Universidad De San Carlos de Guatemala Facultad de Ingeniería

Escuela de Ciencias y Sistemas

**Lenguajes Formales y de Programación Sección -B**

MANUAL TÉCNICO

Kevin Gerardo Ruíz Yupe

201903791

28/10/2021

# Objetivos

### General:

Enfatizar en el uso de clases, variables y métodos escritos en la aplicación para usuarios con conocimiento específico en el tema. Asimismo, para edición de código en futuras actualizaciones.

### Específicos:

* Mostrar el IDE utilizado para el desarrollo del programa
* Describir el proceso de solución, así como las herramientas utilizadas durante el proyecto.

# Introducción

Este manual pretende explicar y describir los temas que se desarrollan durante el proceso de creación del programa. De la misma forma, se indica los requerimientos del sistema, el IDE utilizado y las librerías que se importaron.

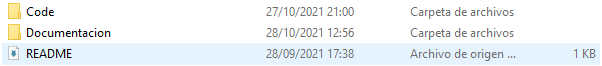
La principal función de esta aplicación es la manejo de datos a partir de un archivo con extensión .lfp, recorriendo cada carácter utilizando ciclos y condicionales. Para verificar que el archivo no contenga errores léxicos y sintácticos se hace uso de expresiones regulares y gramáticas.

# Descripción de la solución

Sea crea una carpeta para el proyecto, esta carpeta contendrá los archivos .py para las clases utilizadas. Existen subcarpetas que guardan imágenes como apoyo adicional al programa. Asimismo, se tienen otras subcarpetas para los archivos css y html.



Carpetas para la documentación y el código fuente



Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

Existen 3 clases para el desarrollo del proyecto:

Patrón de fondo

Descripción generada automáticamente

Texto

Descripción generada automáticamenteComo Clase Principal se tiene la Clase de nombre ***Application***, se muestra a continuación el contenido:

Esta clase inicializa la ventana principal y configura el tamaño y su posición.

Texto

Descripción generada automáticamente

Método **create\_widgtes**



Este método inicializa cada componente de la aplicación y configura el tamaño de los widgets, del mismo modo su posición.

Texto

Descripción generada automáticamente

Método **select\_file**

En este método se define el cuadro de diálogo para seleccionar el archivo .lfp.

Se tiene un ***Try*** para verificar que el archivo exista o si no se seleccionó ningún archivo. Si no cumple, se desplegará un ***messagebox*** de error. Al contrario, se desplegará uno de proceso exitoso.

Texto

Descripción generada automáticamente

Método **Lectura**



Este método verifica si el archivo de entrada .lfp está vacío o si contiene texto. Si está vacío entonces se desplegará un ***messagebox*** de error. Si no, se se llama al método ***AnalizarLéxico***

Texto

Descripción generada automáticamente

Método Auxiliares

Una captura de pantalla de una computadora

Descripción generada automáticamente

Estos métodos auxiliares verifican si la entrada es una Letra, Numero o Símbolo permitidos según la expresión regular. Retorna un valor verdadero si está dentro del rango según el código ascii.

Método **AnalisisLéxico**

En el método Analizar se crean diferentes condiciones para el archivo de entrada, de este modo se lleva un control con los estados. Cada estado sigue el proceso del autómata finito determinista mostrado en el archivo pdf de esta carpeta.



Texto

Descripción generada automáticamente

Cada vez que se reconoce un token se crea un objeto de la clase Token:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Se agrega a la lista de Tokens para posteriormente ser analizado.

Si se encuentra un error se crea un objeto de la Clase Error:

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamenteMétodo **Estado\_Inicial**

Este método llama a la función p\_Claves1 cuya función es verificar si el primer elemento de la lista de tokens es igual al carácter o palabra esperada durante la ejecución

Texto

Descripción generada automáticamenteMétodos **Acciones**

Este método recorre la lista de comandos e inserta en la Consola (Text Widget) según el nombre de la instrucción.

Tabla de Tokens

Texto

Descripción generada automáticamente con confianza media

|  |  |
| --- | --- |
| Identificador | (L(L|D|’\_’)\*)$ |
| Símbolo | (SS\*)$ |
| Numero | ((‘+’|’-‘)?(D+)(‘.’D+)?)$ |
| Cadena | (‘”’(^ “) ´’”’)$ |
| Comentario multilínea | (‘ ‘ ‘ ‘ ‘ ‘ ‘ ‘ ‘ (^ ‘’’’)\* ‘ ‘ ‘ ‘ ‘ ‘ ‘ ‘ ‘)$ |
| Comentario Linea | (‘#’(^ \n)\*)$ |

Editor de código utilizado para la aplicación

**Visual Studio Code de Microsoft versión 1.61.2**

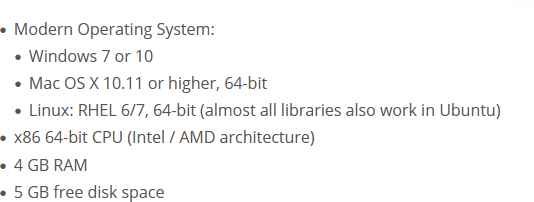
Texto

Descripción generada automáticamente

Captura de pantalla de computadora

Descripción generada automáticamente

Requerimientos para Python



Librerías Utilizadas

Texto

Descripción generada automáticamente

* ***Tkinter****:* Se considera un estándar para la interfaz gráfica de usuario para Python y es el que viene por defecto con la instalación para Microsoft Windows.