Universidad de San Carlos de Guatemala

Facultad de Ingeniería

Escuela de Ciencias y Sistemas

Organización de Lenguajes y Compiladores 1

Practica 1

Manual Técnico

**Nombre: Duglas Francisco Avila Torres**

**Carnet: 201314778**

* Índice

Índice……………………………………………………………………………………………………………2

Introducción.………………………..………………………………………………………………………3

Objetivos y alcance del manual ……………………………………………………………………4

Descripción general de la aplicación……………….……………………………………………5

Diagrama de clases……………………………………….………………………………………………6

Definición de clases, funciones, procedimientos……………………………………………7

Librerías Utilizadas……………………………………………………………………………………...11

* Introducción

El siguiente manual hace una descripción de la aplicación, un miniIDE destinado a realizar el análisis léxico y sintáctico de un texto utilizando la librería JFlex y JavaCup. Esto servirá para que la aplicación reconozca un lenguaje y realice distintas operaciones. A continuación se describirá de manera más detallada la funcionalidad del sistema, como está estructurado, las clases y métodos que se implementaron para que el programa cumpla con su función.

* Objetivos y alcance del manual

-Que se refleje la estructura con la cual se creó el sistema.

-Orientar sobre los métodos que se implementaron en el sistema.

-Que se pueda conocer el funcionamiento del programa.

* Descripción general de la Aplicación

El software realiza la tarea de analizar un texto, el cual contendrá letras, símbolos y números, sus funciones abarcan diferenciar cada lexema y verificar si el texto contiene errores léxicos y/o sintácticos. Las funciones principales son las siguientes:

1. Abrir y Guardar archivos.

2) Copiar, Cortar y Pegar.

3) Analizar texto y mostrar una tabla de símbolos.

4) Analizar en busca de errores léxicos, en caso de existir errores se muestra una tabla que dichos errores y no se muestra la tabla símbolos.

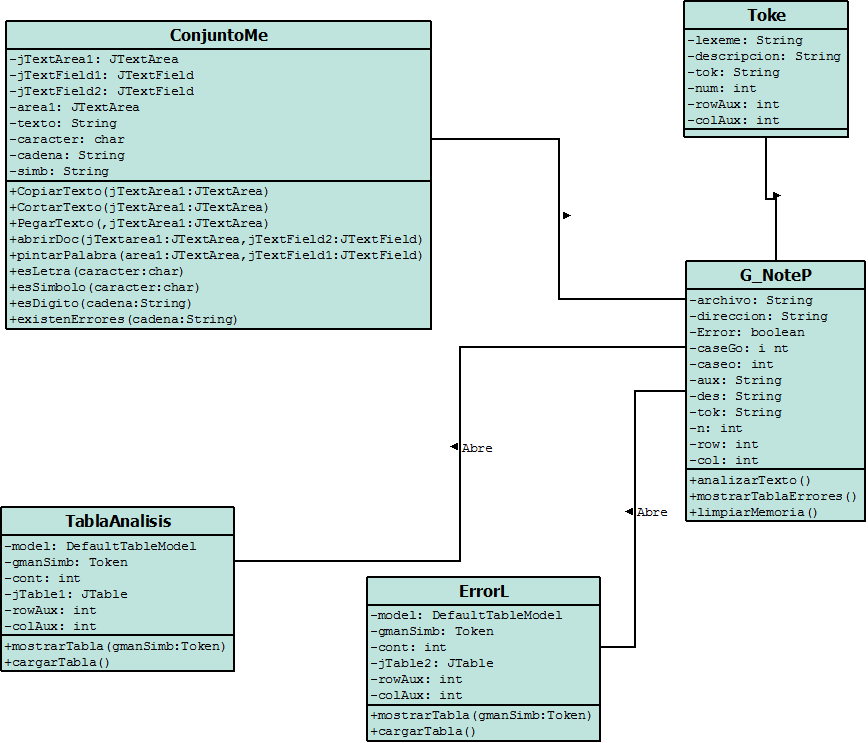
5) Indentar el texto y dar formato.

La aplicación tendrá las siguientes clases:

* Clase: PanelInOut
* Clase: WPrincipal
* Clase: Reporte
* Clase: ReporteE
* Clase: Yytoken

A continuación se definirá de manera detallada las clases y procesos que contendrá la aplicación.

* Diagrama de Clases

****

ReporteE

Reporte

WPrincipal

PanelInOut

Yytoken

* Definición de clases, funciones, procedimientos
* **Clase PanelInOut**

Esta clase está destinada a almacenar todos los métodos que vayan a utilizarse para el analizador.

* **Clase WPrincipal**

Esta clase está destinada a mostrar una interfaz gráfica con las opciones que puedan realizarse.

* **Clase sym**

La clase sym guarda un código de cada token.

* **Clase Reporte**

La clase Reporte está destinada a mostrar una tabla con todos los elementos que se hayan analizado anteriormente.

* **Clase ReporteE**

La clase ReporteE está destinada a mostrar una tabla con los errores que pueda tener el texto. Solo se mostrará si hay errores.

* Descripción de Atributos
* **Clase PanelInOut**

|  |  |
| --- | --- |
| -area\_in | Area de texto de entrada. |
| -area\_out: | Area de texto de salida. |
| simb | Parámetro del método existenErrores. |

* **Clase W\_Principal**

|  |  |
| --- | --- |
| -Error: | Variable boolean para notificar error. |
| -caseGo: | Estado que toma cada case dentro del switch del método para analizar texto. |
| -caseo: | Estado que toma cada case dentro del switch del método para verificar errores. |
| -aux: | Variable utilizada en el método para analizar texto y errores, toma el valor del siguiente carácter analizado, su último valor es el lexema. |
| -des: | Variable descripción del lexema. |
| -tok: | Variable número de tipo token del lexema |
| -n: | Variable número de cada token. |
| -row: | Variable fila. |
| -col: | Variable columna. |

* **Clases Reporte**

|  |  |
| --- | --- |
| -documento: | Variable Document. |
| -gmanSimb: | Variable tipo Token. |
| -cont: | Contador utilizado en el recorrido para llenar la tabla. |
| -ficheroPdf: | Variable tipo FileOutputStream. |

* **Clase ReporteE**

|  |  |
| --- | --- |
| -documento | Variable Document. |
| -gmanSimb: | Variable tipo Token. |
| -cont: | Contador utilizado en el recorrido para llenar la tabla. |
| -ficheroPdf: | Variable tipo FileOutputStream. |

* Descripción de Métodos
* **Clase PanelInOut**

|  |  |
| --- | --- |
| +CopiarTexto(): | Esté método se utilizará para copiar texto del TextArea. |
| +CortarTexto(): | Método utilizado para cortar texto del TextArea. |
| +PegarTexto(): | Método utilizado para pegar texto del TextArea. |
| +abrirDoc(); | Este método abre un documento guardado con la extensión. prac1lfp. |
| +esDigito(); | Método utilizado para verificar que un carácter analizado sea número. |
| +existenErrores() | Método utilizado para verificar que un carácter analizado sea un símbolo considerado error. |

* **Clase WPrincipal**

|  |  |
| --- | --- |
| +analizarTexto(): | Método que se utiliza para analizar el texto dentro del textarea. |
| +limpiarMemoria(); | Método que limpia las variables. |

* **Clase Reporte**

|  |  |
| --- | --- |
| +mostrarTabla(): | Con este método es posible agregar a la tabla de tokens analizados en caso de no mostrar errores. |
| +cargarTabla(); | Método que se utiliza para cargar el la tabla. |

* **Clase ReporteE**

|  |  |
| --- | --- |
| +mostrarTabla(): | Con este método es posible agregar a la tabla los errores encontrados en el texto. |
| +cargarTabla(); | Método que se utiliza para cargar la tabla. |

Librerías utilizadas

java.awt.Color;

java.io.BufferedReader;

java.io.File;

java.io.FileReader;

java.util.logging.Level;

java.util.logging.Logger;

java.util.regex.Matcher;

java.util.regex.Pattern;

javax.swing.JFileChooser;

javax.swing.JFrame;

javax.swing.JOptionPane;

javax.swing.JTextArea;

javax.swing.JTextField;

javax.swing.filechooser.FileNameExtensionFilter;

javax.swing.text.BadLocationException;

javax.swing.text.Caret;

javax.swing.text.DefaultHighlighter;

javax.swing.text.Highlighter;

java.io.BufferedWriter;

java.io.File;

java.io.FileWriter;

java.io.IOException;

java.util.LinkedList;

javax.swing.JFileChooser;

javax.swing.JOptionPane;

javax.swing.text.Caret;

javax.swing.table.DefaultTableModel

java.nio.file.Path;

java.nio.file.Paths;

java.util.logging.Level;

java.util.logging.Logger;