

# Manual de Usuario

## Introducción

Bienvenido al manual de usuario de la aplicación. Este documento le guiará a través de las funcionalidades principales de la aplicación y le mostrará cómo utilizarla de manera efectiva.

## Requisitos del Sistema

- Sistema Operativo: Windows, macOS, Linux
- Graphviz
- ASP.net
- IDE para C# como Visual Studio Code

## Instalación

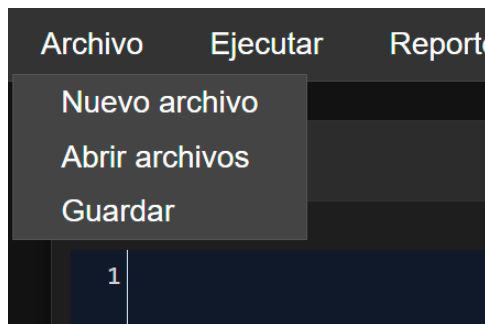
1. Abra su navegador y busque `http://localhost:5192/swagger#`.

## Uso de la Aplicación

### Menú de Operaciones

1. **Cargar Archivo de Prueba**

- Haga clic en el menú Archivo y seleccione abrir archivo.



- Seleccione el archivo de prueba que desea cargar. Recuerde cargar archivos con formato `.glt`

Name	Size	Type	Modified
ilovepdf_converted			18 Feb
Sensores_2			15 Feb
basicas.glt	10.5 kB	Text	Yesterday
funciones.glt	7.1 kB	Text	Yesterday
intermedias.glt	9.0 kB	Text	Yesterday
slices.glt	7.2 kB	Text	Yesterday
structs.glt	8.3 kB	Text	Yesterday

- El contenido del archivo se mostrará en el área de texto.

The screenshot shows a Go IDE with a menu bar (Archivo, Ejecutar, Reportes) and a sidebar. The main editor displays a file named 'Entrada' containing Go code. The code defines a 'main' function that declares variables, prints their values, and checks for correct declarations. The console on the right shows the output of the program, which matches the code's output.

```

1 func main() {
2     puntos := 0
3
4     fmt.Println("=== Archivo de prueba básico ===")
5
6     // 1. Declaración de variables (2 puntos)
7     fmt.Println("==== Declaración de variables ====")
8     puntosDeclaracion := 0
9
10    fmt.Println("Declaración explícita con tipo y valor")
11    var entero int = 42
12    var decimal float64 = 3.14159
13    var texto string = "Hola, mundo!"
14    var booleano bool = true
15
16    fmt.Println("entero:", entero)
17    fmt.Println("decimal:", decimal)
18    fmt.Println("texto:", texto)
19    fmt.Println("booleano:", booleano)
20
21    if entero == 42 && decimal > 3.0 && texto == "Hola, mundo!" && booleano == true {
22        puntosDeclaracion = puntosDeclaracion + 1
23        fmt.Println("OK Declaración explícita con tipo y valor: correcto")
24    } else {
25        fmt.Println("X Declaración explícita con tipo y valor: incorrecto")
26    }
27
28    fmt.Println("\nDeclaración explícita con tipo y sin valor")
29
30 }

```

Consola

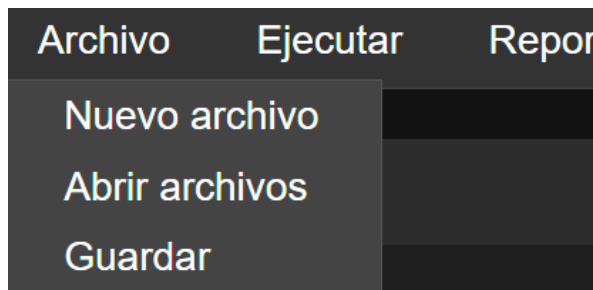
```

1

```

## 2. Guardar Archivo de Prueba

- Haga clic en el menú Archivo y seleccione Guardar.



El contenido del área de texto se guardará en el archivo de prueba.

## Ejecución de Operaciones

```

1 === Archivo de prueba básico ===
2 ==== Declaración de variables ====
3 Declaración explícita con tipo y valor
4 entero: 42
5 decimal: 3.14159
6 texto: Hola, mundo!
7 booleano: true
8 OK Declaración explícita con tipo y valor: correcto
9
10 Declaración explícita con tipo y sin valor
11 enteroSinValor: 0
12 decimalSinValor: 0.0
13 textoSinValor:
14 booleanoSinValor: false
15 OK Declaración explícita con tipo y sin valor: correcto
16
17 Errores de redeclaración
18 OK Errores de redeclaración: correcto
19
20 ==== Asignación de variables ====
21 Asignación con tipo correcto
22 entero: 99
23 decimal: 9.9
24 texto: Texto modificado
25 booleano: false
26 OK Asignación con tipo correcto: correcto
27
28 Asignación con tipo incorrecto
29 OK Asignación con tipo incorrecto: Se detectaron errores de tipo correctamente

```

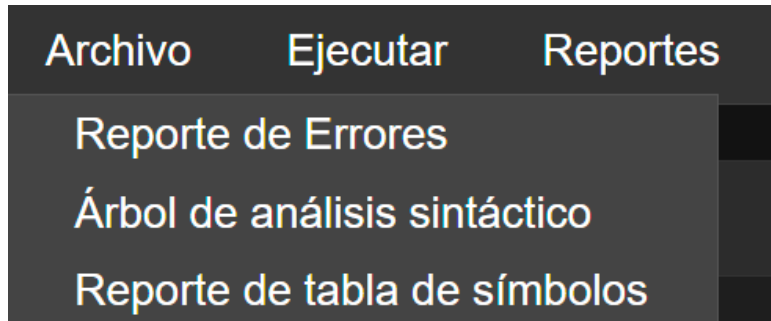
## 1. Obtener Operaciones

- La aplicación leerá y ejecutará el código cargado.
- Los resultados se muestran en consola.

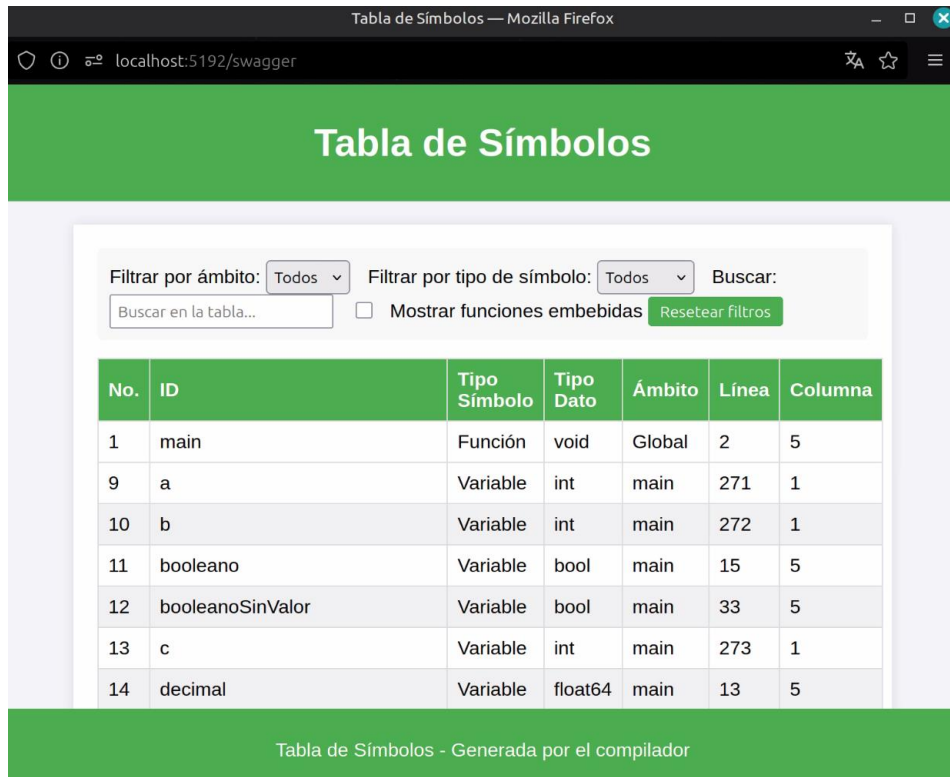
## Visualización de los reportes

### 1. Reporte Tokens

- Seleccione Reportes > Reporte de tabla de símbolos.



- El contenido del archivo se mostrará en una ventana emergente

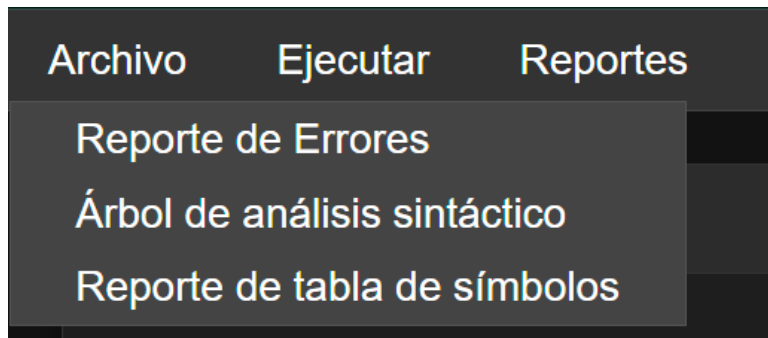


No.	ID	Tipo Símbolo	Tipo Dato	Ámbito	Línea	Columna
1	main	Función	void	Global	2	5
9	a	Variable	int	main	271	1
10	b	Variable	int	main	272	1
11	booleano	Variable	bool	main	15	5
12	booleanoSinValor	Variable	bool	main	33	5
13	c	Variable	int	main	273	1
14	decimal	Variable	float64	main	13	5

Tabla de Símbolos - Generada por el compilador

### 2. Reporte de errores

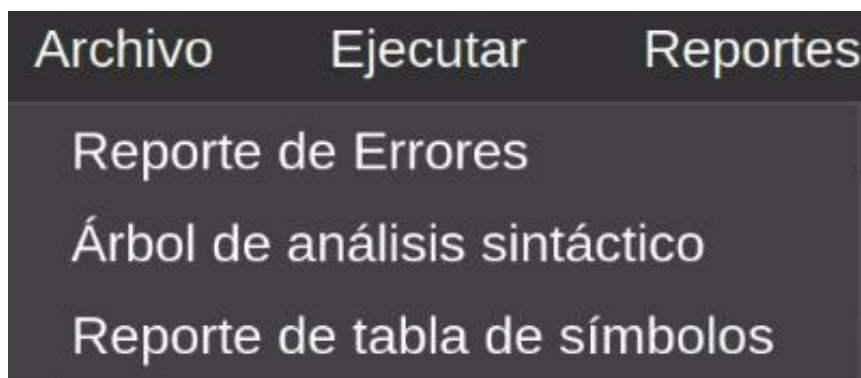
- Seleccione Reportes > Reporte de errores.



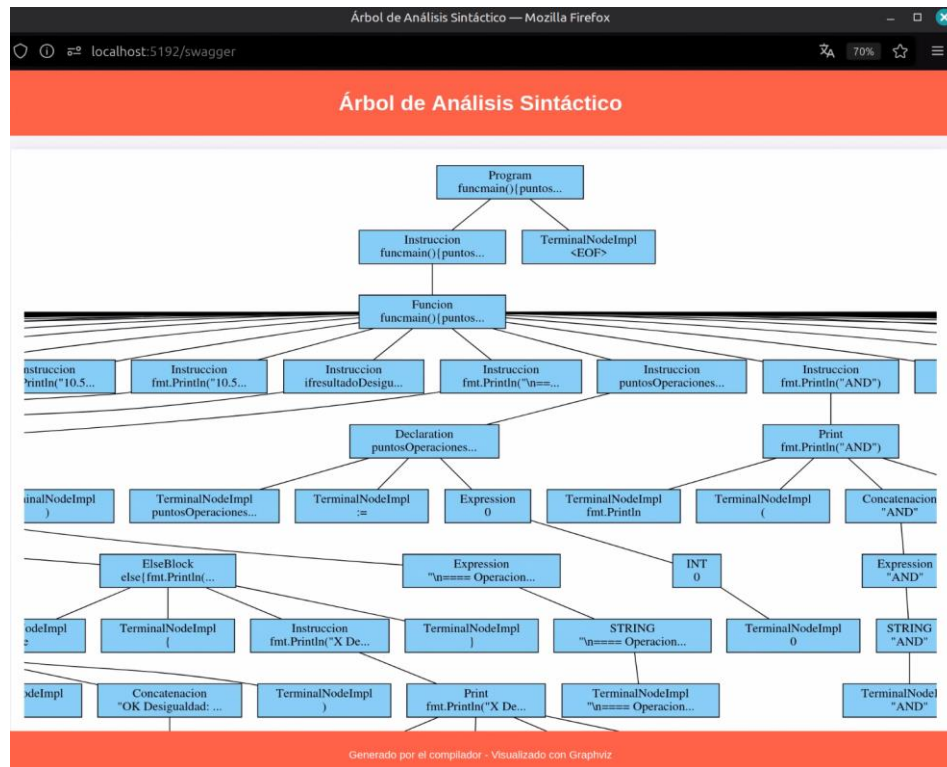
- El contenido del archivo se mostrará en una ventana emergente



- Seleccione Reportes > Árbol de análisis sintáctico



- El contenido del archivo se mostrará en una ventana emergente



- Los resultados de las operaciones se compararán con las evaluaciones esperadas y se mostrarán en la consola.

## Solución de Problemas

- **El archivo de prueba no se carga:**
  - Asegúrese de que el archivo existe y tiene el formato correcto.
  - Verifique que tiene permisos de lectura para el archivo.
- **Las operaciones no se ejecutan correctamente:**
  - Asegúrese de que las operaciones están definidas correctamente en el archivo de prueba.
  - Verifique que el conjunto universal de caracteres está definido correctamente.