



STATPY CONVERTER

MANUAL TECNICO



● Versiones de las herramientas utilizadas

- JDK 19
- Java CUP runtime 11
- Jflex 1.7
- Jcommon 1.0.23
- JFreeChart 1.0.19
- APACHE Netbeans IDE 18



● Especificaciones de entorno

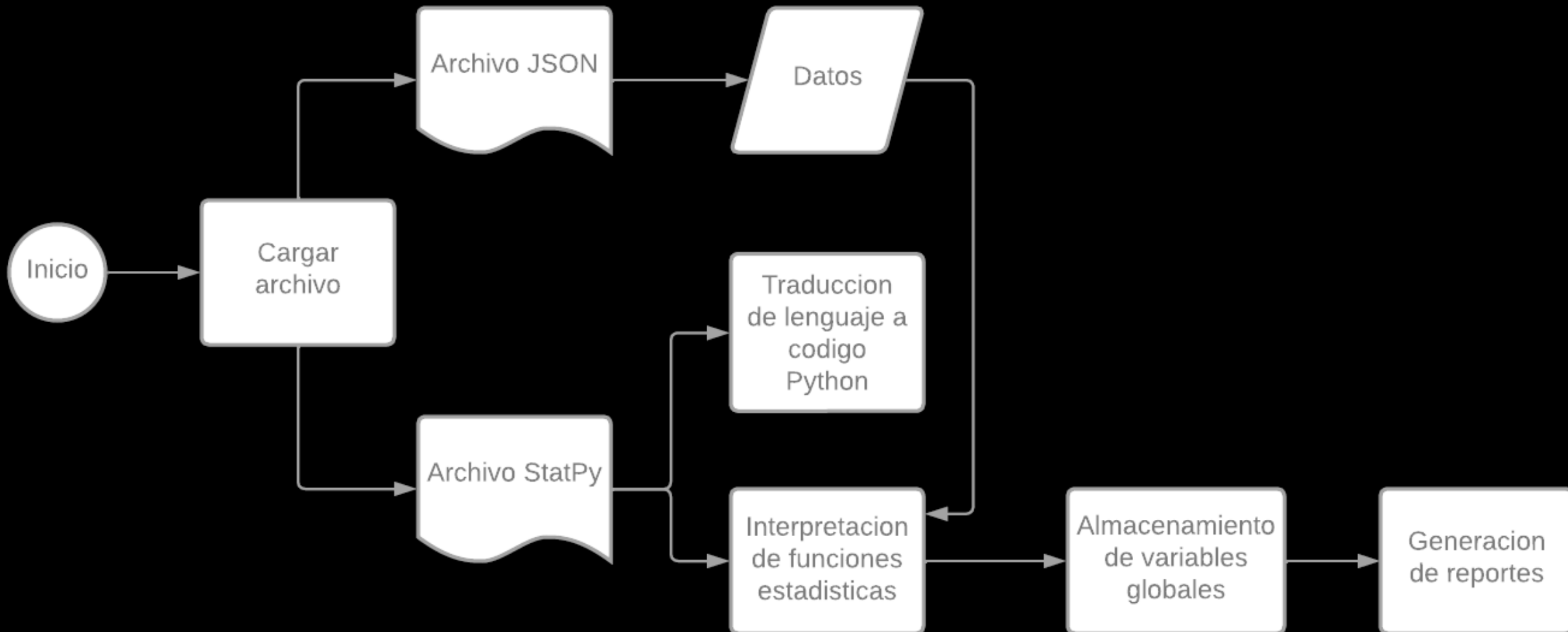
El programa fue desarrollado en el Sistema operativo Windows 11, utilizando el IDE apache Netbeans 18 creando una aplicacion con Java ANT.

Para ejecutar el programa en otros sistemas operativos es necesario contar con la JVM(Java Virtual Machine).

Todos los recursos fueron adquiridos en su archive JAR.



- Diagrama de flujo del programa



● Interfaz Grafica

The image shows a graphical user interface (GUI) for a file analysis tool. The interface is divided into several sections:

- Top Bar (Red):** Contains a dropdown menu labeled "ABRIR ARCHIVO" and three buttons: "ERRORES", "TOKENS", and "GRAFICOS".
- Second Bar (Dark Blue):** Contains a button labeled "CAMBIAR ANALIZADOR" and a label "ANALIZADOR: STATPY".
- Third Bar (Dark Blue):** Contains a button labeled "LIMPIAR TODO".
- Main Area:** Divided into two large panels:
 - ENTRADA:** A large dark blue rectangular area for input.
 - SALIDA:** A large dark blue rectangular area for output.
- Bottom Bar (Green):** Contains a large button labeled "EJECUTAR".



Descripcion de la interfaz



BARRA DE NAVEGACION

- abrir un archivo
 - actualizar un archivo
 - guardar un nuevo archivo
-
- Abrir el reporte de errores
 - Abrir el reporte de tokens
-
- Visualizar los graficos generados

● Editor y terminal

Con el boton “cambiar analizador” variamos entre StatPy y JSON segun el archive que deseemos analizar

CAMBIAR ANALIZADOR

ANALIZADOR: STATPY

LIMPIAR TODO

ENTRADA:

SALIDA:

EJECUTAR





Se cargan los archivos al editor donde se pueden modificar, y al estar listo el archivo se procede a “ejecutar”

StatPy Converter

ABRIR ARCHIVO

ERRORES

TOKENS

GRAFICOS

CAMBIAR ANALIZADOR

ANALIZADOR: STATPY

LIMPIAR TODO

ENTRADA: EJEMPLO.SP

```
// Ya salio compil :)

void main ( ){
    int b = 2;
    int a =1;
    int var1 = 5+8*9;
    if (b > a){
        Console.Write("b mayor que a");
    }else if(a == b){
        Console.Write("a y b son iguales");
    }
    void DefinirGlobales(){
        string reporte1 = "Reporte 1";
        double pe1 = 0.8;
        double pe2 = 0.5;
        double pe3 = 0.2;
        double po1 = ${ NewValor, "ejemploJSON.json", "variable1"};
        double po2 = ${ NewValor, "pruebas.json", "variable1"};
        string var1 = "Valor Obtenido";
        string var2 = "Valor Esperado clase 1";
        string var22 = "Valor Obtenido clase 1";
        string var3 = "Valore Esperado clase 2";
        string var33 = "Valor Obtenido clase 2";
    }
    void GraficaBarras(){
        string Titulo= reporte1;
```

SALIDA:

```
def main():

    b=2

    a=1

    var1=5+8*9

    if b > a:

        print("b mayor que a")

    elif a == b:

        print("a y b son iguales")
```

EJECUTAR

● Explicacion del Analisis

- Los archivos JSON son para obtener datos que despues podemos llamar para guardarlos.
- Los archivos StatPy tienen el proposito de brindar elementos para traducirlos a lenguaje Python y presentar 3 funciones especiales
 - DefinirGlobales(): se definen variables que podremos reutilizar.
 - GraficoBarras(): genera una imagen de un grafico de barras.
 - GraficoPie(): genera una imagen de un grafico de Pie.



● Explicacion de la estructura en Java

- El Proyecto cuenta con 4 paquetes
 - Analizador StatPy
Cuenta con los archivos de la gramatica StatPy generada con el formato BNF.
 - Analizador JSON
Cuenta con los archivos de la gramatica JSON generada con el formato BNF.
 - Reportes
Cuenta con los reports generados.
 - StatPyConverter
Cuenta con las clases objetos y configuraciones necesarias para unir todas las partes y lograr que el programa funcione.

