

# Escuela Politecnica Nacional

Facultad de Ingeniería de Sistemas

## Construcción y Evolución de Software

---

|                 |     |
|-----------------|-----|
| <b>Versión:</b> | 1.0 |
|-----------------|-----|

|               |   |
|---------------|---|
| <b>Grupo:</b> | 5 |
|---------------|---|

|               |                |
|---------------|----------------|
| <b>Fecha:</b> | Noviembre 2025 |
|---------------|----------------|

---

# Manual de Instalación

---

## Conversor de Texto a Braille

Una aplicación web desarrollada con Flask que permite convertir texto normal a símbolos Braille Unicode.

### Características

- ☒ Conversión en tiempo real de texto a Braille
- ☒ Soporte para letras (a-z), números (0-9) y algunos símbolos de puntuación
- ☒ Interfaz moderna y responsiva
- ☒ Función para copiar el texto en Braille al portapapeles
- ☒ Diseño accesible y fácil de usar

### Requisitos

- Python 3.7 o superior
- pip (gestor de paquetes de Python)

### Instalación

#### 1. Clonar o descargar el repositorio

#### 2. Crear un entorno virtual (recomendado)

```
python -m venv .venv
```

#### 3. Activar el entorno virtual

- En Windows (Git Bash):

```
source .venv/Scripts/activate
```

- En Windows (CMD):

```
.venv\Scripts\activate.bat
```

- En Linux/Mac:

```
source .venv/bin/activate
```

## 4. Instalar las dependencias

```
pip install -r requirements.txt
```

## Uso

### 1. Iniciar la aplicación

```
python app.py
```

### 2. Abrir en el navegador

- Ir a: <http://localhost:5000>
- O usar la IP mostrada en la terminal

### 3. Usar el conversor

- Escribir el texto en el área de entrada
- Hacer clic en "Convertir a Braille ⚡"
- Ver el resultado en símbolos Braille
- Opcionalmente, copiar el resultado con el botón "📋 Copiar Braille"

## Estructura del Proyecto

```
Proyecto/
├── app.py                # Aplicación Flask principal
├── requirements.txt      # Dependencias del proyecto
├── templates/
│   └── index.html       # Plantilla HTML principal
├── static/
│   └── css/
│       └── style.css    # Estilos CSS
├── Documentation/       # Documentación Sphinx
│   ├── source/          # Archivos fuente (.rst)
│   ├── build/           # Documentación generada (HTML)
│   └── README.md        # Instrucciones de documentación
├── generar_docs.sh      # Script para generar docs (Bash)
├── generar_docs.bat     # Script para generar docs (Windows)
└── README.md            # Este archivo
```

## Documentación

Este proyecto incluye documentación completa generada con **Sphinx**.

Ver la Documentación desde el apartado como pythondoc

La documentación HTML ya está generada. Abre en tu navegador:

```
start Documentation/build/html/index.html
```

## Regenerar la Documentación

Si modificas el código y quieres actualizar la documentación:

### En Windows (Git Bash):

```
./generar_docs.sh
```






### En Windows (CMD):

```
generar_docs.bat
```

### Manualmente:

```
python -m sphinx -b html Documentation/source Documentation/build/html
```

La documentación incluye:

-  Introducción al sistema Braille
-  Instrucciones de instalación
-  Guía de uso (web y API)
-  Referencia completa de la API REST
-  Documentación automática del código (extraída de los docstrings)

## Tecnologías Utilizadas

- **Backend:** Flask (Python)
- **Frontend:** HTML5, CSS3, JavaScript (Vanilla)
- **Estilos:** CSS personalizado con gradientes y animaciones

## Caracteres Soportados

El conversor actualmente soporta:

- Letras del alfabeto (a-z)
- Números (0-9)

- Espacios
- Signos de puntuación básicos (. , ; : ! ? - ( ) )

## ⚙️ Características Futuras

- ☐ Soporte para más símbolos y caracteres especiales
- ☐ Conversión de Braille a texto normal
- ☐ Soporte para mayúsculas (indicador de mayúscula en Braille)
- ☐ Exportar resultado como imagen o PDF
- ☐ API REST para integración con otras aplicaciones

## 🤝 Contribuciones

Las contribuciones son bienvenidas. Si deseas mejorar este proyecto:

1. Fork el repositorio
2. Crea una rama para tu característica (`git checkout -b feature/nueva-caracteristica`)
3. Commit tus cambios (`git commit -m 'Agregar nueva característica'`)
4. Push a la rama (`git push origin feature/nueva-caracteristica`)
5. Abre un Pull Request

## 📝 Notas

- Los símbolos Braille se muestran usando caracteres Unicode de Braille Patterns (U+2800 a U+28FF)
- Para una mejor visualización, se recomienda usar navegadores modernos
- El sistema Braille aquí implementado es el Braille literario básico (Grado 1)

## 👤 Desarrollo

Este proyecto fue desarrollado como parte del curso de Construcción y Evolución del Software.

## 📄 Licencia

Este proyecto es de código abierto y está disponible para uso educativo.