from 'rxjs/operators';
11
12 @Component({
13   selector: 'app-navigation',
14   templateUrl: './navigation.component.html',
15   styleUrls: ['./navigation.component.css'],
16   standalone: true,
17   imports: [
18     MatToolbarModule,
19     MatButtonModule,
20     MatSidenavModule,
21     MatListModule,
22     MatIconModule,
23     AsyncPipe,
24   ]
25 })
26 export class NavigationComponent {
27   private breakpointObserver = inject(BreakpointObserver);
28
29   isHandset$: Observable<boolean> = this.breakpointObserver.observe(Breakpoints.Handset)
30     .pipe(
31       map(result => result.matches),
32       shareReplay()
33     );
34 }
35
```

Una vez creado el componente vamos a crear un componente Formulario:

a. Aquí esta el comando para crear el componente:

```
PS C:\Users\Hector\Desktop\AngularMaterial\materialAngular> ng generate @angular/material:address-form formulario
CREATE src/app/formulario/formulario.component.html (4018 bytes)
CREATE src/app/formulario/formulario.component.spec.ts (762 bytes)
CREATE src/app/formulario/formulario.component.ts (4136 bytes)
CREATE src/app/formulario/formulario.component.css (310 bytes)
PS C:\Users\Hector\Desktop\AngularMaterial\materialAngular>
```

b. Aquí el código html:

```
formulario.component.html X
materialAngular > src > app > formulario > formulario.component.html > form
1 <form [formGroup]="addressForm" novalidate (ngsubmit)="onSubmit()">
2   <mat-card class="shipping-card">
3     <mat-card-header>
4       <mat-card-title>Shipping Information</mat-card-title>
5     </mat-card-header>
6     <mat-card-content>
7       <div class="row">
8         <div class="col">
9           <mat-form-field class="full-width">
10             <input matInput placeholder="Company" formControlName="company">
11             </mat-form-field>
12           </div>
13         </div>
14         <div class="row">
15           <div class="col">
16             <mat-form-field class="full-width">
17               <input matInput placeholder="First name" formControlName="firstName">
18               @if (addressForm.controls['firstName'].hasError('required')) {
19                 <mat-error>First name is <strong>required</strong></mat-error>
20               }
21             </mat-form-field>
22           </div>
23           <div class="col">
24             <mat-form-field class="full-width">
25               <input matInput placeholder="Last name" formControlName="lastName">
26               @if (addressForm.controls['lastName'].hasError('required')) {
27                 <mat-error>Last name is <strong>required</strong></mat-error>
28               }
29             </mat-form-field>
30           </div>
31         </div>
32         <div class="row">
33           <div class="col">
34             <mat-form-field class="full-width">
35               <textarea matInput placeholder="Address" formControlName="address"></textarea>
36               @if (addressForm.controls['address'].hasError('required')) {
37                 <mat-error>Address is <strong>required</strong></mat-error>
38               }
39             </mat-form-field>
40           </div>
41         </div>
42         <div class="row">
43           <div class="col">
44             @if (hasInitNumber) {
45               <mat-form-field class="full-width">
46                 <textarea matInput placeholder="Address 2" formControlName="address2"></textarea>
47               </mat-form-field>
48             } @else {
49               <button mat-button type="button" (click)="hasInitNumber = !hasInitNumber">
50                 + Add C/O, Apt, Suite, Unit
51               </button>
52             }
53           </div>
54         </div>
55       </div>
56     </mat-card-content>
57   </mat-card>
58 </form>
```

c. Aquí el código CSS:

```
# formulario.component.css ×
materialAngular > src > app > formulario > # formulario.component
1  .full-width {
2    width: 100%;
3  }
4
5  .shipping-card {
6    min-width: 120px;
7    margin: 20px auto;
8  }
9
10 .mat-radio-button {
11   display: block;
12   margin: 5px 0;
13 }
14
15 .row {
16   display: flex;
17   flex-direction: row;
18 }
19
20 .col {
21   flex: 1;
22   margin-right: 20px;
23 }
24
25 .col:last-child {
26   margin-right: 0;
27 }
28
```

d. Aquí el TypeScript:

```
formulario.component.ts ×
materialAngular > src > app > formulario > TS formulario.component.ts > ...
1  import { Component, inject } from '@angular/core';
2
3  import { ReactiveFormsModule, FormBuilder, Validators } from '@angular/forms';
4  import { MatInputModule } from '@angular/material/input';
5  import { MatButtonModule } from '@angular/material/button';
6  import { MatSelectModule } from '@angular/material/select';
7  import { MatRadioModule } from '@angular/material/radio';
8  import { MatCardModule } from '@angular/material/card';
9
10
11 @Component({
12   selector: 'app-formulario',
13   templateUrl: './formulario.component.html',
14   styleUrls: ['./formulario.component.css'],
15   standalone: true,
16   imports: [
17     MatInputModule,
18     MatButtonModule,
19     MatSelectModule,
20     MatRadioModule,
21     MatCardModule,
22     ReactiveFormsModule
23   ]
24 })
25 export class FormularioComponent {
26   private fb = inject(FormBuilder);
27   addressForm = this.fb.group({
28     company: null,
29     firstName: [null, Validators.required],
30     lastName: [null, Validators.required],
31     address: [null, Validators.required],
32     address2: null,
33     city: [null, Validators.required],
34     state: [null, Validators.required],
35     postalCode: [null, Validators.compose([
36       Validators.required, Validators.minLength(5), Validators.maxLength(5)])],
37   ],
38   shipping: ['free', Validators.required]
39 });
40
41 hasUnitNumber = false;
42
```

Ahora creamos la ruta para colocar el componente formulario en el navigation:

- a. Nos vamos al archivo app.routes.ts:

```
TS app.routes.ts X
materialAngular > src > app > TS app.routes.ts > [?] routes
1  import { Routes } from '@angular/router';
2  import { FormularioComponent } from '../formulario/formulario.component';
3
4  export const routes: Routes = [
5    {
6      path: 'form', component: FormularioComponent
7    },
8  ];
9
```

- b. Una vez creada la ruta nos vamos al componente del navigation y con un routerLink enlazamos al componente:

```
<> navigation.component.html X TS navigation.component.ts
materialAngular > src > app > navigation > <> navigation.component.html > mat-sidenav-container.sidenav-conta
1  <mat-sidenav-container class="sidenav-container">
2    <mat-sidenav #drawer class="sidenav" fixedInViewport
3      [attr.role]="(isHandset$ | async) ? 'dialog' : 'navigation'"
4      [mode]="(isHandset$ | async) ? 'over' : 'side'"
5      [opened]="(isHandset$ | async) === false">
6      <mat-toolbar>Menu</mat-toolbar>
7      <mat-nav-list>
8        <a mat-list-item routerLink="/form">Formulario</a>
9        <a mat-list-item href="#">Link 2</a>
10       <a mat-list-item href="#">Link 3</a>
11     </mat-nav-list>
12   </mat-sidenav>
13   <mat-sidenav-content>
14     <mat-toolbar color="primary">
15       @if (isHandset$ | async) {
16         <button
17           type="button"
18           aria-label="Toggle sidenav"
19           mat-icon-button
20           (click)="drawer.toggle()">
21           <mat-icon aria-label="Side nav toggle icon">menu</mat-icon>
22         </button>
23       }
24       <span>materialAngular</span>
25     </mat-toolbar>
26     <!-- Add Content Here -->
27   </mat-sidenav-content>
28 </mat-sidenav-container>
29
```

c. Importamos el routerLink en el navigation.component.ts:

```
<> navigation.component.html TS app.routes.ts TS navigation.component.ts X
materialAngular > src > app > navigation > TS navigation.component.ts > NavigationComponent
1 import { Component, inject } from '@angular/core';
2 import { BreakpointObserver, Breakpoints } from '@angular/cdk/layout';
3 import { AsyncPipe } from '@angular/common';
4 import { MatToolbarModule } from '@angular/material/toolbar';
5 import { MatButtonModule } from '@angular/material/button';
6 import { MatSidenavModule } from '@angular/material/sidenav';
7 import { MatListModule } from '@angular/material/list';
8 import { MatIconModule } from '@angular/material/icon';
9 import { Observable } from 'rxjs';
10 import { map, shareReplay } from 'rxjs/operators';
11 import { Router, RouterLink, RouterOutlet } from '@angular/router';
12
13 @Component({
14   selector: 'app-navigation',
15   templateUrl: './navigation.component.html',
16   styleUrls: ['./navigation.component.css'],
17   standalone: true,
18   imports: [
19     MatToolbarModule,
20     MatButtonModule,
21     MatSidenavModule,
22     MatListModule,
23     MatIconModule,
24     AsyncPipe,
25     RouterLink,
26     RouterOutlet
27   ]
28 })
29 export class NavigationComponent {
30   private breakpointObserver = inject(BreakpointObserver);
31
32   isHandset$: Observable<boolean> = this.breakpointObserver.observe(Breakpoints.Handset)
33     .pipe(
34       map((res) => res.matches)
35     );
36 }
```

d. Nos quedaria algo asi:

The screenshot shows a web application interface with a sidebar menu on the left containing 'Menu', 'Formulario', 'Tabla', 'Dashboard', and 'Tabs'. The main content area is titled 'materialAngular' and displays a 'Shipping Information' form. The form includes a 'Company' field, followed by 'First name' and 'Last name' fields. Below these is an 'Address' field with a clear button. A link '+ Add C/O, Apt, Suite, Unit' is present. The 'City' and 'State' fields are next to each other, followed by a 'Postal Code' field. At the bottom, there are three radio buttons: 'Free Shipping' (selected), 'Priority Shipping', and 'Next Day Shipping'. A 'Submit' button is located at the very bottom of the form. The page number '0 / 5' is visible in the bottom right corner.

Una vez creado el formulario y enlazarlo creamos el componente tabla:

- a. Lo creamos con el siguiente comando:

```
\AngularMaterial\materialAngular> ng generate @angular/material:table tabla
CREATE src/app/tabla/tabla-datasource.ts (3710 bytes)
CREATE src/app/tabla/tabla.component.html (908 bytes)
CREATE src/app/tabla/tabla.component.spec.ts (732 bytes)
CREATE src/app/tabla/tabla.component.ts (1108 bytes)
CREATE src/app/tabla/tabla.component.css (40 bytes)
PS C:\Users\Hector\Desktop\AngularMaterial\materialAngular>
```

- b. Este es el código HTML del componente tabla que nos ha creado material:

```
<> tabla.component.html X
materialAngular > src > app > tabla > <> tabla.component.html > div.mat-elevation-z2
Go to component
1 <div class="mat-elevation-z2">
2   <table mat-table class="full-width-table" matSort aria-label="Elements">
3     <!-- Id Column -->
4     <ng-container matColumnDef="id">
5       <th mat-header-cell *matHeaderCellDef mat-sort-header>Id</th>
6       <td mat-cell *matCellDef="let row">{{row.id}}</td>
7     </ng-container>
8
9     <!-- Name Column -->
10    <ng-container matColumnDef="name">
11      <th mat-header-cell *matHeaderCellDef mat-sort-header>Name</th>
12      <td mat-cell *matCellDef="let row">{{row.name}}</td>
13    </ng-container>
14
15    <tr mat-header-row *matHeaderRowDef="displayedColumns"></tr>
16    <tr mat-row *matRowDef="let row; columns: displayedColumns;"></tr>
17  </table>
18
19  <mat-paginator #paginator
20    [length]="dataSource.data.length"
21    [pageIndex]="0"
22    [pageSize]="10"
23    [pageSizeOptions]="[5, 10, 20]"
24    aria-label="Select page">
25  </mat-paginator>
26 </div>
27
```


c. Este es el código del CSS:

```
# tabla.component.css X
materialAngular > src > app > tabla > # tabla.componente
1  .full-width-table {
2      width: 100%;
3  }
4
```

Una vez creado ese componente modificamos el app.routes.ts para colocar un path y un componente:

a. Así nos debe de quedar el archivo app.routes.ts:

```
TS app.routes.ts X
materialAngular > src > app > TS app.routes.ts > routes > path
1  import { Routes } from '@angular/router';
2  import { FormularioComponent } from '../formulario/formulario.component';
3  import { TablaComponent } from '../tabla/tabla.component';
4
5  export const routes: Routes = [
6      { path: 'formulario', component: FormularioComponent },
7      { path: 'table', component: TablaComponent },
8  ];
9
```

b. Una vez creada la ruta en el html del navigation creamos un routerLink para la tabla:

```
navigation.component.html
materialAngular > src > app > navigation > navigation.component.html > mat-sidenav-container.sidenav-container > mat-sidenav
Go to component
1 <mat-sidenav-container class="sidenav-container">
2   <mat-sidenav #drawer class="sidenav" fixedInViewport
3     [attr.role]="(isHandset$ | async) ? 'dialog' : 'navigation'"
4     [mode]="(isHandset$ | async) ? 'over' : 'side'"
5     [opened]="(isHandset$ | async) === false">
6     <mat-toolbar>Menu</mat-toolbar>
7     <mat-nav-list>
8       <a mat-list-item routerLink="/formulario">Formulario</a>
9       <a mat-list-item routerLink="/table">Tabla</a>
10      <a mat-list-item href="#">Dashboard</a>
11      <a mat-list-item href="#">Tabs</a>
12    </mat-nav-list>
13  </mat-sidenav>
14  <mat-sidenav-content>
15    <mat-toolbar color="primary">
16      <button
17        *ngIf="isHandset$ | async"
18        type="button"
19        aria-label="Toggle sidenav"
20        mat-icon-button
21        (click)="drawer.toggle()">
22        <mat-icon aria-label="Side nav toggle icon">menu</mat-icon>
23      </button>
24      <span>materialAngular</span>
25    </mat-toolbar>
26    <!-- Add Content Here -->
27    <router-outlet></router-outlet>
28  </mat-sidenav-content>
29 </mat-sidenav-container>
30
```

c. Quedaria algo asi:

Menu	materialAngular		Name
Formulario	id		
Tabla	1		Hydrogen
Dashboard	2		Helium
Tabs	3		Lithium
	4		Beryllium
	5		Boron
	6		Carbon
	7		Nitrogen
	8		Oxygen
	9		Fluorine
	10		Neon

Items per page: 10 1 - 10 of 20

Ahora crearemos el componente dashboard:

- a. Con el siguiente comando creamos el componente:

```
PS C:\Users\Hector\Desktop\AngularMaterial\materialAngular> ng generate @angular/material:dashboard dashboard
CREATE src/app/dashboard/dashboard.component.html (1018 bytes)
CREATE src/app/dashboard/dashboard.component.spec.ts (756 bytes)
CREATE src/app/dashboard/dashboard.component.ts (1576 bytes)
CREATE src/app/dashboard/dashboard.component.css (276 bytes)
PS C:\Users\Hector\Desktop\AngularMaterial\materialAngular>
```

- b. Una vez ejecutado el componente este seria el html:

```
dashboard.component.html X
materialAngular > src > app > dashboard > dashboard.component.html > div.grid-container
1 <div class="grid-container">
2   <h1 class="mat-h1">Dashboard</h1>
3   <mat-grid-list cols="2" rowHeight="350px">
4     @for (card of cards | async; track card) {
5       <mat-grid-tile [colspan]="card.cols" [rowspan]="card.rows">
6         <mat-card class="dashboard-card">
7           <mat-card-header>
8             <mat-card-title>
9               {{card.title}}
10              <button mat-icon-button class="more-button" [matMenuTriggerFor]="menu" aria-label="Toggle menu">
11                <mat-icon>more_vert</mat-icon>
12              </button>
13              <mat-menu #menu="matMenu" xPosition="before">
14                <button mat-menu-item>Expand</button>
15                <button mat-menu-item>Remove</button>
16              </mat-menu>
17            </mat-card-title>
18          </mat-card-header>
19          <mat-card-content class="dashboard-card-content">
20            <div>Card Content Here</div>
21          </mat-card-content>
22        </mat-card>
23      </mat-grid-tile>
24    }
25  </mat-grid-list>
26 </div>
27
```

c. Este seria el codigo css:

```
# dashboard.component.css X
materialAngular > src > app > dashboard > # dashboard.component.css > .grid-c
1  .grid-container {
2    margin: 20px;
3  }
4
5  .dashboard-card {
6    position: absolute;
7    top: 15px;
8    left: 15px;
9    right: 15px;
10   bottom: 15px;
11  }
12
13  .more-button {
14    position: absolute;
15    top: 5px;
16    right: 10px;
17  }
18
19  .dashboard-card-content {
20    text-align: center;
21  }
22
```

Una vez creado el componente modificamos el archivo app.routes.ts para enlazar el dashboard con el navigation:

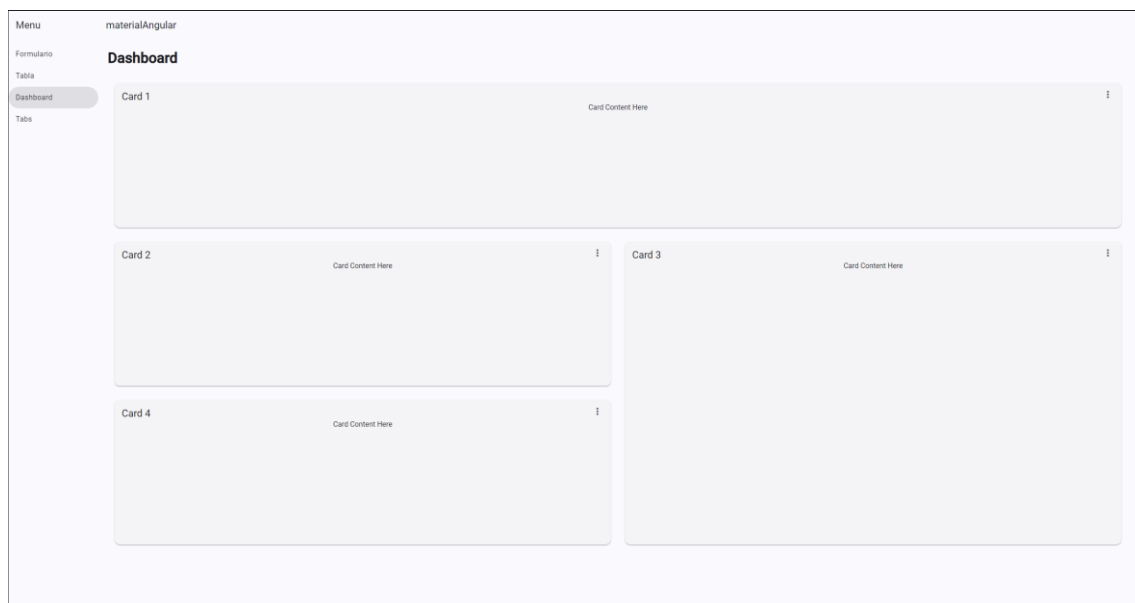
a. Este serie el app.routes.ts:

```
TS app.routes.ts X
materialAngular > src > app > TS app.routes.ts > routes
1  import { Routes } from '@angular/router';
2  import { FormularioComponent } from '../formulario/formulario.component';
3  import { TablaComponent } from '../tabla/tabla.component';
4  import { DashboardComponent } from '../dashboard/dashboard.component';
5
6  export const routes: Routes = [
7    { path: 'formulario', component: FormularioComponent },
8    { path: 'table', component: TablaComponent },
9    { path: 'dashboard', component: DashboardComponent },
10  ];
11
```

b. Modificamos el html del navigation:

```
TS app.routes.ts  <> navigation.component.html X
materialAngular > src > app > navigation > <> navigation.component.html > mat-sidenav-container.sidenav-containe
Go to component
1  <mat-sidenav-container class="sidenav-container">
2    <mat-sidenav #drawer class="sidenav" fixedInViewport
3      [attr.role]="(isHandset$ | async) ? 'dialog' : 'navigation'"
4      [mode]="(isHandset$ | async) ? 'over' : 'side'"
5      [opened]="(isHandset$ | async) === false">
6      <mat-toolbar>Menu</mat-toolbar>
7      <mat-nav-list>
8        <a mat-list-item routerLink="/formulario">Formulario</a>
9        <a mat-list-item routerLink="/table">Tabla</a>
10       <a mat-list-item routerLink="/dashboard">Dashboard</a>
11       <a mat-list-item href="#">Tabs</a>
12     </mat-nav-list>
13   </mat-sidenav>
14   <mat-sidenav-content>
15     <mat-toolbar color="primary">
16       <button
17         *ngIf="isHandset$ | async"
18         type="button"
19         aria-label="Toggle sidenav"
20         mat-icon-button
21         (click)="drawer.toggle()">
22         <mat-icon aria-label="Side nav toggle icon">menu</mat-icon>
23       </button>
24       <span>materialAngular</span>
25     </mat-toolbar>
26     <!-- Add Content Here -->
27     <router-outlet></router-outlet>
28   </mat-sidenav-content>
29 </mat-sidenav-container>
30
```

Una vez creado el dashboard comprobamos que funciones debe de quedar asi:



Por ultimo creamos el componente tabs, en este caso debemos de coger los codigos del componente de la pagina oficial de material:

- a. El comando seria el siguiente:

```
PS C:\Users\Hector\Desktop\AngularMaterial\materialAngular> ng g c tabs
CREATE src/app/tabs/tabs.component.html (20 bytes)
CREATE src/app/tabs/tabs.component.spec.ts (601 bytes)
CREATE src/app/tabs/tabs.component.ts (217 bytes)
CREATE src/app/tabs/tabs.component.css (0 bytes)
PS C:\Users\Hector\Desktop\AngularMaterial\materialAngular> |
```

- b. Este serie el html de los tabs:

```
<> tabs.component.html X TS tabs.component.ts 1
materialAngular > src > app > tabs > <> tabs.component.html > mat-tab-group
Go to component
1 <mat-tab-group>
2   <mat-tab label="First"> Content 1 </mat-tab>
3   <mat-tab label="Second"> Content 2 </mat-tab>
4   <mat-tab label="Third"> Content 3 </mat-tab>
5 </mat-tab-group>
6
```

- c. Creamos en el archivo app.routes.ts la ruta del componente tabs:

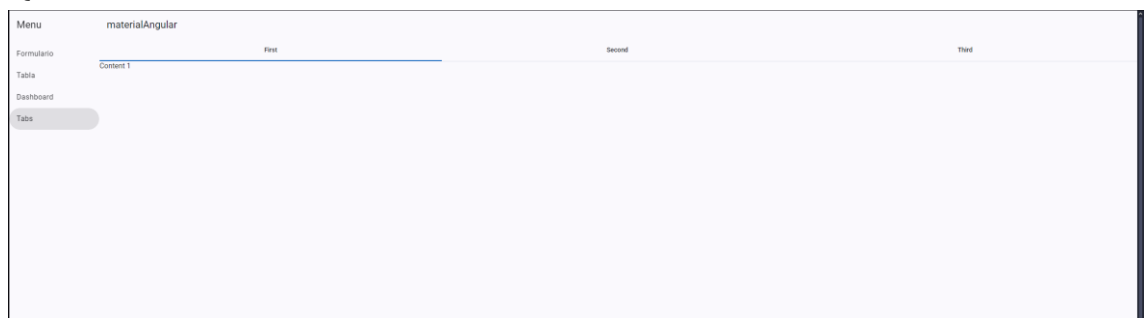
```
materialAngular > src > app > TS app.routes.ts > routes
1 import { Routes } from '@angular/router';
2 import { FormularioComponent } from './formulario/formulario.component';
3 import { TablaComponent } from './tabla/tabla.component';
4 import { DashboardComponent } from './dashboard/dashboard.component';
5 import { TabsComponent } from './tabs/tabs.component';
6
7 export const routes: Routes = [
8   { path: 'formulario', component: FormularioComponent },
9   { path: 'tabla', component: TablaComponent },
10  { path: 'dashboard', component: DashboardComponent },
11  { path: 'tabs', component: TabsComponent },
12 ];
13
```

En el componente navigation creamos un routerlink para enlazar el componente tabs:

a. Asi quedaria el html del componente navigation:

```
navigation.component.html X TS app.routes.ts
materialAngular > src > app > navigation > navigation.component.html > mat-sidenav-container.sidenav-container > mat-s
Go to component
1 <mat-sidenav-container class="sidenav-container">
2   <mat-sidenav #drawer class="sidenav" fixedInViewport
3     [attr.role]="(isHandset$ | async) ? 'dialog' : 'navigation'"
4     [mode]="(isHandset$ | async) ? 'over' : 'side'"
5     [opened]="(isHandset$ | async) === false">
6     <mat-toolbar>Menu</mat-toolbar>
7     <mat-nav-list>
8       <a mat-list-item routerLink="/formulario">Formulario</a>
9       <a mat-list-item routerLink="/table">Tabla</a>
10      <a mat-list-item routerLink="/dashboard">Dashboard</a>
11      <a mat-list-item routerLink="/tabs">Tabs</a>
12    </mat-nav-list>
13  </mat-sidenav>
14  <mat-sidenav-content>
15    <mat-toolbar color="primary">
16      <button
17        *ngIf="isHandset$ | async"
18        type="button"
19        aria-label="Toggle sidenav"
20        mat-icon-button
21        (click)="drawer.toggle()">
22        <mat-icon aria-label="Side nav toggle icon">menu</mat-icon>
23      </button>
24      <span>materialAngular</span>
25    </mat-toolbar>
26    <!-- Add Content Here -->
27    <router-outlet></router-outlet>
28  </mat-sidenav-content>
29 </mat-sidenav-container>
30
```

b. Quedaría asi:



En el tab1 añadiremos una card de Shiba Inu de material:

a. Este seria el codigo html de la card dentro del tab:

```
materialAngular > src > app > tabs > tabs.component.html > mat-tab-group > mat-tab > mat-card.example-card
Go to component
1 <mat-tab-group>
2   <mat-tab label="First">
3     <mat-card class="example-card" appearance="outlined">
4       <mat-card-header>
5         <div mat-card-avatar class="example-header-image"></div>
6         <mat-card-title>Shiba Inu</mat-card-title>
7         <mat-card-subtitle>Dog Breed</mat-card-subtitle>
8       </mat-card-header>
9       
10      <mat-card-content>
11        <p>
12          The Shiba Inu is the smallest of the six original and distinct spitz breeds of dog from Japan.
13          A small, agile dog that copes very well with mountainous terrain, the Shiba Inu was originally
14          bred for hunting.
15        </p>
16      </mat-card-content>
17      <mat-card-actions>
18        <button mat-button>LIKE</button>
19        <button mat-button>SHARE</button>
20      </mat-card-actions>
21    </mat-card>
22  </mat-tab>
23  <mat-tab label="Second"> Content 2 </mat-tab>
24  <mat-tab label="Third"> Content 3 </mat-tab>
25</mat-tab-group>
```

b. Este seria el codigo css:

```
materialAngular > src > app > tabs > # tabs.component.css > ...
1 .example-card {
2   max-width: 400px;
3 }
4
5 .example-header-image {
6   background-image: url('https://material.angular.io/assets/img/examples/shiba1.jpg');
7   background-size: cover;
8 }
9
```

c. Este seria el tab con la card:

