

Universidad de San Carlos de Guatemala

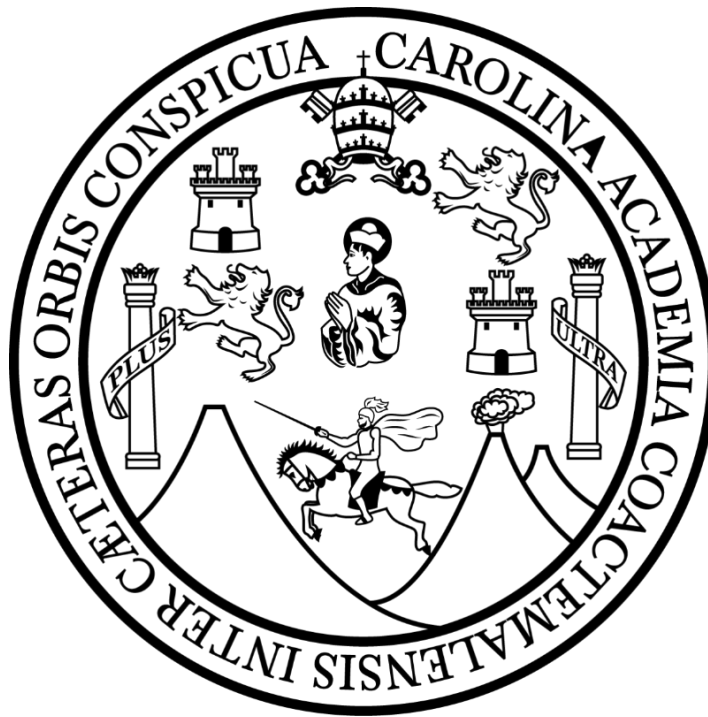
Facultad de Ingeniería

Curso: Lenguajes Formales

Sección: B-

Ing. Zulma Aguirre

Tutor Académico:



Proyecto#2

Nombre: Juan Carlos Humberto Reyes Chavarria







Carné: 202307699

Contenido

 Introducción.....	3
Características principales:.....	3
 Requisitos del Sistema	4
 Instalación y Ejecución.....	4
Paso 1: Abrir dos terminales	4
Paso 2: Abrir en el navegador	4
 Interfaz de Usuario	5
 Funcionalidades	5
1 Botón "  Nuevo"	5
2 Botón "  Abrir"	5
3 Botón "  Guardar"	6
4 Botón "  Guardar Como"	6
5 Botón "  Analizar"	7
 Ejemplos de Uso	9
Ejemplo 1: Traducción simple	9
Ejemplo 2: Estructuras de control	9
Ejemplo 3: Bucles	10
 Solución de Problemas.....	11
✗ "Error al analizar el código"	11
✗ "Primero debe generar la traducción"	11
✗ La página no carga	11

JavaBridge es una aplicación web que traduce código Java a código Python de manera automática. La herramienta analiza léxica y sintácticamente el código Java, genera reportes de errores y produce código Python equivalente.

Características principales:

-  Traducción automática de Java a Python
-  Análisis léxico y sintáctico
-  Reportes de errores detallados
-  Tabla de tokens generados
-  Carga y descarga de archivos
-  Interfaz amigable e intuitiva



Requisitos del Sistema

- **Navegador web moderno** (Chrome, Firefox, Edge, Safari)
- **Node.js** versión 16 o superior (solo para instalación)
- **Conexión a Internet** (para primera instalación)
- **Sistema Operativo:** Windows, macOS o Linux



Instalación y Ejecución

Paso 1: Abrir dos terminales

Terminal 1 - Backend:

```
cd c:\Users\juaan\Desktop\LFP\LFP_2S2025_202307699
```

```
npm install
```

```
npm run dev
```

Verás: **Servidor escuchando en http://localhost:4000**

Terminal 2 - Frontend:

```
cd c:\Users\juaan\Desktop\LFP\LFP_2S2025_202307699\Frontend
```

```
npm install
```

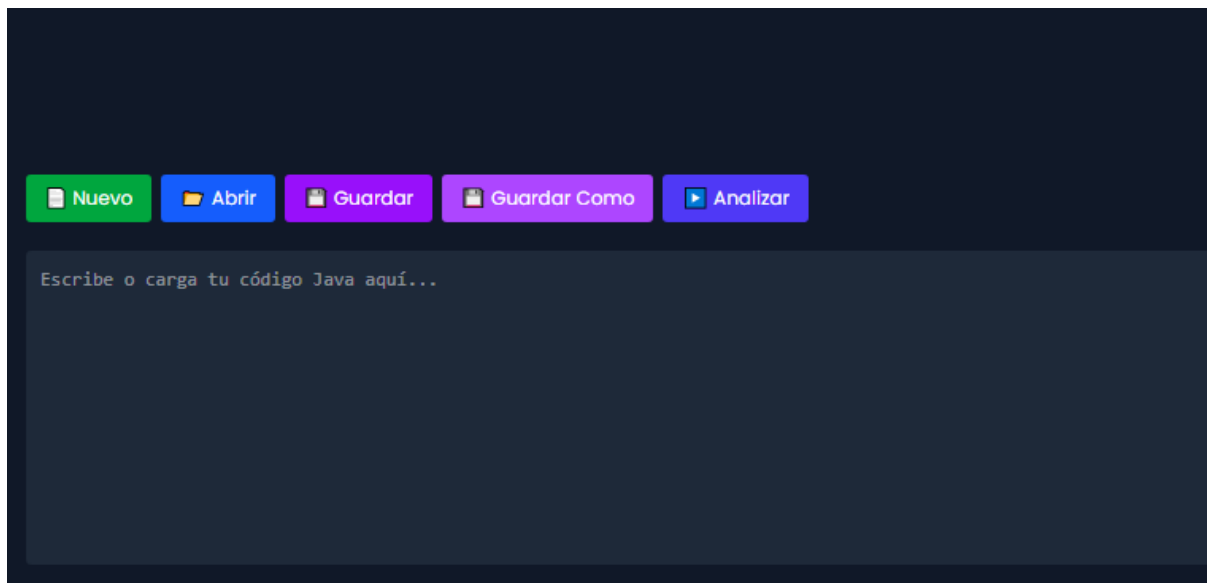
```
npm run dev
```

Verás una URL como: **http://localhost:5173**

Paso 2: Abrir en el navegador

Abre tu navegador en la URL proporcionada (generalmente `http://localhost:5173`)

Interfaz de Usuario




Funcionalidades

1 Botón " Nuevo"

Función: Limpia el área de entrada y todos los resultados para empezar un nuevo archivo.


Cómo usar:

1. Click en el botón " Nuevo"
2. Si hay cambios sin guardar, aparecerá una confirmación
3. Click en "Aceptar" para continuar o "Cancelar" para mantener el código actual

2 Botón " Abrir"

Función: Carga un archivo **.java** desde tu computadora.

Cómo usar:

1. Click en el botón " Abrir"
2. Selecciona un archivo **.java** o **.txt** de tu computadora
3. El contenido aparecerá en el área de entrada

4. El nombre del archivo se mostrará en la parte superior



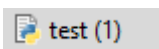
3 Botón "📄 Guardar"

Función: Descarga el código Python generado con el nombre del archivo original.

Cómo usar:

1. Primero debes hacer click en "▶ Analizar" para generar la traducción
2. Click en "📄 Guardar"
3. El archivo **.py** se descargará automáticamente

Nota: Si abriste un archivo llamado **Main.java**, se guardará como **Main.py**

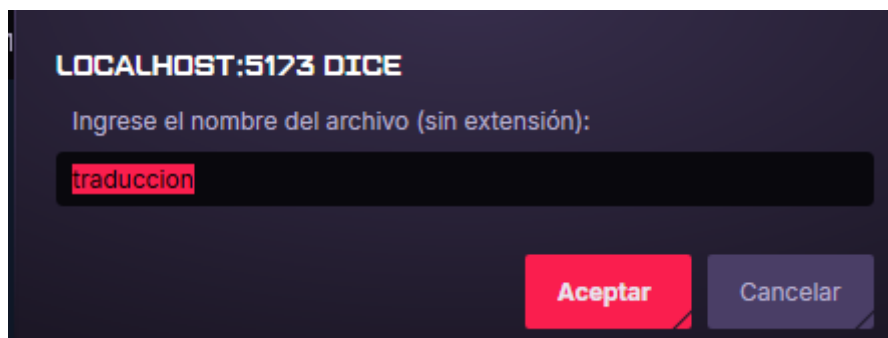


4 Botón "📄 Guardar Como"

Función: Descarga el código Python con un nombre personalizado.

Cómo usar:

1. Primero debes hacer click en "▶ Analizar" para generar la traducción
2. Click en "📄 Guardar Como"
3. Ingresa el nombre deseado (sin extensión)
4. El archivo se descargará con el nombre especificado + **.py**



5 Botón "▶ Analizar"

Función: Procesa el código Java y genera la traducción a Python.

Cómo usar:

1. Escribe o carga código Java en el área de entrada
2. Click en "▶ Analizar"
3. Verás los resultados en las secciones inferiores:
 - **Tokens:** Lista de elementos reconocidos
 - **Errores Léxicos:** Problemas en la estructura del código
 - **Errores Sintácticos:** Problemas en la gramática
 - **Código Python:** Traducción generada

Tokens						
#	Tipo	Valor	Línea	Columna		
1	PUBLIC	public	1	1		
2	CLASS	class	1	8		
3	IDENTIFICADOR	SistemaGestionAvanzado	1	14		
4	SIMBOLO	{	1	37		
5	PUBLIC	public	2	6		
6	STATIC	static	2	13		
7	VOID	void	2	20		
8	MAIN	main	2	25		
9	SIMBOLO	(2	29		
10	STRING_TYPE	String	2	30		
11	SIMBOLO	{	2	36		
12	ARGS	args	2	37		
13	SIMBOLO	}	2	43		
14	SIMBOLO	{	2	45		
15	STRING_TYPE	String	4	10		
16	IDENTIFICADOR	nombre	4	17		
17	OPERADOR	=	4	24		
18	STRING	Sistema Avanzado	4	26		
19	SIMBOLO	;	4	44		
20	INT_TYPE	int	5	10		
21	IDENTIFICADOR	edad	5	14		
22	OPERADOR	=	5	19		
23	INT	25	5	21		
24	SIMBOLO	;	5	23		
25	DOUBLE_TYPE	double	6	10		
26	IDENTIFICADOR	salario	6	17		
27	OPERADOR	=	6	25		
28	DOUBLE	50000.75	6	27		
29	SIMBOLO	;	6	35		
30	CHAR_TYPE	char	7	10		
31	IDENTIFICADOR	categoría	7	15		
32	OPERADOR	=	7	25		
33	CHAR	A	7	27		
34	SIMBOLO	;	7	30		
35	BOOLEAN_TYPE	boolean	8	10		
36	IDENTIFICADOR	activo	8	18		

Traducción a Python

```
nombre = "Sistema Avanzado"
edad = 25
salario = 50000.75
categoria = "A"
activo = True
bonus = 0.0
salarioFinal = 0.0
diasTrabajados = 250
tasaImpuesto = 0.15
totalProyectos = 5
productividad = 0.0
if edad >= 18:
    if salario > 40000.0:
        if diasTrabajados >= 240:
            bonus = salario * 0.10
            if totalProyectos > 3:
                bonus = bonus + salario * 0.05
            else:
                bonus = bonus + salario * 0.02
        else:
            bonus = salario * 0.05
    else:
        bonus = salario * 0.03
else:
    bonus = 0.0
salarioFinal = salario + bonus
impuesto = salarioFinal * tasaImpuesto
salarioNeto = salarioFinal - impuesto
print("Salario Neto: " + str(salarioNeto))
recibeBono = bonus > 0.0
esContratado = activo != False
```




Ejemplos de Uso

Ejemplo 1: Traducción simple

Código Java:

```
public class Main {  
    public static void main(String[] args) {  
        int numero = 42;  
        System.out.println(numero);  
    }  
}
```

Resultado Python:

```
numero = 42  
print(numero)
```

Ejemplo 2: Estructuras de control

Código Java:

```
public class Main {  
    public static void main(String[] args) {  
        int x = 10;  
        if (x > 5) {  
            System.out.println("Mayor");  
        } else {  
            System.out.println("Menor");  
        }  
    }  
}
```

Resultado Python:

```
x = 10
```

```
if x > 5:
```

```
    print("Mayor")
```

```
else:
```

```
    print("Menor")
```

Ejemplo 3: Bucles

Código Java:

```
public class Main {  
    public static void main(String[] args) {  
        for (int i = 0; i <= 5; i++) {  
            System.out.println(i);  
        }  
    }  
}
```

Resultado Python:

```
for i in range(0, 5 + 1):  
    print(i)
```

Solución de Problemas

✗ "Error al analizar el código"

Problema: El servidor backend no está corriendo

Solución:

1. Verifica que el backend esté ejecutándose en `http://localhost:4000`
2. Revisa la terminal del backend por errores
3. Reinicia el servidor backend con `npm run dev`

✗ "Primero debe generar la traducción"

Problema: Intentas guardar sin haber analizado

Solución: Click en " Analizar" antes de intentar guardar

✗ La página no carga

Problema: El frontend no está corriendo

Solución:

1. Verifica que el frontend esté en `http://localhost:5173`
2. Revisa la terminal del frontend por errores
3. Reinicia con `npm run dev` en la carpeta Frontend