

# Guía clase 1

## Introducción

Esta guía te llevará paso a paso por el proceso de instalación, configuración y creación de una nueva página en un proyecto Ionic 7, además de cómo visualizar la página en el navegador. También incluye soluciones a posibles errores que pueden surgir durante el proceso.

## Paso 1: Preparar el Entorno

### 1. Instalar Node.js y npm:

- Descarga e instala Node.js desde [Node.js Download](#).
- Verifica la instalación con:

```
#Verifica la versión de Node
node -v
#verifica la versión de npm
npm -v
```

### 2. Instalar Ionic CLI:

- Abre una terminal y ejecuta el siguiente comando para instalar Ionic CLI globalmente:

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

- Verifica la instalación con:

```
ionic
```

## Paso 2: Crear un Nuevo Proyecto Ionic

### 1. Crear un Proyecto:

- Navega a la carpeta donde deseas crear tu proyecto en la terminal y ejecuta:

```
# Nombre: Blank (En blanco)
# Crea un nuevo proyecto de Ionic con una plantilla en
# blanco, sin componentes predefinidos. Ideal para empeza
# r desde cero.
ionic start elNombreDeTuProyecto blank

# Nombre: Tabs (Pestañas)
# Crea un nuevo proyecto de Ionic con una navegación po
# r pestañas en la parte inferior. Útil para aplicaciones
# con múltiples secciones.
ionic start elNombreDeTuProyecto tabs

# Nombre: Side Menu (Menú lateral)
# Crea un nuevo proyecto de Ionic con un menú lateral d
# eslizable, que se puede usar para la navegación princip
# al de la aplicación.
ionic start elNombreDeTuProyecto sidemenu

# Nombre: Side Menu + Tabs (Menú lateral con pestañas)
# Crea un nuevo proyecto de Ionic que combina un menú l
# ateral con una navegación por pestañas, ideal para apli
# caciones con estructura compleja.
ionic start elNombreDeTuProyecto sidemenu --tabs

# Nombre: Conference (Conferencia)
# Crea un nuevo proyecto de Ionic basado en una plantil
# la avanzada para conferencias, que incluye calendario,
```

```
lista de sesiones, y más.  
ionic start elNombreDeTuProyecto conference  
  
# Nombre: Tutorial (Tutorial de introducción)  
# Crea un nuevo proyecto de Ionic con una serie de pantallas de tutorial que guían al usuario a través de la aplicación.  
ionic start elNombreDeTuProyecto tutorial  
  
# Nombre: List (Lista)  
# Crea un nuevo proyecto de Ionic que presenta una lista de elementos, ideal para aplicaciones de tareas, contactos, o listas de compras.  
ionic start elNombreDeTuProyecto list  
  
# Nombre: My First App (Mi primera app)  
# Crea un nuevo proyecto de Ionic diseñado para principiantes, con componentes básicos como botones, tarjetas, listas y formularios.  
ionic start elNombreDeTuProyecto tabs
```

- Selecciona Angular como framework cuando se te pregunte.
- Elige `NgModules` cuando se te pregunte si deseas usar `NgModules` o `Standalone Components`.

## 2. Navegar al Directorio del Proyecto:

- Cambia al directorio del proyecto:

```
cd elNombreDeTuProyecto
```

## Paso 3: Instalar Dependencias Adicionales

### 1. Instalar `@angular/cli`:

- Si no lo has hecho ya, instala `@angular/cli` globalmente:

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

## 2. Instalar Dependencias del Proyecto:

- Asegúrate de que todas las dependencias necesarias estén instaladas:

```
npm install
```

## Paso 4: Generar una Nueva Página

### 1. Generar la Página:

- En el directorio del proyecto, ejecuta el comando para generar una nueva página llamada `Acceso` :

```
ionic generate page Acceso
```

- Responde `y` o `n` a la pregunta sobre compartir datos con Google según tu preferencia.

### 2. Verificar la Generación de Archivos:

- Los siguientes archivos deberían ser creados:

- `src/app/acceso/acceso-routing.module.ts`
- `src/app/acceso/acceso.module.ts`
- `src/app/acceso/acceso.page.html`
- `src/app/acceso/acceso.page.scss`
- `src/app/acceso/acceso.page.ts`
- `src/app/acceso/acceso.page.spec.ts`

## Paso 5: Configurar la Ruta de la Página

### 1. Abrir el Archivo de Rutas del Módulo:

- Navega a `src/app/acceso/acceso-routing.module.ts` y verifica que la configuración de la ruta sea correcta:

```
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { AccesoPage } from './acceso.page';

const routes: Routes = [
  {
    path: '',
    component: AccesoPage
  }
];

@NgModule({
  imports: [RouterModule.forChild(routes)],
  exports: [RouterModule],
})
export class AccesoPageRoutingModule {}
```

## 2. Agregar la Ruta al Módulo Principal:

- Abre `src/app/app-routing.module.ts` y agrega la nueva ruta:

```
import { NgModule } from '@angular/core';
import { PreloadAllModules, RouterModule, Routes } from '@angular/router';

const routes: Routes = [
  {
    path: '',
    redirectTo: 'inicio',
    pathMatch: 'full'
  },
  {
    path: 'inicio',
    loadChildren: () => import('./inicio/inicio.module').then(m => m.InicioPageModule)
  }
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule {}
```

```

    },
    {
      path: 'acceso',
      loadChildren: () => import('./acceso/acceso.module').then(m => m.AccesoPageModule)
    }
  ];

  @NgModule({
    imports: [
      RouterModule.forRoot(routes, { preloadingStrategy: PreloadAllModules })
    ],
    exports: [RouterModule]
  })
  export class AppRoutingModule {}

```

## Paso 6: Ejecutar la Aplicación

### 1. Abrir la Terminal:

- Asegúrate de estar en el directorio raíz del proyecto:

```
cd C:\\Users\\marce\\OneDrive\\Escritorio\\ionic\\myApp
```

### 2. Ejecutar el Servidor de Desarrollo:

- Ejecuta el siguiente comando para iniciar el servidor de desarrollo:

```
ionic serve
```

### 3. Ver la Aplicación en el Navegador:

- Una vez que la aplicación esté ejecutándose, abre tu navegador y navega a <http://localhost:8100/acceso> para ver la nueva página **Acceso**.

## Posibles Errores y Soluciones

### 1. Error: `@angular/cli` es necesario:

- Solución: Instala `@angular/cli` globalmente:

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

### 2. Error: Faltan paquetes de Node.js:

- Solución: Asegúrate de que todas las dependencias estén instaladas:

```
npm install
```

### 3. Error: `ng.cmd` no se encuentra:

- Solución: Verifica que `@angular/cli` esté instalado globalmente y asegúrate de estar en el directorio correcto del proyecto.

### 4. Error de Rutas:

- Solución: Verifica que la ruta esté configurada correctamente en `app-routing.module.ts` y `acceso-routing.module.ts`.

### 5. Limpieza de Caché de npm:

- Si continúas teniendo problemas, intenta limpiar la caché de npm:

```
npm cache clean --force
```

### 6. Reinstalar Dependencias:

- Si los problemas persisten, elimina `node_modules` y `package-lock.json`, luego reinstala las dependencias:

```
rm -rf node_modules  
rm package-lock.json  
npm install
```