



VERSIÓN 1.0

27 DE NOVIEMBRE DE 2025

Manual de Instalación

ESCUELA POLITECNICA NACIONAL – FACULTAD DE INGENIERIA DE SISTEMAS
CONSTRUCCIÓN Y EVOLUCIÓN DE SOFTWARE
[PROYECTO-TRANSCRIPTOR-BRAILLE]



CONTENIDO

1.	REQUISITOS PREVIOS A LA INSTALACIÓN.....	2
2.	INSTALACIÓN DEL SOFTWARE	2
2.1.	CLONACIÓN DEL REPOSITORIO	2
2.1.1.	En un directorio de preferencia, abrir una terminal o consola.	2
2.1.2.	Ejecutar los siguientes comandos en el orden indicado.	2
2.2.	VERIFICACIÓN DE REQUISITOS PREVIOS.....	2
2.3.	CONSTRUCCIÓN E INCIALIZACIÓN DE LOS CONTENEDORES	3
2.4.	VERIFICACION DE FUNCIONAMIENTO.....	3
2.5.	DETENCIÓN DE SERVICIOS.....	4

1. REQUISITOS PREVIOS A LA INSTALACIÓN

Antes de instalar el software de este manual, asegúrese de que el ordenador cumpla con los siguientes requisitos.

Sistema Operativo	Windows 10/11, Linux o macOS
Memoria RAM	Mínimo 4GB de RAM libre
Software necesario	Docker (versión recomendada: ≥ 20.10) Docker Compose (versión recomendada: ≥ 1.29 o integrar con Docker Desktop que ya lo incluya) Git
Puertos disponibles	Puerto 8000 Puerto 3000

2. INSTALACIÓN DEL SOFTWARE

2.1. CLONACIÓN DEL REPOSITORIO

2.1.1. En un directorio de preferencia, abrir una terminal o consola.

2.1.2. Ejecutar los siguientes comandos en el orden indicado.

git clone <https://github.com/KealPetu/Proyecto-transcriptor-braille>

```
Paul@DESKTOP-171FH8K MINGW64 ~/Desktop/Repositorios
$ git clone https://github.com/KealPetu/Proyecto-transcriptor-braille
Cloning into 'Proyecto-transcriptor-braille'...
remote: Enumerating objects: 211, done.
remote: Counting objects: 100% (74/74), done.
remote: Compressing objects: 100% (53/53), done.
remote: Total 211 (delta 21), reused 51 (delta 14), pack-reused 137 (from 1)
Receiving objects: 100% (211/211), 393.16 KiB | 1.47 MiB/s, done.
Resolving deltas: 100% (60/60), done.
```

cd Proyecto-transcriptor-braille

```
Paul@DESKTOP-171FH8K MINGW64 ~/Desktop/Repositorios
$ cd Proyecto-transcriptor-braille/

Paul@DESKTOP-171FH8K MINGW64 ~/Desktop/Repositorios/Proyecto-transcriptor-braille
(main)
$ |
```

2.2. VERIFICACIÓN DE REQUISITOS PREVIOS

Antes de continuar, se debe verificar que el equipo cumple con los requisitos establecidos en la **sección 1** de este manual. Específicamente:

- Docker y Docker Compose están instalados correctamente.
- Los puertos **3000** y **8000** se encuentran disponibles.

```
Paul@DESKTOP-171FH8K MINGW64 ~/Desktop/Repositorios/Proyecto-transcriptor-braille
(main)
$ docker --version
Docker version 28.3.0, build 38b7060

Paul@DESKTOP-171FH8K MINGW64 ~/Desktop/Repositorios/Proyecto-transcriptor-braille
(main)
$ docker compose version
Docker Compose version v2.38.1-desktop.1
```

2.3. CONSTRUCCIÓN E INICIALIZACIÓN DE LOS CONTENEDORES

En la raíz del proyecto (ubicación del archivo `docker-compose.yml`) ejecutar el siguiente comando:

docker compose up --build -d

Este comando descargará las imágenes necesarias, instalará dependencias y levantará el frontend y backend.

```
Paul@DESKTOP-171FH8K MINGW64 ~/Desktop/Repositorios/Proyecto-transcriptor-braille (main)
$ docker compose up --build -d
time="2025-11-27T17:29:40-05:00" level=warning msg="C:\\Users\\Paul\\Desktop\\Repositori
s\\Proyecto-transcriptor-braille\\docker-compose.yml: the attribute `version` is obsolet
, it will be ignored, please remove it to avoid potential confusion"
[+] Building 1.4s (23/23) FINISHED
=> [internal] load local bake definitions                                0.0s
=> => reading from stdin 888B                                          0.0s
=> [frontend internal] load build definition from Dockerfile           0.0s
=> => transferring dockerfile: 576B                                    0.0s
=> [backend internal] load build definition from Dockerfile           0.0s
=> => transferring dockerfile: 727B                                    0.0s
=> [frontend internal] load metadata for docker.io/library/node:18-alpine 0.5s
=> [backend internal] load metadata for docker.io/library/python:3.11-slim 0.6s
=> [frontend internal] load .dockerignore                             0.0s
=> => transferring context: 2B                                         0.0s
=> [backend internal] load .dockerignore                             0.0s
=> => transferring context: 2B                                         0.0s
```

Al finalizar este proceso se debe comprobar que la construcción de las imágenes fue correcta y que los contenedores se encuentran en ejecución.

```
[+] Running 4/4
✓ backend          Built
✓ frontend         Built
✓ Container braille_backend Started
✓ Container braille_frontend Started
```

2.4. VERIFICACION DE FUNCIONAMIENTO

Con los servicios en ejecución, se debe validar su correcto funcionamiento mediante las siguientes direcciones:

- Servicio de Backend

Servicio	URL	Estado esperado
Backend (FastAPI)	http://localhost:8000/docs	Se debe visualizar la documentación Swagger de la API del sistema

Braille Translator API 1.0.0 GA5.1
OpenAPI JSON
 API para la traducción bidireccional Español <-> Braille con generación de imágenes y PDFs

Translation	
POST	/api/v1/translation/to-braille Translate To Braille
POST	/api/v1/translation/to-text Translate To Text
Generation	
POST	/api/v1/generation/image Generate Image
POST	/api/v1/generation/pdf Generate Pdf
GET	/api/v1/generation/formats Get Available Formats

- Servicio de Frontend

Servicio	URL	Estado esperado
Frontend (React)	http://localhost:3000	Se debe visualizar la interfaz del sistema



2.5. DETENCIÓN DE SERVICIOS

Dependiendo de la acción que el usuario desee realizar, puede utilizar los siguientes comandos para detener correctamente los servicios.

Comando	Descripción
<code>docker compose down</code>	Detiene y elimina contenedores
<code>docker compose down --volumes</code>	Detiene y elimina contenedores junto a sus volúmenes asociados
<code>docker compose down --rmi all</code>	Detiene y elimina los contenedores, los volúmenes e imágenes generadas.
<code>docker compose stop</code>	Detiene temporalmente los contenedores sin eliminarlos.