

# *Manual de instalación IEF*

## 1. Introducción

Bienvenido al manual de usuario de la **Aplicación Web de Generación de Excel**. Esta aplicación permite a los usuarios cargar archivos, procesarlos y generar un archivo Excel de salida con los datos combinados o transformados según las necesidades del sistema.

## 2. Requisitos del Sistema

Antes de comenzar a usar la aplicación, asegúrate de que tu sistema cumpla con los siguientes requisitos:

- **Sistema Operativo:** Windows, macOS, o Linux.
- **Navegador Web:** Google Chrome, Mozilla Firefox o cualquier navegador moderno.
- **Requisitos de Software:**
  - **Node.js** (v20.17.0 o superior)
  - **npm** (v10.8.2 o superior)
  - **Docker** (en caso de querer usarlo)
- **Requisitos de hardware recomendados:**
  - Mínimo 4 GB de RAM.
  - Procesador multinúcleo.
  - Almacenamiento suficiente para los archivos procesados.

## 3. Instalación

### 3.1. Clonación del Repositorio

Para comenzar, primero debes clonar el repositorio en tu máquina local. Sigue estos pasos:

- i. Abre una terminal en tu máquina.
- ii. Ejecuta el siguiente comando para clonar el repositorio:
- iii. `git clone <URL del repositorio>`
- iv. Navega a la carpeta del repositorio clonado:
- v. `cd <nombre del repositorio>`

### 3.2. Instalación de Dependencias

Una vez dentro del directorio del proyecto, instala las dependencias necesarias tanto en la carpeta **Frontend** y **Backend**:

- i. `npm install`

### 3.3. Configuración del Entorno

- i. Crea un archivo `.env` en la raíz del proyecto y agrega las variables necesarias según las configuraciones del sistema. Puedes seguir los ejemplos proporcionados en el archivo `.env.example`.

## 4. Ejecución de la Aplicación

### 4.1. Iniciar el Frontend

Para iniciar la interfaz de usuario (frontend), navega a la carpeta **Frontend** y ejecuta el siguiente comando.

***npm start***

Esto iniciará el servidor de desarrollo y podrás acceder a la aplicación en tu navegador en la dirección `http://localhost:3009` (o el puerto que se haya configurado).

### 4.2. Iniciar el Backend

Para iniciar el servidor backend que procesa los datos y genera el archivo Excel, navega a la carpeta **Backend** y ejecuta:

***node index.js***

Esto iniciará el servidor backend en el puerto configurado, generalmente en `http://localhost:4009`.

### 4.3. Usar Docker (Opcional)

Requisitos previos

1. Tener Docker y Docker Compose instalados.
  - a. En una consola ejecutar : `docker --version`.
  - b. En una consola ejecutar : `docker-compose --version`
  - c. En caso de que salga la versión significa que ya está instalado, en caso de que no, se debe instalar.
2. Asegurarte de tener los siguientes archivos en la raíz del proyecto:
  - a. `docker-compose.yml` en la raíz del proyecto
  - b. `Dockerfile` en la carpeta frontend y backend.

#### 4.3.1 Crear las imágenes docker y ejecutar en carpeta raíz

Si prefieres ejecutar la aplicación en contenedores, puedes usar Docker y Docker Compose. Para ello:

1. Abrir consola en la carpeta raíz del proyecto.

2. Ejecuta el siguiente comando para levantar ambos servicios (frontend y backend) con Docker Compose:
  - a. En caso de que haya hecho un cambio en los archivos: `docker-compose up --build -d`
  - b. En caso de que no: `docker-compose up -d`
3. Para ver la ejecución del programa: `docker-compose logs -f`
4. Para finalizar la ejecución: `docker-compose down`

### **4.3.2 Crear las imágenes y ejecutarlas en otro directorio**

Asegurarse de tener en el directorio en donde se moverá:

- a. Archivo `docker-compose.yml`.
- b. Carpeta Backend, dentro de Backend la carpeta storage y dentro de storage los archivos Excel.
- c. Carpeta Frontend y dentro de esta el archivo `.env`.
- d. Asegurarse de que el archivo `docker-compose.yml` tenga :
  - i. En la sección de frontend: `image: project_taller-frontend`
  - ii. En la sección de backend: `image: project_taller-backend`
  - iii. En caso de tener la sección del build, eliminarla.

En la carpeta raíz ejecutar pasos 1 y 2 de **4.3.1**

1. Crear los 2 archivos en la carpeta raíz (carpeta en donde está alojado el proyecto):
  - a. `docker save -o frontend.tar project_taller-frontend`
  - b. `docker save -o backend.tar project_taller-backend`
2. Detener contenedor: `docker-compose down`
3. Mover los archivos generados en la carpeta raíz (`frontend.tar` y `backend.tar`) hacia el destino que desee.
4. abrir consola en el directorio en donde se movieron los archivos.
5. Cargar las imágenes:
  - a. `docker load -i frontend.tar`
  - b. `docker load -i backend.tar`
6. ejecutar contenedor: `docker-compose up -d`
7. Para ver la ejecución del programa: `docker-compose logs -f`
8. Para finalizar la ejecución: `docker-compose down`.