



VERSIÓN 1.0
27 DE NOVIEMBRE DE 2025

Casos de Prueba

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



CONTENIDO

1.	Introducción de Pruebas.....	2
2.	Objetivo General de las Pruebas	2
2.1.	Objetivos específicos.....	2
3.	Alcance de las Pruebas.....	2
3.1.	Tipos de pruebas implementadas.....	2
3.2.	Pruebas unitarias.....	2
3.3.	Pruebas de integración.....	2
4.	Casos de Prueba	3
4.1.	Suite de pruebas: traductor (test_translator.py).....	3
4.2.	Suite de pruebas: generador de imágenes (test_generator.py).....	3
4.3.	Suite de pruebas: generador de pdfs (test_generator.py)	4
4.4.	Suite de pruebas: integración (test_generator.py).....	4
5.	Resultados y Análisis	4
6.	Anexos	5

1. INTRODUCCIÓN DE PRUEBAS

El presente documento detalla el proceso de pruebas implementado para validar la funcionalidad del sistema de generación y traducción Braille. Se han desarrollado dos suites de pruebas exhaustivas que verifican tanto la correcta traducción de texto a código Braille como la generación de representaciones visuales (imágenes PNG y documentos PDF).

2. OBJETIVO GENERAL DE LAS PRUEBAS

El objetivo principal del proceso de pruebas es garantizar que el sistema de generación Braille cumpla con los siguientes criterios de calidad:

2.1. Objetivos específicos

- **Validar la precisión** de la traducción texto-Braille según las normas internacionales
- **Verificar la correcta generación** de imágenes PNG con representaciones visuales de celdas Braille
- **Asegurar la creación válida** de documentos PDF con contenido Braille
- **Confirmar la bidireccionalidad** de las traducciones (texto → Braille → texto)
- **Probar el manejo correcto** de caracteres especiales, números y mayúsculas
- **Validar casos extremos** como strings vacíos, espacios múltiples y secuencias largas

3. ALCANCE DE LAS PRUEBAS

3.1. Tipos de pruebas implementadas

El proyecto incluye dos niveles complementarios de pruebas:

3.2. Pruebas unitarias

Verifican el comportamiento individual de cada componente del sistema:

- Inicialización correcta de generadores
- Traducción de caracteres individuales
- Aplicación correcta de prefijos (numéricos y de mayúsculas)
- Generación de elementos básicos (celdas, imágenes, páginas)

3.3. Pruebas de integración

Validan el flujo completo del sistema:

- Proceso end-to-end de generación de imágenes
- Proceso end-to-end de generación de PDFs
- Manejo de casos de uso reales (señalética de transporte)
- Compatibilidad entre módulos de traducción y generación

4. CASOS DE PRUEBA

4.1. Suite de pruebas: traductor (test_translator.py)

ID	Caso de Prueba	Entrada	Resultado Esperado	Prioridad
TR-01	Alfabeto básico	"hola"	4 celdas: [[1,2,5], [1,3,5], [1,2,3], [1]]	Alta
TR-02	Números continuos	"12"	Prefijo numérico + 2 celdas	Alta
TR-03	Números separados	"1 2"	Prefijo + celda + espacio + prefijo + celda	Alta
TR-04	Mayúsculas	"A"	Prefijo mayúscula [4,6] + celda [1]	Alta
TR-05	Mezcla letras-números	"Bus 15"	8 celdas con prefijos correctos	Alta
TR-06	Bidireccionalidad	"Hola 123 Mundo"	Roundtrip idéntico al original	Crítica
TR-07	Espacios múltiples	"a b"	2 espacios detectados correctamente	Media
TR-08	String vacío	""	Lista vacía []	Media
TR-09	Secuencia numérica	"12345"	Un solo prefijo numérico	Alta

4.2. Suite de pruebas: generador de imágenes (test_generator.py)

ID	Caso de Prueba	Entrada	Resultado Esperado	Prioridad
GI-01	Inicialización	N/A	cell_width=40, cell_height=60, dot_radius=6	Alta
GI-02	Imagen simple	"a"	PNG válido, dimensiones > 0	Alta
GI-03	Imagen con texto	"Hola", include_text=True	Altura > 100px (incluye texto)	Media
GI-04	Imagen sin texto	"Hola", include_text=False	Altura ≤ 100px (solo celdas)	Media
GI-05	Múltiples celdas	"abc"	Ancho ≥ 3×(40+10)+40 = 190px	Alta
GI-06	Función conveniencia	"test"	PNG válido usando función wrapper	Media

4.3. Suite de pruebas: generador de pdfs (test_generator.py)

ID	Caso de Prueba	Entrada	Resultado Esperado	Prioridad
GP-01	Inicialización	N/A	page_size = A4	Alta
GP-02	PDF simple	"Baño"	PDF válido, ≥1 página	Alta
GP-03	PDF con título	"Salida", title="..."	PDF con título personalizado	Media
GP-04	Texto largo	"Salida de emergencia..."	PDF con contenido completo	Alta
GP-05	Función conveniencia	"test", title="..."	PDF válido usando función wrapper	Media

4.4. Suite de pruebas: integración (test_generator.py)

ID	Caso de Prueba	Entrada	Resultado Esperado	Prioridad
IT-01	Flujo completo imagen	"Baño"	PNG válido RGB con dimensiones correctas	Crítica
IT-02	Flujo completo PDF	"Entrada principal"	PDF válido con 1 página	Crítica
IT-03	Caracteres especiales	"Baño - Señalización"	Imagen y PDF procesados correctamente	Alta
IT-04	Números en contexto	"Piso 3"	Imagen y PDF con números renderizados	Alta

5. RESULTADOS Y ANÁLISIS

Las pruebas realizadas evidenciaron un funcionamiento óptimo en todos los módulos (100% PASS). El módulo de **traducción** cumple correctamente con las normas Braille en letras, números, mayúsculas, espacios y casos extremos, garantizando bidireccionalidad sin pérdida. El módulo de **imágenes** genera archivos PNG válidos con parámetros adecuados y dimensiones dinámicas, diferenciando correctamente modos con y sin texto. El módulo de **PDF** produce documentos estructurados en formato A4 con contenido verificable y soporte para títulos personalizados. Finalmente, las **pruebas de integración** confirmaron el correcto flujo completo de generación tanto de imágenes como PDFs, incluidas entradas con caracteres especiales y casos de uso reales como señalética de baños, accesos y pisos.

Módulo	Total Pruebas	Exitosas	Fallidas	Tasa de Éxito
Traductor	9	9	0	100%
Generador Imágenes	6	6	0	100%
Generador PDFs	5	5	0	100%
Integración	4	4	0	100%
TOTAL	24	24	0	100%

6. ANEXOS

[1] Comandos de Ejecución para las pruebas: `pytest -vv --log-cli-level=INFO --html=report.html --self-contained-html --full-trace`

[2] Las evidencias y resultados detallados de cada ejecución individual se encuentran disponibles en el archivo **report.html**, ubicado en el mismo directorio del presente documento.