

**Sistema de Detección de Plagio Académico**  
**PlagiTracker**

**GUÍA DE INSTALACIÓN**

**Versión 1.1**

PlagiTracker
Documento de Guía de Instalación

## 1. Índice:

<b>Portada .....</b>	<b>1</b>
<b>Índice .....</b>	<b>2</b>
<b>Introducción .....</b>	<b>3</b>
<b>Requisitos Previos .....</b>	<b>3</b>
<b>Descarga del Software .....</b>	<b>4</b>
<b>Instalación Paso a Paso .....</b>	<b>5</b>
<b>Configuración Adicional .....</b>	<b>7</b>
<b>Resolución de Problemas .....</b>	<b>10</b>

PlagiTracker
Documento de Guía de Instalación

## 2. Introducción:

Esta guía de instalación proporciona las instrucciones necesarias para descargar, instalar y configurar correctamente el software. Incluye los requisitos previos, pasos detallados para la instalación, configuración adicional opcional y pruebas básicas para verificar su correcto funcionamiento. Además, se ofrecen soluciones a problemas comunes que puedan surgir durante el proceso.

Esta guía está dirigida a **docentes y alumnos** que utilizarán el sistema de detección de plagio. Los docentes podrán emplear la herramienta para analizar y evaluar posibles casos de plagio en códigos fuente, mientras que los alumnos podrán utilizarla para verificar y mejorar la originalidad de sus trabajos antes de entregarlos.

## 3. Requisitos Previos:

Para instalar y utilizar el sistema de detección de plagio, asegúrese de cumplir con los siguientes requisitos:

### Hardware Necesario

- Procesador: Mínimo 4 GHz, multinúcleo recomendado.
- Memoria RAM: 8 GB o superior.
- Espacio en Disco: 10 GB disponibles para instalación y almacenamiento de datos.

### Software Requerido

1. **Navegador Web:**
  - Firefox (última versión) o Chrome (última versión).
2. **Herramientas de Desarrollo y Ejecución:**
  - **Docker:** Para contenerizar y ejecutar servicios del sistema.
  - **Vue 3:** Framework para la interfaz de usuario.
  - **Node.js:** Versión 20 o superior para gestionar dependencias y ejecutar scripts de frontend.
  - **Visual Studio 2022:** Compatible con .NET 8 para desarrollo y compilación del backend.
3. **Base de Datos:**
  - **PostgreSQL 16** para almacenar datos relacionados con el sistema.
  - **DBeaver:** Es una aplicación de software cliente de SQL y una herramienta de administración de bases de datos.
4. **Conexión a Internet:**
  - Conexión estable requerida para descargar dependencias, imágenes de Docker y para el correcto funcionamiento del sistema.

### Permisos Requeridos

PlagiTracker
Documento de Guía de Instalación

- Permisos de administrador en el equipo para instalar software y configurar Docker.

Asegúrese de que su entorno cumple con todos los requisitos antes de proceder con la instalación.

#### 4. Descarga del Software:

A continuación, se proporcionan los enlaces oficiales para descargar los componentes necesarios para instalar y ejecutar el sistema de detección de plagio:

##### 1. Navegador Web

- **Firefox:** <https://www.mozilla.org/firefox/download>
- **Google Chrome:** <https://www.google.com/chrome/>

##### 2. Herramientas de Desarrollo y Ejecución

- **Docker:** <https://www.docker.com/get-started/>
- **Vue 3:** No es necesario descargar un instalador, ya que Vue se instala como dependencia en un proyecto Node.js. Consulte las instrucciones en <https://vuejs.org/guide/quick-start.html>.
- **Node.js (v20 o superior):** <https://nodejs.org/en/download/>
- **Dbeaver Community:** <https://dbeaver.io/download/>
- **Visual Studio 2022 (con .NET 8):** <https://visualstudio.microsoft.com/vs/>

##### 3. Base de Datos

- **PostgreSQL 15:** <https://www.postgresql.org/download/>

##### 4. Conexión a Internet

No se requiere descarga, pero asegúrese de tener acceso a una conexión estable para descargar estos componentes y para el funcionamiento del sistema.

---

**Nota:** Es importante descargar las versiones más recientes de cada software desde sus páginas oficiales para garantizar compatibilidad y seguridad.

PlagiTracker
Documento de Guía de Instalación

## 5. Instalación Paso a Paso:

### 5.1 Configuración de PostgreSQL

#### 1. Ejecutar el Contenedor de PostgreSQL

Abra **PowerShell** con permisos de administrador y ejecute el siguiente comando para crear un contenedor Docker con PostgreSQL:

Copiar código

```
docker run --name plagitrackerDb -e POSTGRES_PASSWORD=admin -d -p 5000:5432 postgres
```

-Si les sale un error indicando que el puerto está ocupado se tendrá que cambiar el puerto 5000 por alguno otro que no esté ocupado (si se cambia se tendrá que cambiar en todos los demás lugares donde se usa el mismo puerto)

#### 2. Verificar la Creación del Contenedor

Una vez ejecutado el comando, acceda al contenedor mediante Docker Desktop o ejecutando en el powershell:

```
docker ps
```

Confirme que el contenedor **plagitrackerDb** esté en ejecución.

Una vez creado el contenedor procederemos a clonar los proyectos.

También puede verificar si el contenedor se creó en la Aplicación desktop de Docker

### 5.2 Clonación del Proyecto

#### 1. Crear Carpeta de Trabajo

- Cree una carpeta en cualquier parte de su disco duro para organizar los archivos del proyecto. Por ejemplo:

```
■ C:\Plagitracker
```

#### 2. Clonar el Backend

- Abra la dentro de la carpeta creada y cree una nueva Carpeta llamada "PlagiTrackerBackend".
- Abra **cmd** (línea de comandos) dentro de la carpeta creada.
- Ejecute el siguiente comando para clonar el repositorio del Backend:

Copiar código

```
git clone --single-branch --branch LuqueBackend https://github.com/lfuis201/Plagitracker
```

○

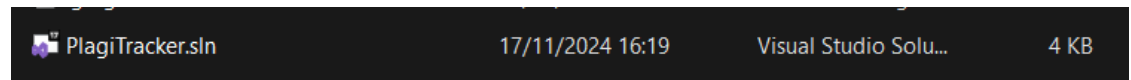
#### 3. Clonar el Frontend

- Abra la dentro de la carpeta creada y cree una nueva Carpeta llamada “PlagiTrackerFrontend”.
- Abra **cmd** (línea de comandos) dentro de la carpeta creada.
- Ejecute el siguiente comando para clonar el repositorio del Backend:  
Copiar código  
`git clone --branch Frontend https://github.com/lfuis201/Plagitrapper.git`

### 5.3 Configuración del Backend

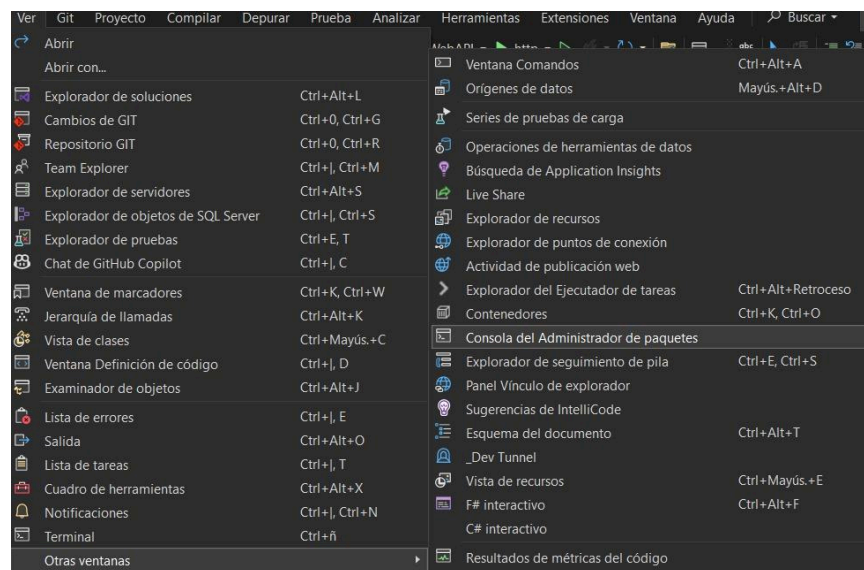
#### 1. Abrir el Proyecto en Visual Studio

- Dentro de la carpeta que creó usted tendrá que entrar a **PlagitrapperBackend\Backend\PlagiTracker**, dentro de esa dirección busque y abra el archivo **.sln** con **Visual Studio 2022**.

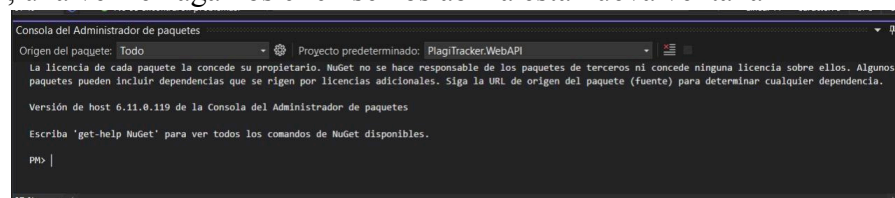


#### 2. Actualizar la Base de Datos

- Abra la consola del administrador de paquetes en Visual Studio.



- En la imagen se muestra como acceder a la Consola de administrador de paquetes, una vez le hagamos click se nos abrirá esta nueva ventana



PlagiTracker
Documento de Guía de Instalación

- En esta ventana tendremos que ejecutar el siguiente comando

Copiar código:

```
Update-Database -Project PlagiTracker.Data -StartupProject
PlagiTracker.WebAPI
```

- Una vez terminado de ejecutar el código, la base de datos se habrá creado y se habrán creado las tablas necesarias para poder hacer las pruebas correspondientes.
- Esta configuración solo se tendrá que ejecutar la primera vez que se ejecute el proyecto de la BD si el Docker la BD se eliminan tendrán que volver a ejecutarlo.
- Ahora se tendrá que probar en el Dbeaver para verificar la creación de la BD con sus tablas

### 3. Conectar a Dbeaver

Abra DBeaver y cree una nueva conexión.

Seleccione PostgreSQL como el tipo de base de datos.

Complete los campos con la siguiente información:

- Host: localhost
- Port: 5000 (o el que haya configurado)
- Database: PlagiTracker (importante poner el nombre para que la encuentre)
- Username: postgres
- Password: admin

Tal como se muestra en la imagen:

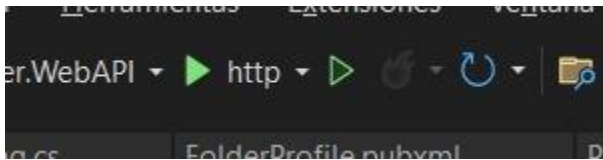
Haga clic en **Test Connection** para verificar la conexión.

Una vez verificada la conexión, podrá realizar consultas SQL o inspeccionar la base de datos.

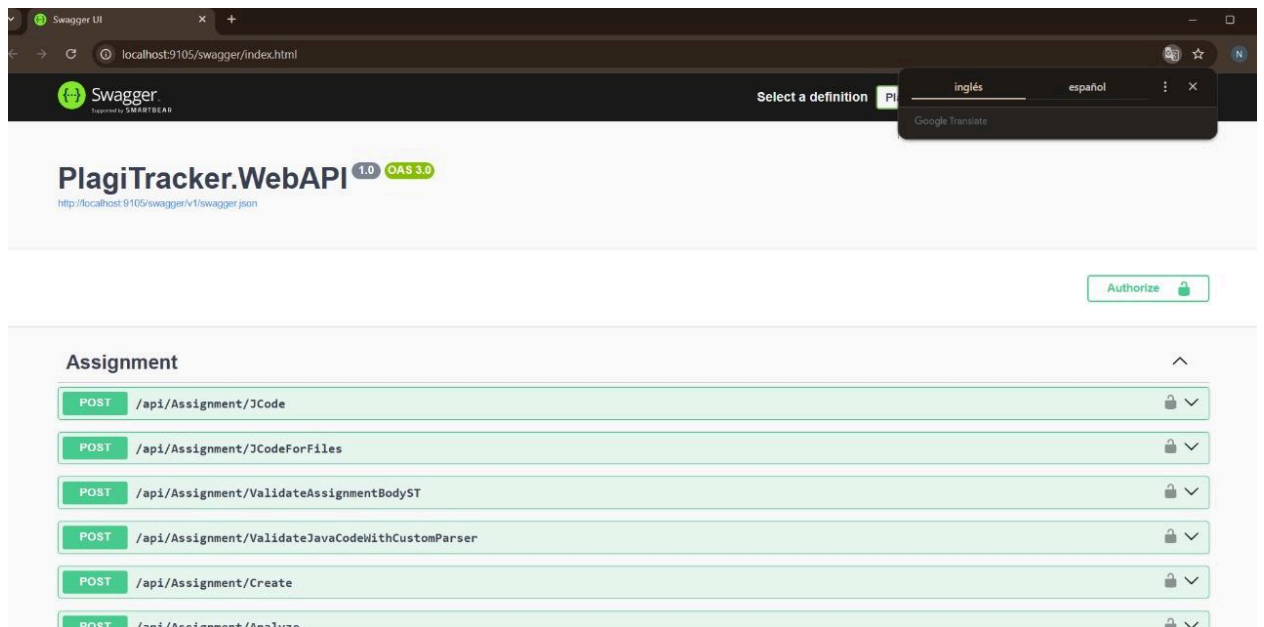
Tendrá que crear un script en la base de datos para probar las consultas que están en: <https://codeshare.io/R7wBVB>, dichos queries serán importantes para ver las inserciones de la BD y como están estructuradas las tablas.

#### 5.4. Visualización de Backend:

Una vez configurada la BD y viendo que está operativa volveremos al archivo .sln del Backend y tendremos que darle click al botón de “play” como en la imagen.



Y se nos abrirá una ventana del navegador con “Swagger” donde se podrán visualizar los “endpoints”

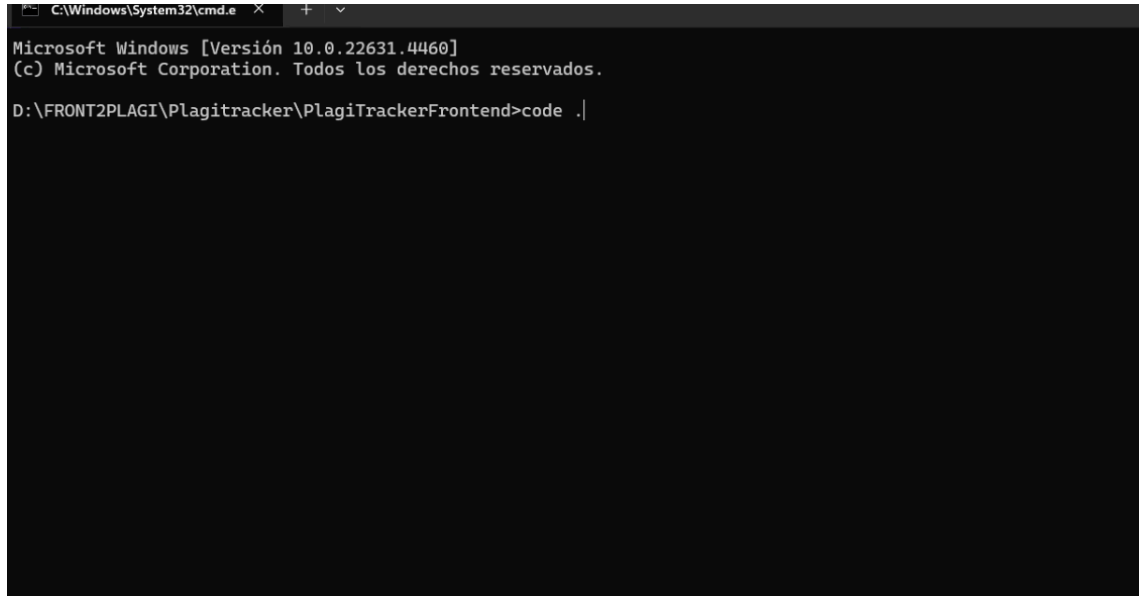




## 5.5 Configuración del Frontend

### 1. Instalar Dependencias del Frontend

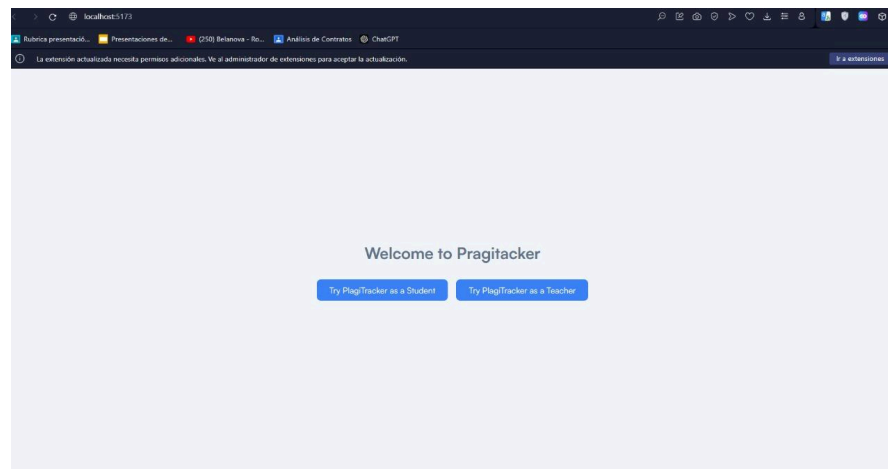
- Diríjase a la carpeta donde descargó el proyecto y entre a la carpeta “PlagiTrackerFronted” y dentro de esa dirección abra un “cmd” y escriba “code .” Tal como se ve en la imagen.



- Una vez presionado Enter , se nos abrirá una ventana de VS Code , en la cual tendremos que abrir una nueva terminal con “**Ctrl + Shift + Ñ**”, en dicha terminal tendremos que ejecutar el siguiente comando : **npm install** (se necesita tener el Node.js instalado para poder usar ese comando)

### 2. Iniciar el Servidor de Desarrollo

- Una vez instaladas las dependencias, inicie el servidor de desarrollo ejecutando:  
Copiar código  
**npm run dev**
- Acceda al frontend en el navegador en la URL que indique la terminal (por ejemplo, <http://localhost:8080>).

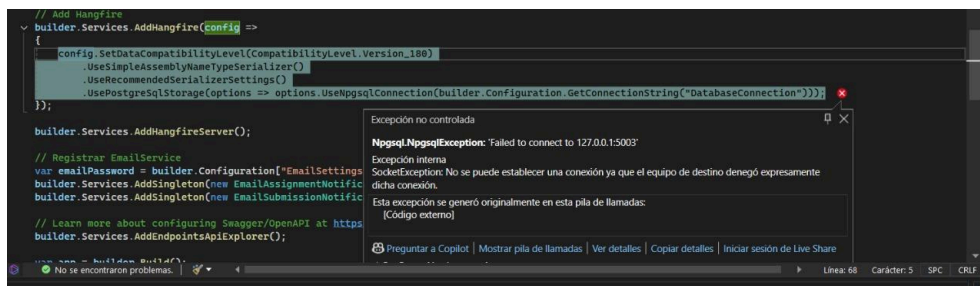


- 
- Una vez dentro del localhost se nos dará la bienvenida a “PlagiTracker”, viendo que el Frontend se levantó de manera exitosa.

## 6. Resolución de Problemas:

Si experimenta problemas durante el proceso de instalación o configuración, consulte las soluciones a continuación para resolver los inconvenientes más comunes:

### 6.1 Problemas de conexión a la hora de crear o Conectar la Base de datos



Solución:

A la hora de crear el Docker cuando ingrese el comando PostgreSQL: `docker run --name plagitrackerDb -e POSTGRES_PASSWORD=admin -d -p 5000:5432 postgres`.

Cambien el puerto 5000 a otro que no esté ocupado Ejem 5005

Una vez cambiado el docker tendrá que ir al Backend al archivo .sln y ejecutarlo, una vez dentro ir al archivo appsetting.json, y cambiar el puerto de la base datos con el que puso en el docker, en la imagen se refleja el cambio.

```
    },
    "AllowedHosts": "*",
    "ConnectionStrings": {
      "DatabaseConnection": "Server=localhost;Port=5005;Database=PlagiTracker;User Id=postgres;Password=admin;"
    },
    "Kestrel": {
      "Endpoints": {
```

Nota: Para hacer la conexión el Dbeaver para realizar las “queries” se requerirá que apunte se apunte al puerto con el que se creó el Docker.

## 6.2 Problemas para encontrar el localhost por que ya está en uso



Solución:

En archivo appsetting.json se tendrá que cambiar el puerto Kestrel de los Endpoints para a unos que no estén siendo utilizados. Como se ve en la imagen esos son los puertos que se tienen que cambiar para que la aplicación funcione correctamente.

```
    "Kestrel": {
      "Endpoints": {
        "Http": {
          "Url": "http://localhost:9100"
        },
        "Https": {
          "Url": "https://localhost:9200"
        }
      }
    }
```