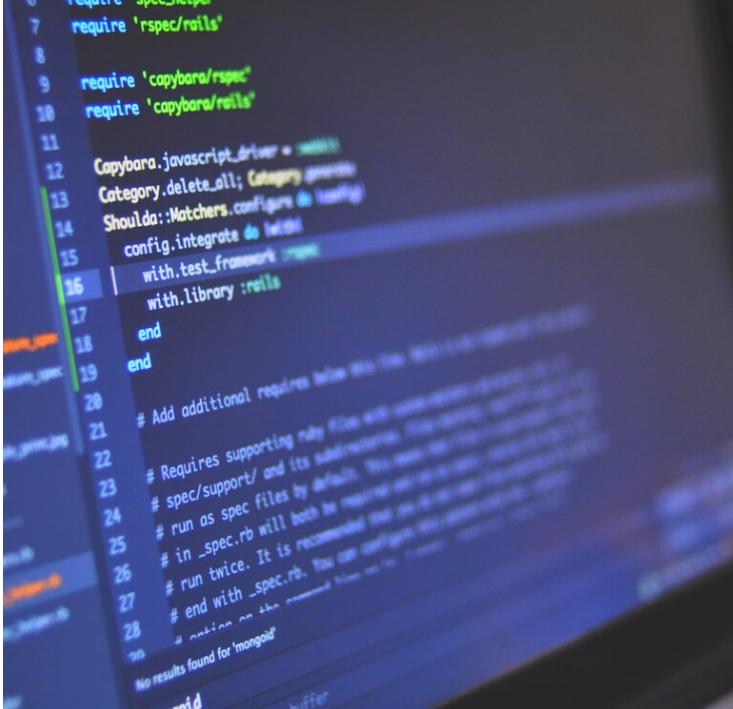
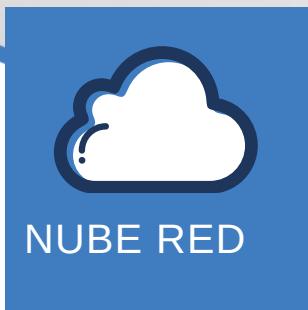


# PROYECTO

```
function(){cards();});  
window).on('resize', function(){cards();});  
function cards(){  
var width = $(window).width();  
if(width < 750){  
cardssmallscreen();  
}  
else{  
cardsbigscreen();  
}  
}  
function cardssmallscreen(){  
cards = $('.card').length;  
i++ = 0;  
cards; i++){  
}
```



# MANUAL SUSARIO



NUBE RED



## SOBRE P1

Aplicacion ideada para la traduccion de lenguaje en pseudocodigo hacia python y golang para facilitar la programacion de personas que aun no esten familiarizadas con los lenguajes

**1** CPU

(Intel Core 2 Duo 2 GHz  
recomendados)  
(intel core inside recomendado)

**2** RAM

(2 GB recomendados)

**3** Memoria

200MB

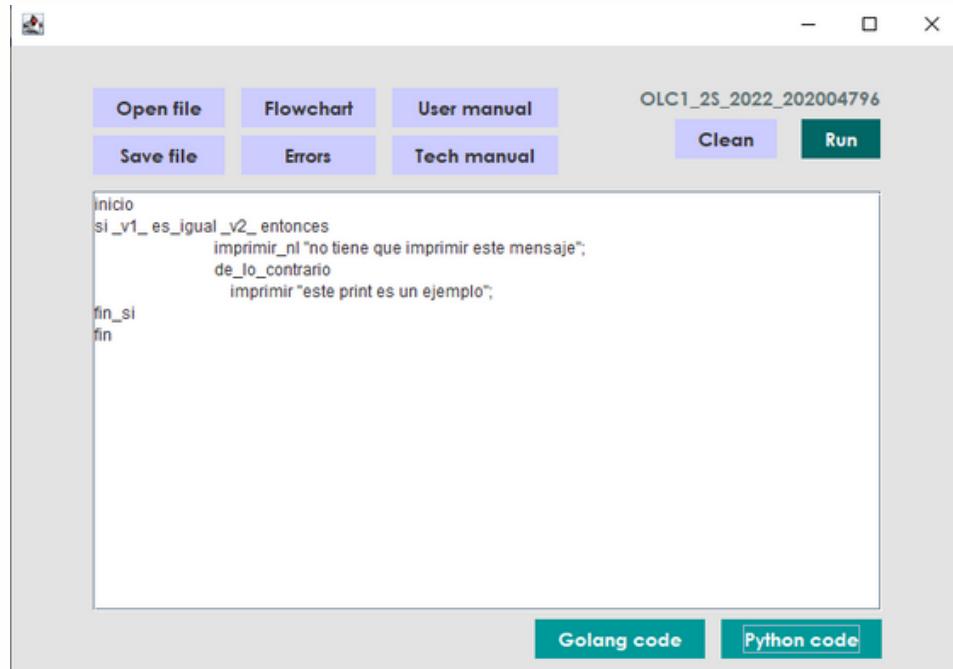
## ¡COMENCEMOS AHORA!

Encontraras el funcionamiento de la aplicación detallada paso por paso

# INICIO

La interfaz es muy amigable para el usuario, haciendola sencilla de usar y haciendo que todas las funcionalidades que esta posee, esten debidamente identificadas para facilitar su uso

**PRECAUCION:** el programa NO distingue entre mayusculas y minusculas por lo que no deberá preocuparse de esto



The screenshot shows a software window titled "OLC1\_2S\_2022\_202004796". The interface includes a toolbar with buttons for "Open file", "Flowchart", "User manual", "Save file", "Errors", "Tech manual", "Clean", and "Run". Below the toolbar is a code editor containing the following pseudocode:

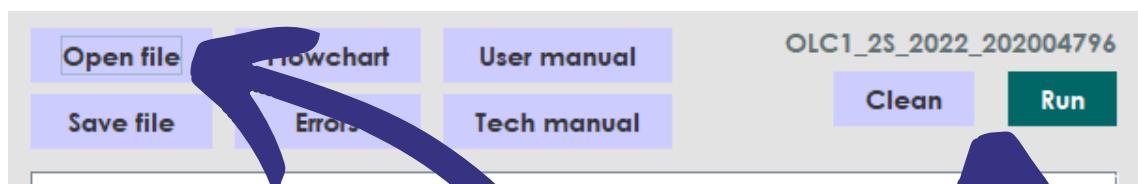
```
inicio
    si _v1_ es igual _v2_ entonces
        imprimir_nl "no tiene que imprimir este mensaje";
        de_lo contrario
            imprimir "este print es un ejemplo";
    fin_si
fin
```

At the bottom of the code editor are two buttons: "Golang code" and "Python code".

# CARGAR AL PROGRAMA

COOn esta opcion se podra seleccionar un archivo de entrada .OLC el cual será analazido y traducido por el programa, este aparatado se mostrará automaticamente en pantalla para su modificación antes de analizar

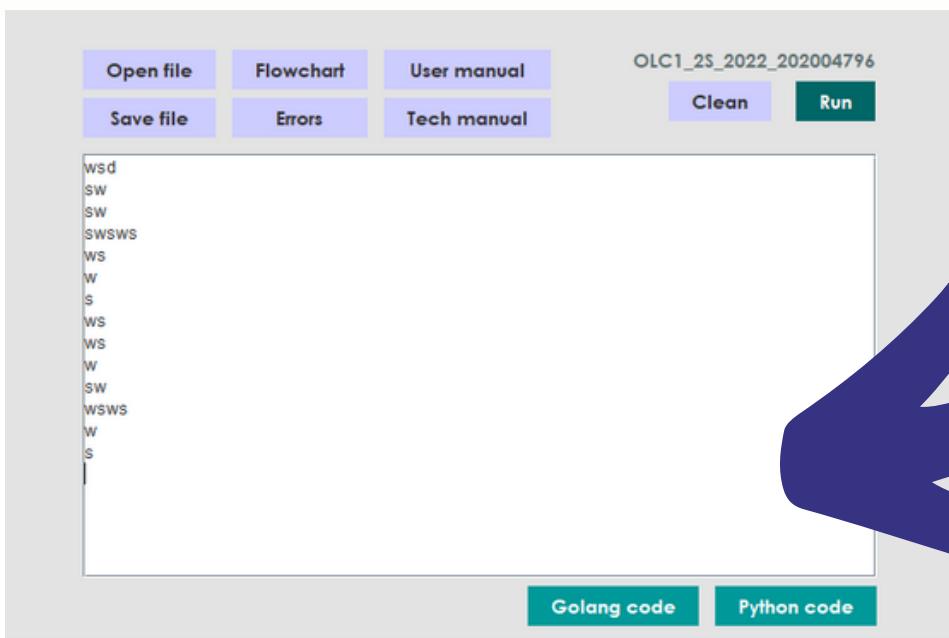
BOTON CLEAN Y RUN: con estos botones se limpiara el texto en pantalla y se analixzar, respectivamente



# RESPUESTA EN PANTALLA

Analizara tu mensaje y mostrara en pantalla una advertencia si no entiende el mensaje, esto significa que el comando ingresado es incorrecto

Se recomienda ser revisado y volver a escribir la instrucción



The screenshot shows a software interface with a central command-line window. The window contains the following text:

```
wsd
SW
SW
SWSWS
WS
W
S
WS
WS
W
SW
WSWS
W
S
```

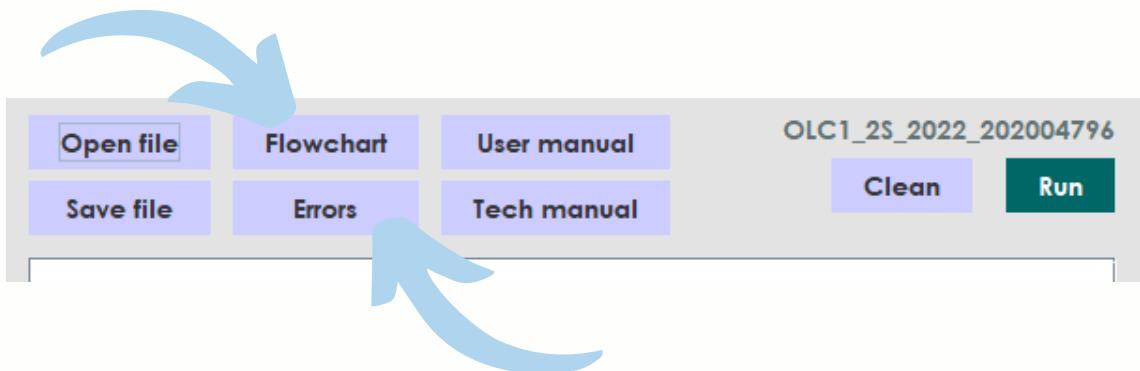
At the top of the interface, there is a navigation bar with the following buttons: Open file, Flowchart, User manual, Save file, Errors, Tech manual, Clean, and Run. The text "OLC1\_2S\_2022\_202004796" is displayed above the Clean button. At the bottom of the interface, there are two buttons: Golang code and Python code.

# REPORTES

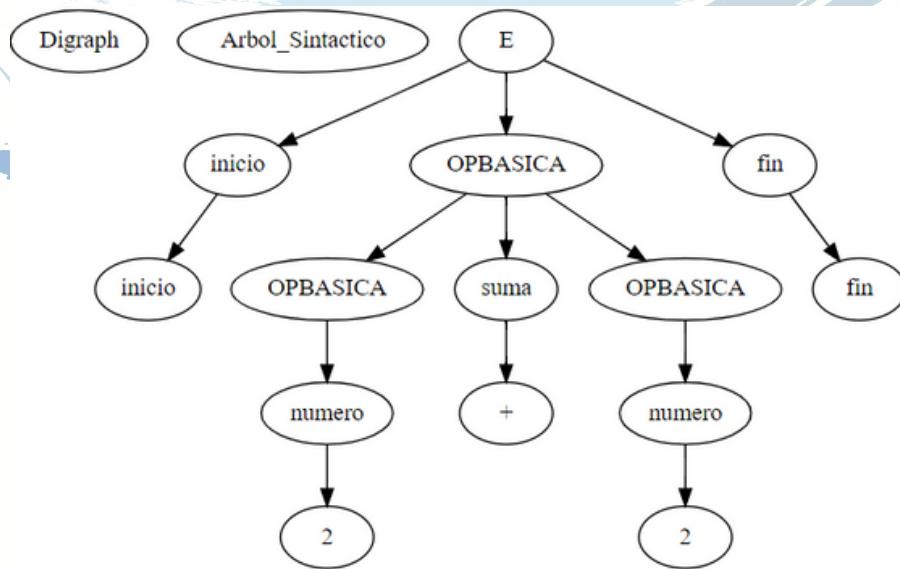
Estos botones generan y abren automáticamente los reportes de errores y el diagrama de arbol que contenía el archivo

**ATENCION: LA INFORAMCION ANALIZADA ES LA PERTENECIENTE AL AREA DE TEXTO MODIFICABLE, ASI QUE LOS CAMBIOS SE VERAN REFLEJADOS**

**SI DETECTA ERRORES UNICAMENTE SE CREARA EL REPORTE DE ERRORES PERO NO EL AST**



# EJEMPLO DE REPORTES



## Lista de Errores

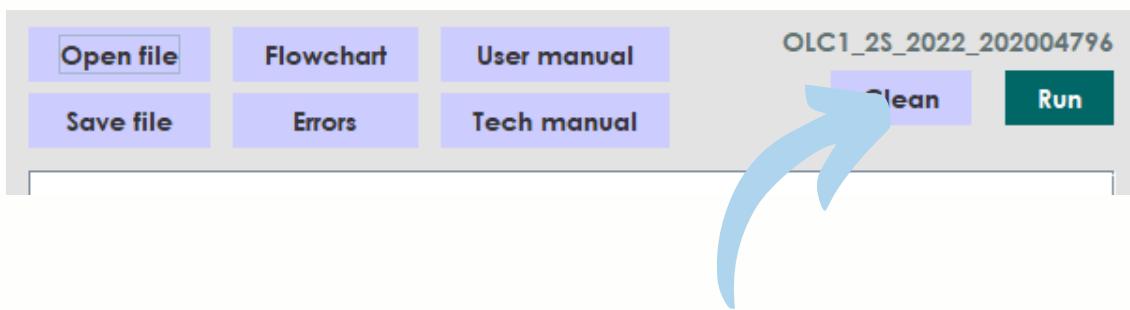
Creado por: Gerhard Benjamin Ardon Valdez 202004796

Tipo	Lexema	Línea
SINTACTICO	VS	1

# LIMPIAR LOG DE TOKENS Y ERRORES

Estos botones eliminan automaticamente el registro de tokens o de errores, luego de pulsar el boton se mostrara una lista vacía

**ATENCION: LA INFORAMCION ANALIZADA ES LA PERTENECIENTE AL AREA DE TEXTO MODIFICABLE, ASI QUE LOS CAMBIOS SE VERAN REFLEJADOS**



# ENSEÑAR MANUALES

En estos botones se enseñaran los manuales ambos, tecnico y de usuario para su correcta compresion



# EJEMPLO DE ESCRITURA ARCHIVO OLC

A continuacion se muestra como deberá ir escrito el archivo OLC para el correctofuncionamiento, este archivo puede cargarse en cualquier momento

```
imprimir_nl _encabezado1_;
imprimir_nl _encabezado2_;
imprimir "...";
imprimir _anio1_ ;
imprimir _anio2_ ;
imprimir _anio3_ ;
imprimir _anio4_ ;
imprimir_nl ".";
imprimir_nl (_v3_);

si _v1_ es igual _v2_ entonces
|   imprimir_nl "Al parecer no funciona la asignacion";
|   mientras not (_variable1_ mayor_o igual 5*5+8/2) hacer
|   |   imprimir _variable1_;
|   |   _variable1_ -> _variable1_ + 1;
|   fin_mientras
fin_si

si _v1_ es igual _v2_ entonces
|   imprimir_nl "no tiene que imprimir este mensaje";
|   de_lo_contrario
|   |   imprimir "este print es un ejemplo";
fin_si
```