# **Manual de Instrucciones: Página Web de Panadería**

## **Introducción**

Este manual te guiará a través del proceso de encender y configurar la página web de la panadería. La aplicación está dividida en dos carpetas principales: **backend** y **main**. En el backend, utilizamos Node.js para gestionar el servidor y en el frontend, Angular se encarga de la interfaz de usuario. Además, la página incluye funcionalidades como login, registro, y la capacidad de arrastrar archivos Excel o CSV para procesarlos.

## **Capítulo 1: Estructura del Proyecto**

### **1.1 Carpetas Principales**

El proyecto está dividido en dos carpetas:

* **backend**: Aquí se encuentra el servidor de Node.js.
* **main**: Aquí está el código frontend de Angular.

### **1.2 Componentes Clave**

* **Backend (Node.js)**:  
  + **index.js**: Archivo principal donde se ejecuta el servidor.
  + **APIs**: Para manejar el login, el registro y otras interacciones de datos.
* **Frontend (Angular)**:  
  + **Componentes**: Incluye los formularios de login y registro, la visualización de datos, y la funcionalidad de arrastrar y soltar archivos.
  + **Servicios**: Se encarga de la comunicación con el backend, manejando autenticación, registros y archivos.

## **Capítulo 2: Instalación**

### **2.1 Requisitos Previos**

Antes de empezar, asegúrate de tener instalados los siguientes programas:

* **Node.js** (versión 14 o superior)
* **Angular CLI** (instalado globalmente)
* **Git** (opcional, para el control de versiones)

### **2.2 Instalación del Backend**

Ejecuta el siguiente comando para instalar las dependencias del backend:

npm install

Una vez que se hayan instalado las dependencias, ejecuta el servidor con:  
node index.js

1. Esto iniciará el servidor en el puerto configurado (por defecto en el puerto 3000).

### **2.3 Instalación del Frontend (Angular)**

Instala las dependencias del proyecto frontend ejecutando:  
npm install

Una vez instaladas las dependencias, puedes arrancar la aplicación Angular con:  
ng serve -o

Esto abrirá la aplicación en tu navegador, normalmente en:

http://localhost:4200.

## **Capítulo 3: Funcionalidades**

### **3.1 Login y Registro**

1. **Login**: Los usuarios pueden acceder a la página web con su correo y contraseña. Si los datos son correctos, se les permitirá acceder a la zona privada de la página.
2. **Registro**: Los nuevos usuarios pueden registrarse proporcionando su nombre, correo electrónico, y contraseña. El sistema valida que no exista ya un usuario con el mismo correo.
3. **Autenticación**: Las credenciales se gestionan de manera segura en el backend utilizando JWT (JSON Web Tokens) para mantener la sesión activa.

### **3.2 Arrastrar y Soltar Archivos**

1. La funcionalidad de arrastrar y soltar permite a los usuarios subir archivos Excel, CSV, o cualquier otro formato compatible para procesar los datos.
2. **Procesamiento de Archivos**:  
   * El sistema analiza el archivo subido y convierte los datos a un formato adecuado para su visualización en la página.
   * Los usuarios pueden ver los datos en una tabla interactiva.
3. Para cargar un archivo:  
   * Arrastra y suelta el archivo en la zona indicada en la interfaz de usuario.
   * El sistema automáticamente detectará el formato y lo procesará.

## **Capítulo 4: Despliegue**

### **4.1 Despliegue del Backend**

Para desplegar el backend en un servidor de producción, sigue estos pasos:

1. Configura las variables de entorno (como la base de datos y el puerto) en el archivo .env.

Ejecuta el backend en un servidor de producción utilizando PM2 o algún otro manejador de procesos.  
  
  
pm2 start index.js

### **4.2 Despliegue del Frontend**

Para desplegar el frontend en producción:

Realiza un build de la aplicación Angular:  
ng build --prod

1. Sube los archivos generados en la carpeta dist/ a tu servidor web (por ejemplo, a través de FTP o un servicio de hosting como Firebase, Netlify o AWS).

## **Capítulo 5: Solución de Problemas Comunes**

### **5.1 Backend**

* **Error de puerto ocupado**: Si el puerto por defecto (3000) está ocupado, cambia el puerto en el archivo index.js y en las variables de entorno.
* **Dependencias faltantes**: Si alguna dependencia no se ha instalado correctamente, ejecuta nuevamente npm install.

### **5.2 Frontend**

* **No se carga la aplicación**: Si al ejecutar ng serve -o la aplicación no se abre, asegúrate de que Node.js esté correctamente instalado y que las dependencias estén bien instaladas.
* **Error al arrastrar y soltar archivos**: Verifica que la librería de procesamiento de archivos (por ejemplo, xlsx) esté correctamente instalada.

## **Capítulo 6: Personalización y Mejora**

### **6.1 Mejorar el Diseño**

Puedes personalizar los estilos utilizando **CSS** o **Tailwind CSS** para que el sitio tenga un aspecto más atractivo. Angular permite utilizar el sistema de componentes para modularizar y reutilizar el diseño.

### **6.2 Agregar Funcionalidades**

* **Reportes**: Agrega funcionalidades adicionales para que los usuarios puedan generar reportes con los datos cargados de los archivos.
* **Notificaciones**: Implementa un sistema de notificaciones para avisar a los usuarios cuando el archivo se ha cargado correctamente o cuando ocurre un error.

## **Conclusión**

Con estos pasos, tu página web de panadería estará lista para usarse tanto en desarrollo como en producción. La estructura modular de este proyecto facilita la incorporación de nuevas funcionalidades o el ajuste de las existentes. ¡Disfruta trabajando en tu proyecto!