

# Actividad guía Angular Material/ionic Lista de tareas

## Actividad Práctica: Aplicación de Lista de Tareas con Angular Material e Ionic

### Objetivo:

Los estudiantes crearán una aplicación sencilla de **lista de tareas** utilizando **ionic** y **Angular Material**, donde podrán agregar, marcar como completada y eliminar tareas. La actividad está diseñada para aplicar los conceptos básicos aprendidos sobre **botones**, **tarjetas** y **interactividad** con **ionic** y **Angular Material**.

---

### Instrucciones:

#### 1. Instalar Ionic CLI (Si no está instalado)

Si no tienes Ionic instalado, instala la CLI globalmente con el siguiente comando:

```
npm install -g @ionic/cli
```

#### 2. Crear un nuevo proyecto Ionic

Crea una nueva aplicación con la plantilla en blanco:

```
ionic start listaDeTareas blank
```

#### 3. Navegar al directorio del proyecto

Cambia al directorio del proyecto recién creado:

```
cd listaDeTareas
```

#### 4. Instalar Angular Material

Instala **Angular Material** junto con **CDK** y **Animations** para agregar estilos a los componentes de la aplicación:

```
ng add @angular/material
```

- Selecciona un tema (por ejemplo, **indigo-pink**).
- Habilita las animaciones cuando se te pida.

## 5. Agregar Módulos de Angular Material

Asegúrate de que los módulos necesarios estén instalados:

```
npm install @angular/material @angular/cdk @angular/animations
```

## 6. Configurar el archivo `global.scss`

Abre el archivo `src/global.scss` y añade la importación del tema seleccionado de Angular Material:

```
@import '~@angular/material/prebuilt-themes/indigo-pink.css';
```

## 7. Modificar el archivo `app.module.ts`

Agrega los módulos necesarios en `src/app/app.module.ts`:

```
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';
import { IonicModule, IonicRouteStrategy } from '@ionic/angular';
import { RouteReuseStrategy } from '@angular/router';
import { AppComponent } from './app.component';
import { AppRoutingModule } from './app-routing.module';

// Importar Angular Material
import { MatButtonModule } from '@angular/material/button';
```

```

import { MatCardModule } from '@angular/material/card';
import { MatInputModule } from '@angular/material/input';
import { MatIconModule } from '@angular/material/icon';
import { BrowserAnimationsModule } from '@angular/platform-
browser/animations';

@NgModule({
  declarations: [AppComponent],
  imports: [
    BrowserModule,
    IonicModule.forRoot(),
    AppRoutingModule,
    MatButtonModule, // Para los botones
    MatCardModule,   // Para las tarjetas
    MatInputModule,  // Para los inputs
    MatIconModule,   // Para los íconos
    BrowserAnimationsModule // Para las animaciones
  ],
  providers: [{ provide: RouteReuseStrategy, useClass: Ion
icRouteStrategy }],
  bootstrap: [AppComponent],
})
export class AppModule {}

```

## 8. Crear la interfaz en `app.component.html`

Añade los siguientes elementos en el archivo `src/app/app.component.html` para crear la interfaz de la lista de tareas:

```

<ion-app>
  <ion-header>
    <ion-toolbar>
      <ion-title>Lista de Tareas</ion-title>
    </ion-toolbar>
  </ion-header>

```

```

<ion-content class="ion-padding">
  <!-- Input para agregar tareas -->
  <mat-form-field class="example-full-width">
    <mat-label>Nueva tarea</mat-label>
    <input matInput [(ngModel)]="nuevaTarea" placeholder
="Escribe una tarea" />
    <button mat-icon-button matSuffix color="primary" (c
lick)="agregarTarea()">
      <mat-icon>add</mat-icon>
    </button>
  </mat-form-field>

  <!-- Mostrar la lista de tareas -->
  <div *ngFor="let tarea of listaDeTareas; let i = inde
x">
    <mat-card>
      <mat-card-content>
        <p [ngStyle]="{'text-decoration': tarea.completa
da ? 'line-through' : 'none'}">{{ tarea.nombre }}</p>
      </mat-card-content>
      <mat-card-actions>
        <button mat-button color="primary" (click)="comp
letarTarea(i)">
          {{ tarea.completada ? 'Desmarcar' : 'Completa
r' }}
        </button>
        <button mat-button color="warn" (click)="elimina
rTarea(i)">Eliminar</button>
      </mat-card-actions>
    </mat-card>
  </div>
</ion-content>
</ion-app>

```

## 9. Agregar la lógica en `app.component.ts`

Implementa la lógica para agregar, completar y eliminar tareas en el archivo

`src/app/app.component.ts`:

```
import { Component } from '@angular/core';

interface Tarea {
  nombre: string;
  completada: boolean;
}

@Component({
  selector: 'app-root',
  templateUrl: 'app.component.html',
  styleUrls: ['app.component.scss'],
})
export class AppComponent {
  nuevaTarea: string = '';
  listaDeTareas: Tarea[] = [];

  // Método para agregar una tarea
  agregarTarea() {
    if (this.nuevaTarea.trim() !== '') {
      this.listaDeTareas.push({ nombre: this.nuevaTarea, c
        ompletada: false });
      this.nuevaTarea = ''; // Limpiar el input después de
        agregar la tarea
    }
  }

  // Método para marcar o desmarcar una tarea como complet
    ada
  completarTarea(indice: number) {
    this.listaDeTareas[indice].completada = !this.listaDeT
      areas[indice].completada;
  }
}
```

```
// Método para eliminar una tarea
eliminarTarea(indice: number) {
  this.listaDeTareas.splice(indice, 1);
}
}
```

## 10. Ejecutar la aplicación

Inicia el servidor de desarrollo y prueba la aplicación:

```
ionic serve
```

La aplicación se abrirá en tu navegador, y deberías poder agregar, completar y eliminar tareas de la lista.

---

## Resumen de la Actividad

**Objetivo:** Crear una aplicación sencilla de lista de tareas que permita a los estudiantes aplicar los siguientes conceptos:

- **Uso de botones y tarjetas de Angular Material.**
- **Interactividad básica con Ionic y Angular.**
- **Vinculación de datos** (binding) para mostrar y manipular la lista de tareas.

### Resultados esperados:

- Los estudiantes deben poder agregar tareas, marcarlas como completadas o eliminarlas.
- Entenderán cómo usar **inputs, botones y tarjetas** en **Angular Material**.
- Enviar el link del repositorio de GitHub con la carpeta src de esta actividad.