

# Manual Tecnico



UNIVERSIDAD DE SAN CARLOS DE GUATEMALA  
*LABORATORIO OLC 1*

*Steven Josue González  
Monroy  
201903974*

## PyGO-Code

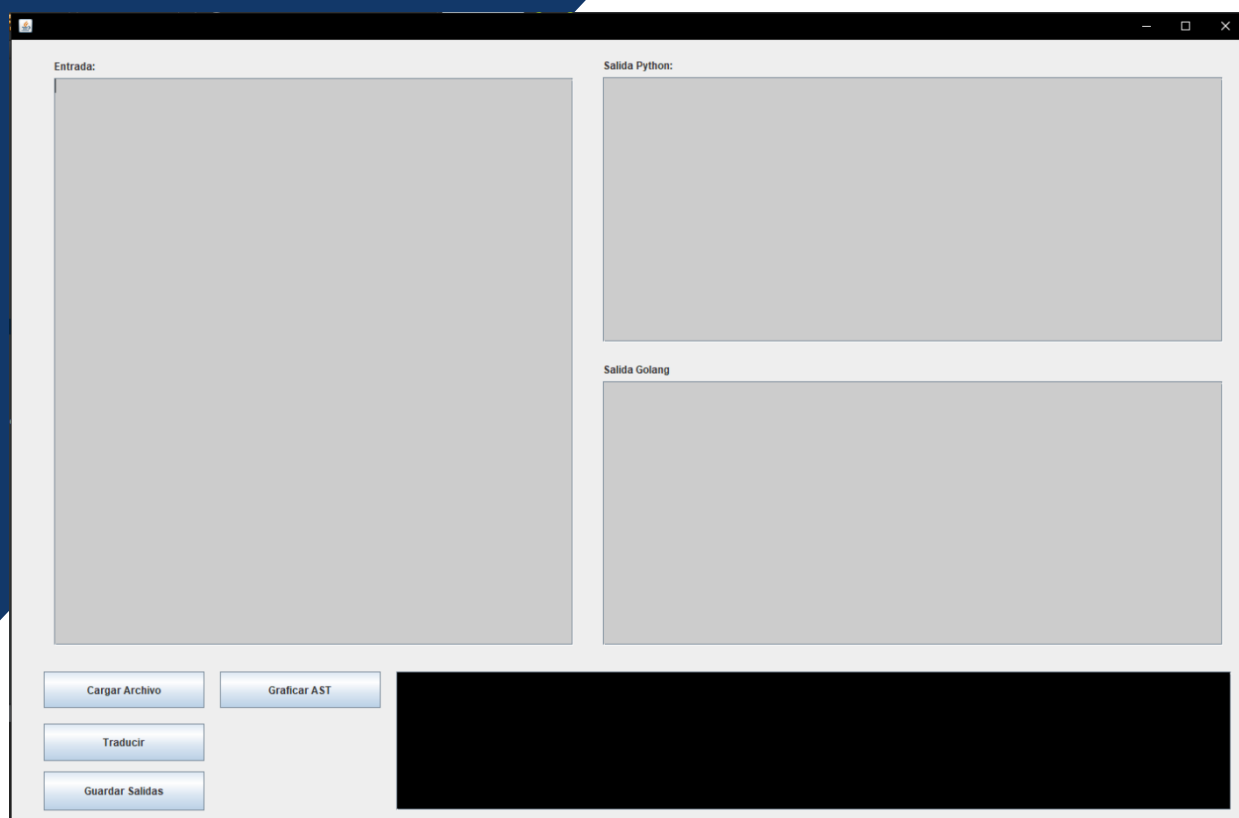
*PYGO-Code es un analizador de pseudocodigo con el cual traducir las instrucciones que se te ocurran a los lenguajes Python y Go. A su vez este programa permite crear graficos de Arboles AST correspondiente a las instrucciones del pseudocodigo que reciba como entrada.*

El programa fue elaborado en Java y apoyandose del uso de herramientas de creacion de analisis lexico y sintactico , en este documento dirigido a tecnocos o desarrolladores, se encuentra toda la informacion de las Herramientas y funciones utilizadas para la elaboracion de la aplicación.

### *Requerimientos:*

- *JDK (Proyecto elaborado en jdk17)*
- *Librerias jflex y Cup(Disponibles en el Repositorio)*
- *Archivos de entrada*
- *Documento de requerimiento*
- *Ide o editor de codigo para Java(Proyecto realizado en Netbeans 12.6)*

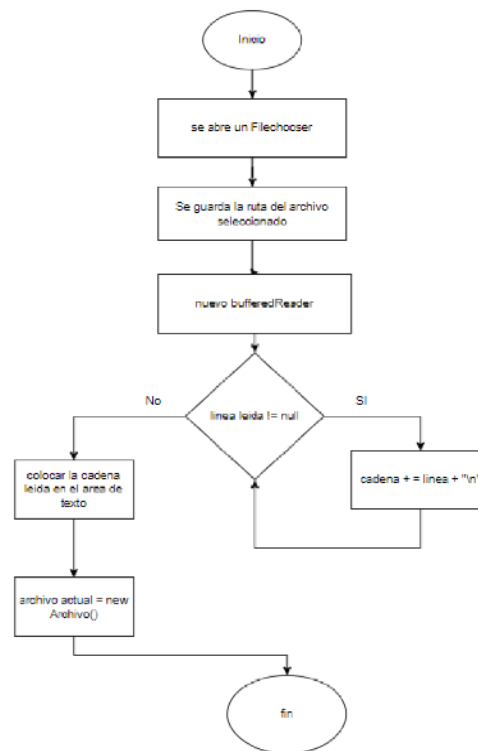
# Intefaz



La ventana con todos los componentes en ella fueron realizados en el IDE NetBeans 12.6 con la herramienta drag and drop a continuación se adjunta una imagen del constructor con todos los componentes de la librería swing utilizados:

```
1 private javax.swing.JTextArea Consola;
2 private javax.swing.JButton botonAnalizar;
3 private javax.swing.JButton botonCargar;
4 private javax.swing.JButton botonGraficar;
5 private javax.swing.JButton botonGuardar;
6 private javax.swing.JLabel jLabel1;
7 private javax.swing.JLabel jLabel2;
8 private javax.swing.JLabel jLabel3;
9 private javax.swing.JScrollPane jScrollPane1;
10 private javax.swing.JScrollPane jScrollPane5;
11 private javax.swing.JScrollPane jScrollPane6;
12 private javax.swing.JScrollPane jScrollPane7;
13 private java.awt.Panel panel1;
14 private java.awt.Panel panel3;
15 private javax.swing.JTextArea textArea_Entrada;
16 private javax.swing.JTextArea textArea_Go;
17 private javax.swing.JTextArea textArea_Python;
18 // End of variables declaration//GEN-END:variables
19 public Utils.Archivo actual;
20 public static listaErrores errores;
21 public Arbol arbol;
22 public LinkedList<Instruccion> instrucciones;
23 public String traduccionPython, traduccionGolang;
```

# Apertura de Archivos



```
1 private void botonCargarActionPerformed(java.awt.event.ActionEvent evt) {GEN-FIRST:event_botonCargarActionPerformed
2     Principal referencia = this;
3     try {
4         // TODO add your handling code here:
5         JFileChooser fileChooser = new JFileChooser();
6         fileChooser.showOpenDialog(referencia);
7         String pat = fileChooser.getSelectedFile().getPath();
8         File doc = new File(pat);
9
10        try (BufferedReader obj = new BufferedReader(new FileReader(doc))) {
11            String strng, texto = "";
12            while ((strng = obj.readLine()) != null) {
13                texto = texto + strng + "\n";
14            }
15            textArea_Entrada.setText(texto);
16            String nombre = doc.getName();
17            actual = new Utils.Archivo(pat, nombre);
18            obj.close();
19        }
20    } catch (IOException ex) {
21        Logger.getLogger(Principal.class.getName()).log(Level.SEVERE, null, ex);
22    }
23 }
24 }GEN-LAST:event_botonCargarActionPerformed
```

# Flujo de ejecucion

