Universidad de San Carlos de Guatemala Facultad de Ingeniería

Ciencias y Sistemas

Lab. Lenguajes Formales y de Programación

**MANUAL TECNICO**

Justin Josue Aguirre Román 202004734

# INTRODUCCION

El presente documento describe los aspectos técnicos informáticos del sistema de información. El documento familiariza al personal técnico especializado encargado de las actividades de mantenimiento, revisión, solución de problemas, instalación y configuración del sistema, árboles y grafo de autómata utilizado para su desarrollo.

# OBJETIVO DE ESTE MANUAL

El objetivo primordial de este Manual es ayudar y guiar al técnico a informarse y utilizar herramientas utilizadas en el software auxiliar para la Facultad de Ingenieria de la Universidad de San Carlos de Guatemala, para de esa manera poder hacer uso de la información deseada para poder despejar todas las dudas existentes y para poder comprender:

* Guía para gestión de herramientas para poner en funcionamiento el sistema de reproducción de pensum.
* Conocer cómo utilizar el sistema, mediante una descripción detallada e ilustrada de las opciones.
* Conocer el alcance de toda la información por medio de una explicación detallada e ilustrada de cada una de las páginas que lo conforman el manual técnico.

# REQUERIMIENTOS

El sistema pude ser instalado en cualquier sistema operativo que cumpla con los siguientes requerimientos:

a. Sistema Operativo: Cualquiera con una fecha de salida del 2014 en adelante.

1. Windows
   1. Windows 10 (8u51 y superiores)
   2. Windows 8.x (escritorio)
   3. Windows 7 SP1
   4. Windows Vista SP2
   5. Windows Server 2008 R2 SP1 (64 bits)
   6. Windows Server 2012 y 2012 R2 (64 bits)
   7. RAM: 128 MB
   8. Espacio en disco: 124 MB para JRE; 2 MB para Java Update
   9. Procesador: Mínimo Pentium 2 a 266 MHz
   10. Exploradores: Internet Explorer 9 y superior, Firefox
2. Mac OS X
   1. Mac con Intel que ejecuta Mac OS X 10.8.3+, 10.9+
   2. Privilegios de administrador para la instalación
   3. Explorador de 64 bits
   4. Se requiere un explorador de 64 bits (Safari, por ejemplo) para ejecutar Oracle Java en Mac.
3. Linux
   1. Oracle Linux 5.5+1
   2. Oracle Linux 6.x (32 bits), 6.x (64 bits)2
   3. Oracle Linux 7.x (64 bits)2 (8u20 y superiores)
   4. Red Hat Enterprise Linux 5.5+1 6.x (32 bits), 6.x (64 bits)2
   5. Red Hat Enterprise Linux 7.x (64 bits)2 (8u20 y superiores)
   6. Suse Linux Enterprise Server 10 SP2+, 11.x
   7. Suse Linux Enterprise Server 12.x (64 bits)2 (8u31 y superiores)
   8. Ubuntu Linux 12.04 LTS, 13.x
   9. Ubuntu Linux 14.x (8u25 y superiores)
   10. Ubuntu Linux 15.04 (8u45 y superiores)
   11. Ubuntu Linux 15.10 (8u65 y superiores)

**DESCRIPCION DE ANALIZADORES Y GRAMATICA**

Para la utilización de un autómata se debe contar con la creación de su representación gráfica (por medio de un grafo) y a su vez la creación de los métodos de árbol que generan el grafo y la delimitación de los Tokens y que tipo de Tokens son aceptados por el mismo.

Lista de Tokens Aceptados por el proceso de análisis léxico:

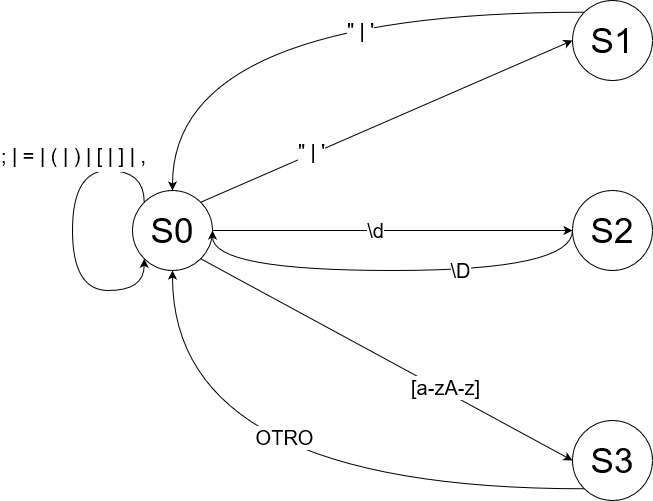
|  |  |
| --- | --- |
| **TOKEN** | **PATRON** |
| NOMBRE\_DE\_RED | Palabra nombre\_de\_red |
| CREAR\_CURSO | Palabra crearcurso |
| CONSOLA | Palabra consola |
| CONSOLALN | Palabra consolaln |
| CURSO\_POR\_NOMBRE | Palabra cursoPorNombre |
| CURSOS\_POR\_SEMESTRE | Palabra cursosPorSemestre |
| CURSO\_POR\_CODIGO | Palabra cursoPorCodigo |
| CURSOS\_PRE | Palabra cursosPretrrequisitos |
| CURSOS\_POST | Palabra cursosPostrrequisitos |
| GENERAR\_RED | Palabra generar\_Red |
| IGUAL | Carácter igual |
| PARENTESIS\_ABRE | Carácter paréntesis que abre |
| PARENTESIS\_CIERRA | Carácter paréntesis que cierra |
| COCHETE\_ABRE | Carácter corchete que abre |
| CORCHETE\_CIERRA | Carácter corchete que cierra |
| PUNTO\_COMA | Carácter punto y coma |
| COMA | Carácter coma |
| ENTERO | Números naturales |
| CADENA | Inicia y termina con " o ' |
| EOF | Carácter signo de dólar ($) |

**Tokens Validos, Elaboración propia 2021**

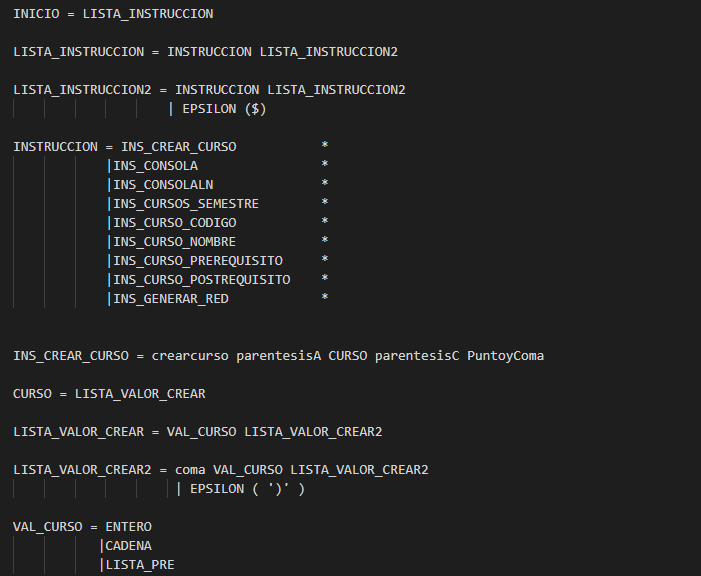
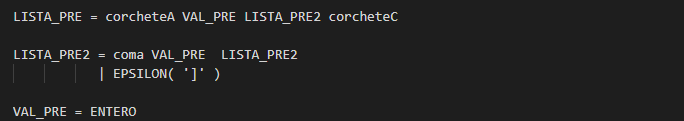
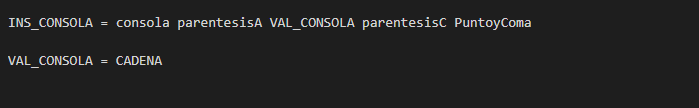
# DIAGRAMA AUTÓMATA ANÁLISIS LÉXICO

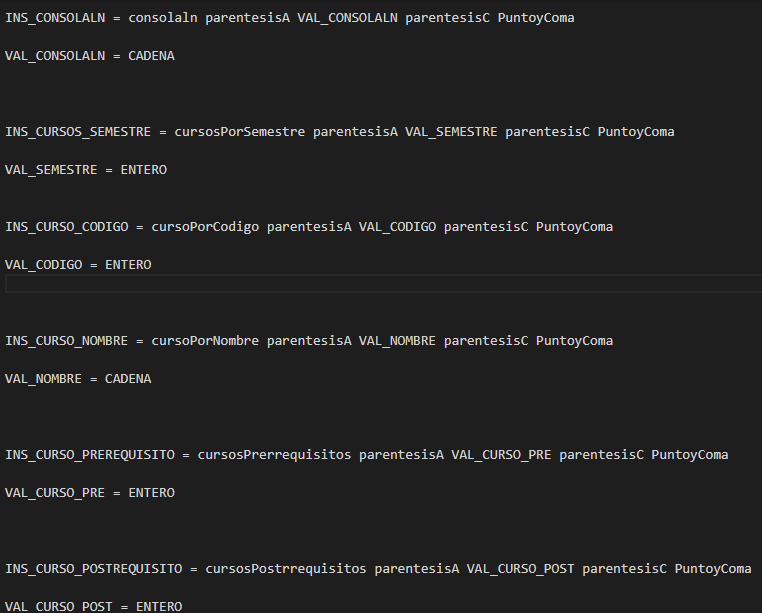
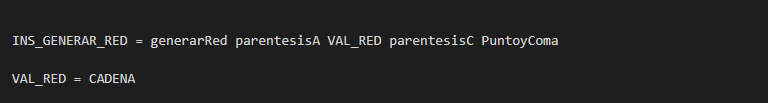
Se dividió el proceso de reconocimiento de los diferentes tokens con un análisis léxico que ayuda a identificar tanto las “palabras reservadas” como distintos símbolos que son aceptados por el lenguaje.

A continuación, el diagrama del autómata:

****

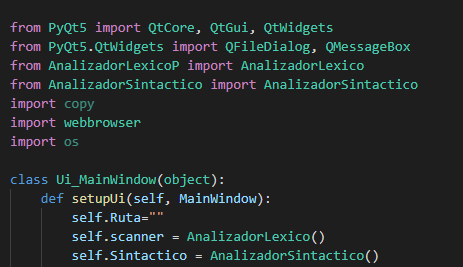
**GRAMATICA LIBRE DEL CONTEXTO**

****

****

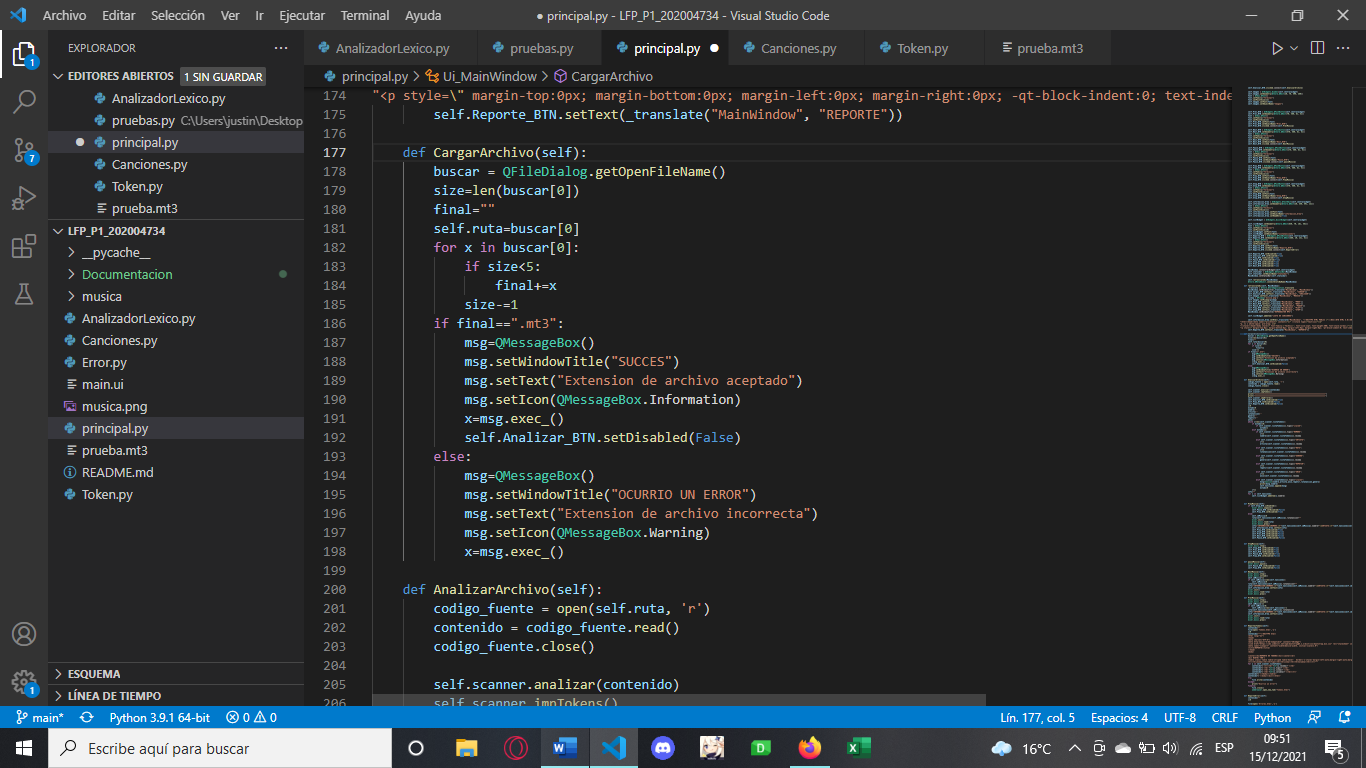
**METODOS UTILIZADOS**

Para el desarrollo del software se dispuso de la creación de distintas variables globales para el intercambio de información entre los distintos métodos y de la importación de distintas librerías añadidas de forma por defecto en Python.

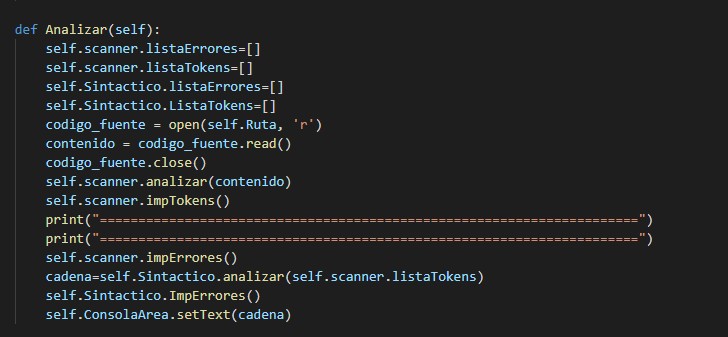


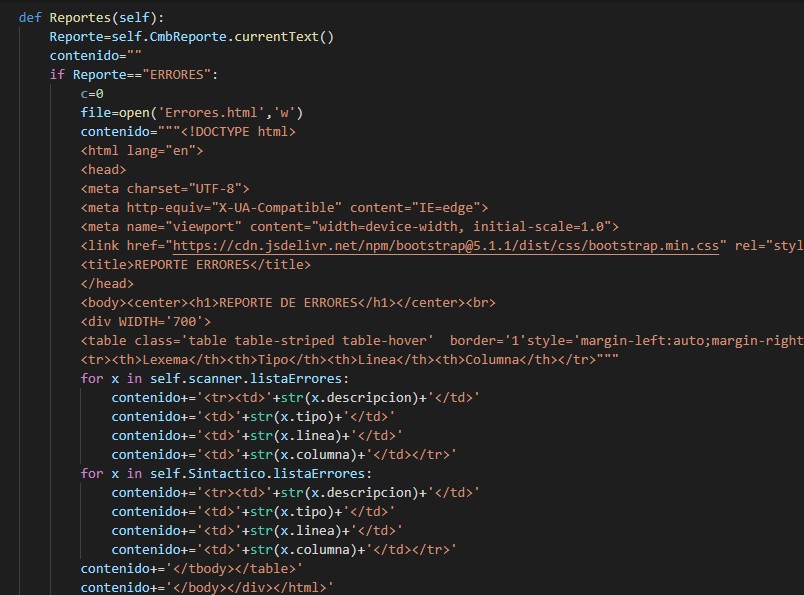
Y la creación de distintos métodos para la lectura, análisis y generación de reportes que enlazaran y conectaran las distintas clases y archivos tipo py que conforman y hacen posible el funcionamiento del software.

Para la función cargar se utiliza la librería y herramientas QFileDialog, que guardara la ruta del archivo a analizar, esta ruta será utilizada para saber la ubicación del documento, abrirlo y obtener el contenido en texto, sin embargo, primero se verificara el tipo de extensión del archivo seleccionado, la cual obligatoriamente debe ser una extensión de tipo “mt3”.



La función para analizar el archivo llama los métodos de los analizadores léxico y sintáctico, para ejecutar sus funciones de inicio justamente en ese orden. Al completar mostrar un mensaje que según se haya completado o no la lectura y análisis de forma satisfactoria será el mensaje a mostrar en la interfaz gráfica.



El método para los archivos de tipo HTML para los Reportes de Errores y Tokens usaran la misma base, las listas llenadas por medio de los análisis serán las llamadas en este apartado, recorriéndolas y guardándola en forma de una cadena de texto para llenar el formato de un archivo HTML.