MANUAL TÉCNICO

Proyecto 1 | Lenguajes Formales y de Programación

Estudiante: Carlos Alfredo López de León
Carnet: 3355697810901

Índice

Introducción

El presente manual muestra las herramientas y la lógica utilizada para desarrollar una solución de software para la empresa “Grafos Guatemala“, la cual se dedica a la elaboración de grafos a través de un archivo de texto de entrada.

La aplicación tiene como objetivo procesar la información de un archivo de texto con extensión “.code“ para generar una imágen con los grafos descritos. El análisis del archivo consiste en el análisis léxico del mismo, separando los tokens y mostrando los errores contenidos, utilizando un autómata finito determinista (AFD).

Objetivos

General

- Proporcionar al lector una explicación de la lógica, clases y atributos implementados en el desarrollo de la aplicación programada para facilitar el mantenimiento y futuras modificaciones realizadas por terceros.

Específicos

- Exponer una descripción del SO, IDE y otros elementos utilizados durante el desarrollo de la aplicación.
- Otorgar al lector una explicación técnica de los métodos y atributos que componen la parte operativa de la aplicación.
- Proporcionar el conocimiento necesario para el mantenimiento sencillo del software.

Entorno de desarrollo

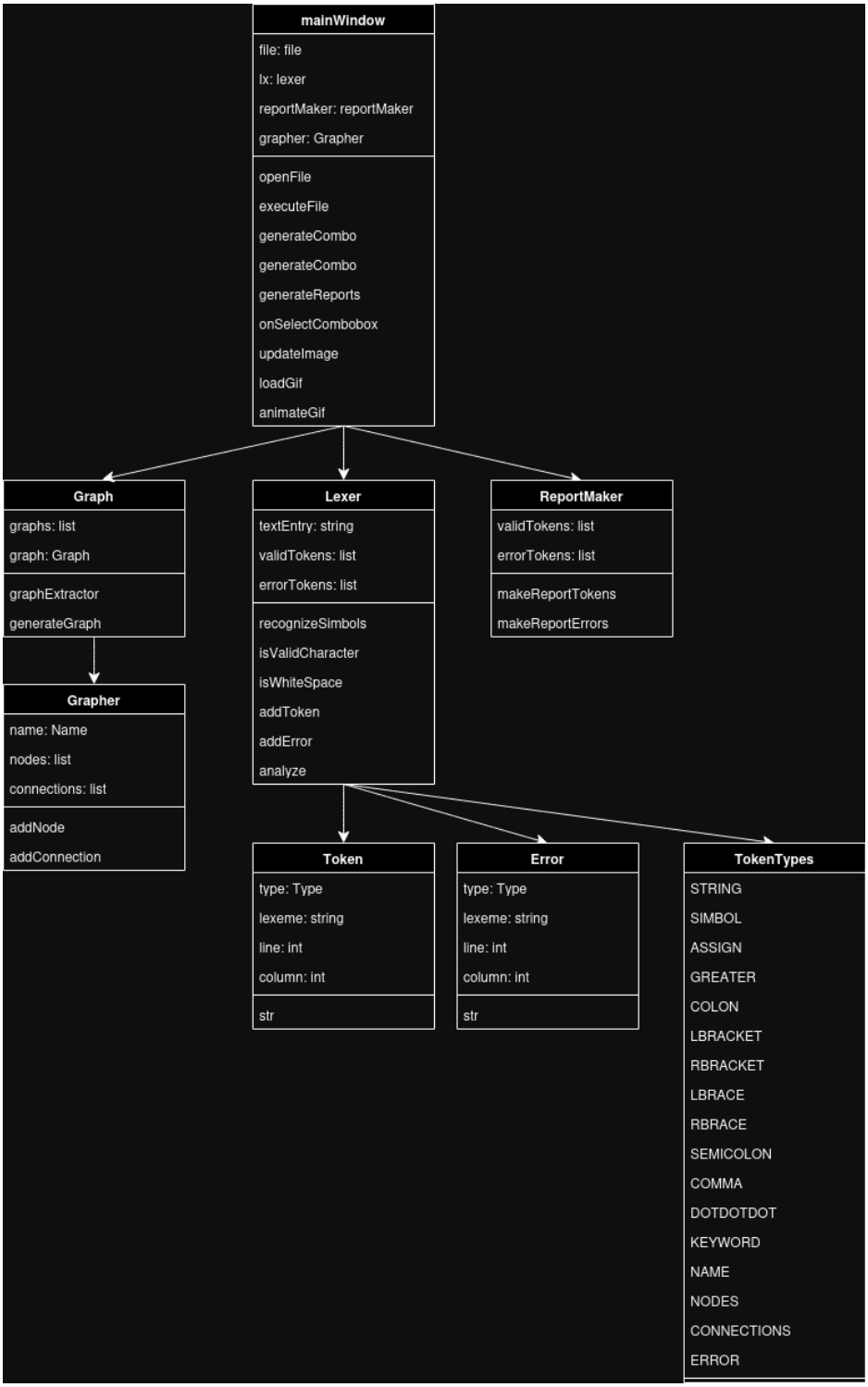
En el desarrollo de la aplicación fue utilizado el IDE PyCharm, debido a ser un entorno de trabajo completo y que nos proporciona una gran cantidad de funcionalidades, entre ellas la herramienta de Copilot, la cual agiliza el trabajo de desarrollo.

Lenguaje de programación:
Python 3.12.3

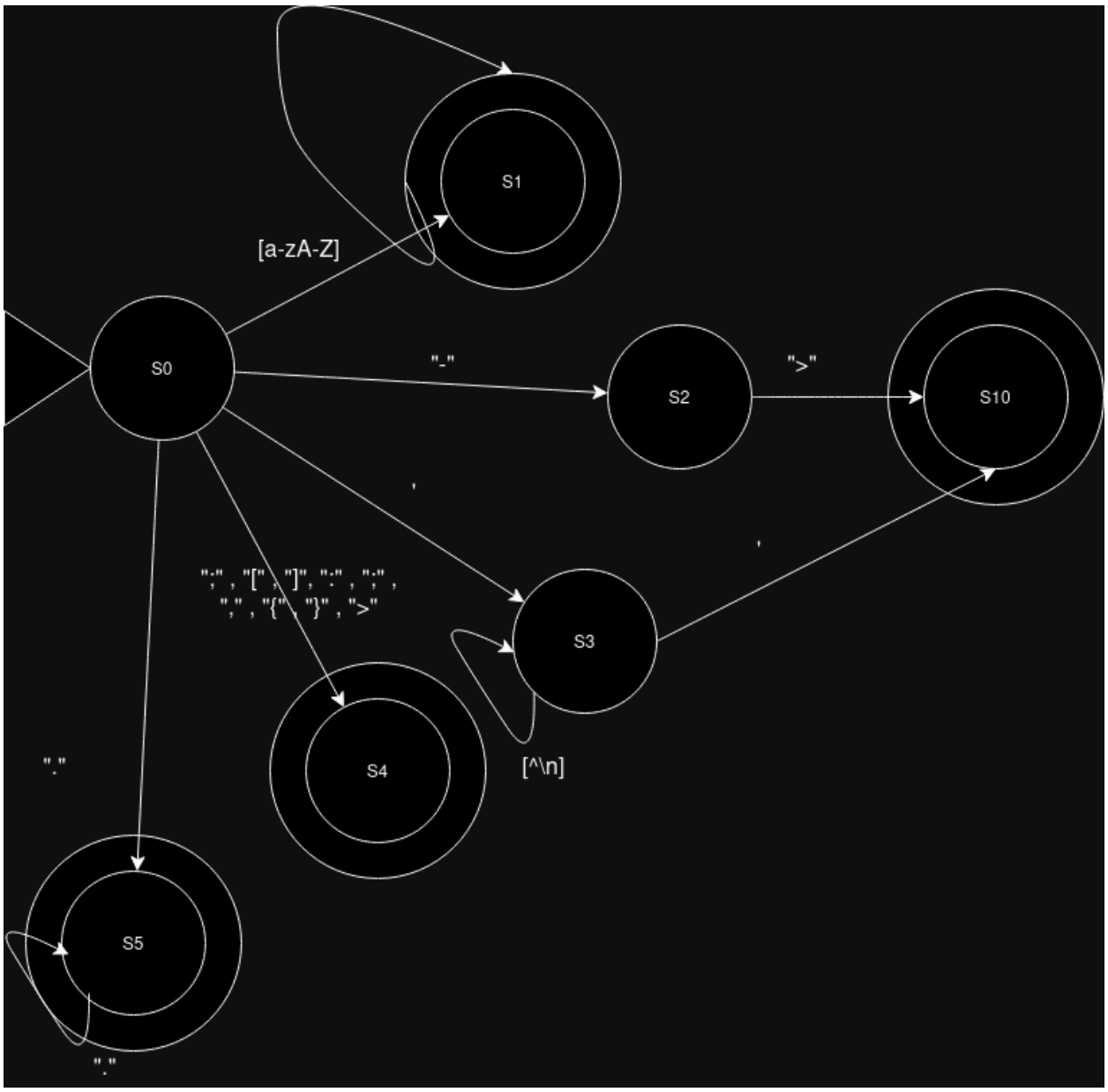
Sistema operativo:
Ubuntu 24.04 LTS

Arquitectura del proyecto

Diagrama de clases



Autómata finito determinista



Librerías utilizadas

Las librerías utilizadas en el proyecto son las siguientes:

- tkinter | Para la creación de la interfaz gráfica
- pillow | Para el manejo de imágenes
- graphviz | Para el manejo de grafos

Descripción de la solución

Para poder desarrollar este proyecto, se analizó lo solicitado por el cliente, sus restricciones, ambiente y forma de trabajo.

Entre estas, las principales son:

- Al ingresar un archivo, este debe ser exclusivamente con extensión ".code", donde se analizará el texto con un lexer y si no hay errores se procederá a la generación de los grafos contenidos.
- En caso de que un archivo contenga errores, entonces se le notificará al usuario y los grafos no serán generados
- Al finalizar el análisis, se crearán los reportes de los tokens analizados, tanto válidos como errores. estos en archivos diferentes de tipo "html"
- El grafo generado se mostrará en un cuadro dentro de la aplicación.

Lógica de la solución

La lógica utilizada para el desarrollo de la aplicación es la siguiente:

El usuario carga un archivo de entrada, con extensión ".code" y este es analizado por el AFD con el fin de identificar

1. Tokens
2. Errores léxicos

Luego, esta lista de tokens es enviada al graficador el cuál tiene el objetivo de identificar los atributos y separar individualmente cada grafo identificador. Este también tiene la función de generar el grafo seleccionado.

Solo los archivos que no contengan errores podrán ser convertidos en grafos, de lo contrario será notificado al usuario.

Los reportes serán generados en archivos de tipo "html", donde uno contendrá los tokens válidos y otro contendrá los errores.

Al cargar un nuevo archivo, cada opción se limpia y queda con la nueva información ingresada.