

Universidad de San Carlos de Guatemala

Centro Universitario de Occidente

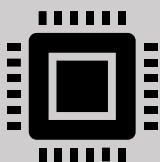
División de Ciencias de la ingeniería

Ingeniería

Lenguajes Formales y de Programacion Sección “A”

Ing. Daniel Gonzalez

Segundo Semestre 2025



Estudiante:

Pablo Alejandro Maldonado de León

Carné: 202430233

Manual de usuario Proyecto #1 2025



Descripción:

En el siguiente manual se le describirá paso a paso como se deberá hacer uso del analizador léxico y como interactuar con el para reconocer tokens y exportar el texto escrito en la aplicación. Además de mencionarle aspectos importantes que debe respetar o tener en cuenta el usuario para hacer uso de la aplicación al momento de experimentar toda la variedad de funcionalidades que la aplicación contiene y para poder visualizar las transiciones que realiza el automata.

“Bienvenido usuario”

Como primer punto se requieren ciertos recursos básicos para poder ejecutar el programa. Debido a que el programa cuenta con un ejecutable que permite usar la aplicación en cualquier dispositivo.

Recomendaciones:

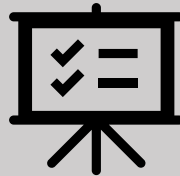
La aplicación se pensó para poderlo ejecutar desde la terminal un comando de Linux Ubuntu para desplegar la UI. Por lo tanto, se requiere para comodidad del usuario utilizar:

- Tener instalado el paquete JDK mínimo versión 17.01 para poder ejecutar el programa
- Computadora con Sistema Operativo Linux Ubuntu desde la versión 22 en adelante. Y si no se cuenta con Linux. Un sistema operativo Windows 7 en adelante.
- Tener un Mouse para mayor comodidad y un teclado
- Manejo de la terminal del Sistema Operativo

Pasos iniciales:

1. Encender el equipo
2. Descargar el archivo .zip
3. Extraer el archivo comprimido
4. Abrir la carpeta descomprimida
5. Buscar dentro de la carpeta “target”
6. Buscar el archivo ejecutable **.jar** y su ruta de acceso
7. Copiar la ruta de acceso

8. Abrir la terminal del sistema operativo (Ubuntu/Windows)
9. Copiar el directorio donde se descargó el .zip y escribir la terminación “cd” y pegar la ruta de acceso copiada, posteriormente presionando “Enter”.
10. Ingresar el comando .jar



INICIALIZACION DEL PROGRAMA

Para iniciar el juego se debe abrir la terminal del sistema en la ubicación del archivo “.jar”, para ello debe escribir el siguiente comando en la terminal:

“java -jar ProyectoNo1LFP-1.0-SNAPSHOT.jar”

Al momento de presionar “**Enter**” se empezará a ejecutar el programa indicando que se ha iniciado el programa mostrando directamente el Menu principal para que se pueda empezar a operar con la aplicación.

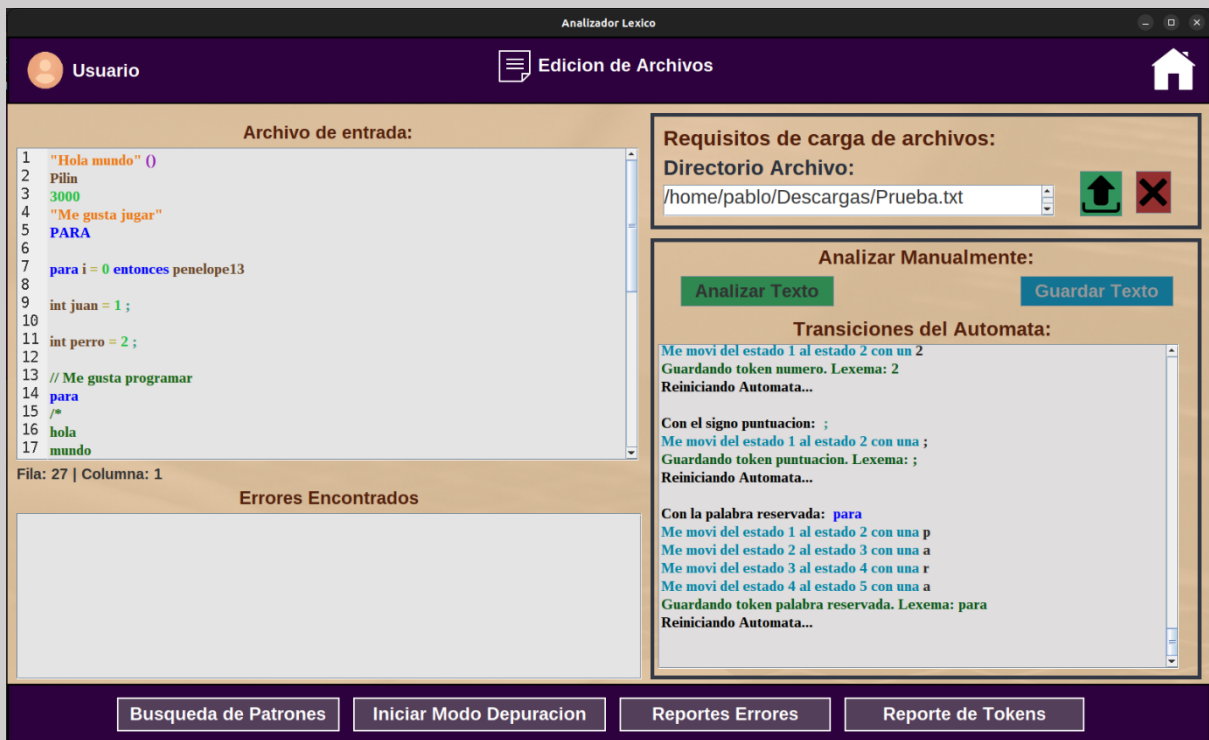
Tras darle “**Enter** ” se muestra la ventana principal la cual cuenta con una interfaz intuitiva donde a través de la operación que se desee realizar con tan solo darle click se va a poder ejecutar funciones que permita usar búsquedas, entrar en modo depuración o reportes además esta es capaz de ir reconociendo en tiempo real las palabras que se van escribiendo.

Es importante saber que la clasificación de tokens se da en base a los espacios o saltos de línea con los que cuenta el Log de entrada, así que se debe siempre respetar el debido espaciamiento para tokens, en especial para caracteres pertenecientes a la gramática del automata, esta gramática se encuentra definida por expresiones regulares, ofreciendo modularidad, también es importante mencionar que si se escriben palabras juntas y estas tienen errores el automata es capaz de poder identificar los tokens.

Menú Principal

Una vez lanzado el ejecutable se vera la interfaz principal que cuenta con una organización básica que en base a lo que desee realizar se le desplegara un menú que contiene el nombre de las diversas funcionalidades.

La interfaz está pensada para que sea lo más intuitiva para el usuario de usar. Al igual que está restringida para que el usuario no toque acciones que no se deben cuando se esté procesado datos o haciendo cierta acción que pueda ser interrumpida o hecha de forma que se hizo al revés, esta aplicación cuenta con la validación necesaria para que el usuario haga las cosas progresivamente.

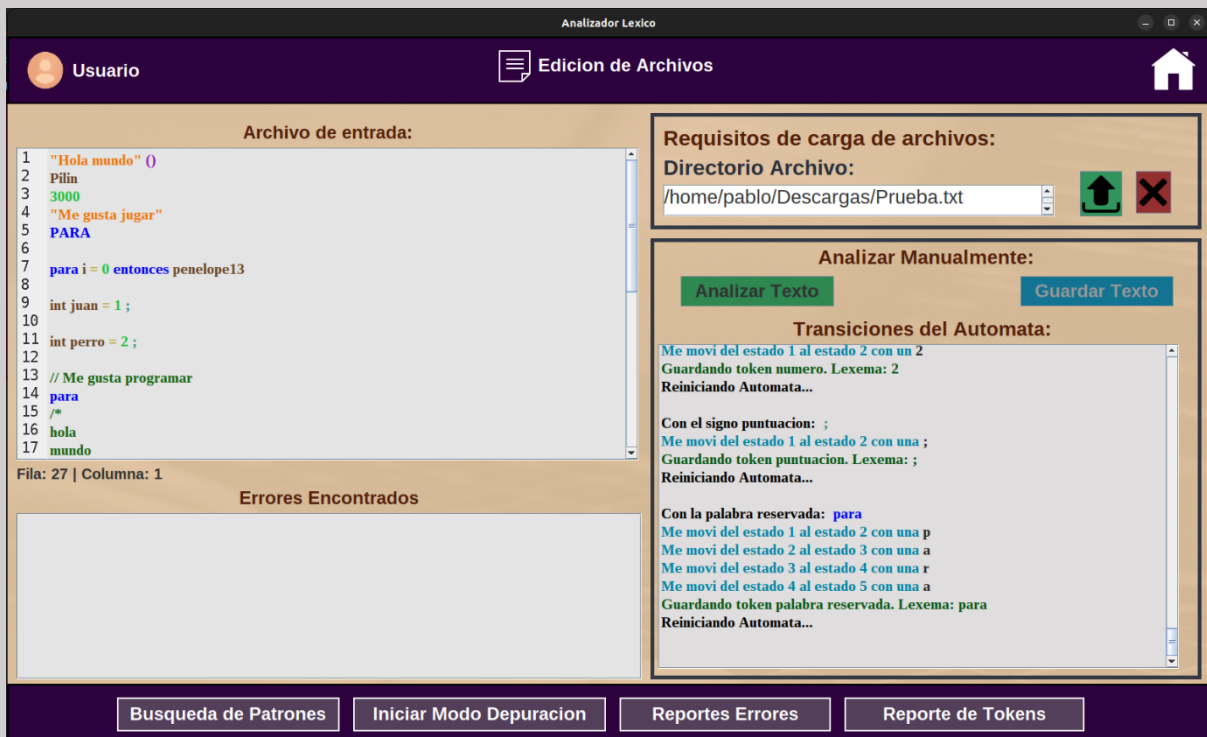


Detalles:

Apenas iniciar la aplicación se le desplegará el menú principal, donde se podrá subir un archivo o directamente poder escribir en el log y la aplicación o el analizador léxico lo ira detectando automáticamente y A TIEMPO REAL.

Opciones:

- **Analizar texto:** Esta permite analizar manualmente el texto por si en dado caso falla, pero es casi improbable asi que es mejor el reconocimiento automatico.
- **Guardar Texto:** Esta funcionalidad lo que permite es exportar a cualquier destinatario que se elija apenas presionar el botón se podrá seleccionar donde se desea guardar y posteriormente se exportara el Log de entrada hacia el destinatario elegido de tal forma que se puedan crear archivos de texto en base al analizador lexico.
- **Log de Errores:** Este marca los errores que surgen al momento de ir escribiendo en el log de entrada para que el usuario sea consciente y lo corrija.
- **Log de Transiciones:** Este ilustra a tiempo real las transiciones que genera el automata para poder reconocer que tipo de token es y representar cuando este reinicia su lectura para pasar al análisis de la siguiente cadena de entrada.



Visualizacion de las transiciones del automata:

Transiciones del Automata:

Con la cadena de texto: "Hola mundo"

Me movi del estado 1 al estado 2 con una "

Me movi del estado 2 al estado 3 con una H

Me movi del estado 3 al estado 4 con una o

Me movi del estado 4 al estado 5 con una l

Me movi del estado 5 al estado 6 con una a

Me movi del estado 6 al estado 7 con una

Me movi del estado 7 al estado 8 con una m

Me movi del estado 8 al estado 9 con una u

Me movi del estado 9 al estado 10 con una n

Me movi del estado 10 al estado 11 con una d

Me movi del estado 11 al estado 12 con una o

Me movi del estado 12 al estado 13 con una "

Guardando token cadena de texto. Lexema: "Hola mundo"

Reiniciando Automata...

Con el signo de agrupacion: ()

Me movi del estado 1 al estado 2 con una (

Log de errores:

Fila: 7 | Columna: 32

Errores Encontrados

()@ <- Error, en Signo de agrupacion no registrado en ()@

penelope1@33 <- Error, en No es letra ni numero en penelope1@

Botón de Home:

Está ubicado en la parte superior derecha cuya funcionalidad es regresar al menú principal donde se puede reelegir cierta acción hacia la que se desee viajar. Es importante saber que media vez se presione este botón todo lo que se está haciendo se va a reiniciar al estado principal que es el analizador lexico y sus demás logs de salida que dispone para mostrar la información.

Busqueda de patrones

Esta es una de las 3 funcionalidades bases del programa que permite ubicar cualquier palabra o terminación en el log de entrada, con tan solo presionar en la parte inferior izquierda se deshabilitara la edición de texto y se habilitara la búsqueda permitiendo escribir la palabra o las palabras que sean y las ubicara automáticamente.



Detalles a tener en cuenta:

- La búsqueda se basa en cualquier similitud de la palabra subrayando todas las posibles coincidenticas por palabra que existan.
- Si la palabra no se encuentra avisara que no existe
- Es importante saber que buscara y resaltara la palabra encontrada en cualquier línea del documento en la que se repita

Deteccion de errores avanzada

El programa cuenta con la detección de palabras reservadas avanzada, esta consta de una lógica cuya funcionalidad se basa en marcar como error si una palabra reservada está mal escrita, y al momento de pasar el cursor encima aparecerá la corrección de la palabra para indicar como se debe de escribir.



Detalles a tener en cuenta:

- Estas palabras que se escriban similares a una palabra reservada se marcaran como error. Por lo tanto evitar plantear identificadores con estructura muy similar a las palabras reservadas.

Modo depuración

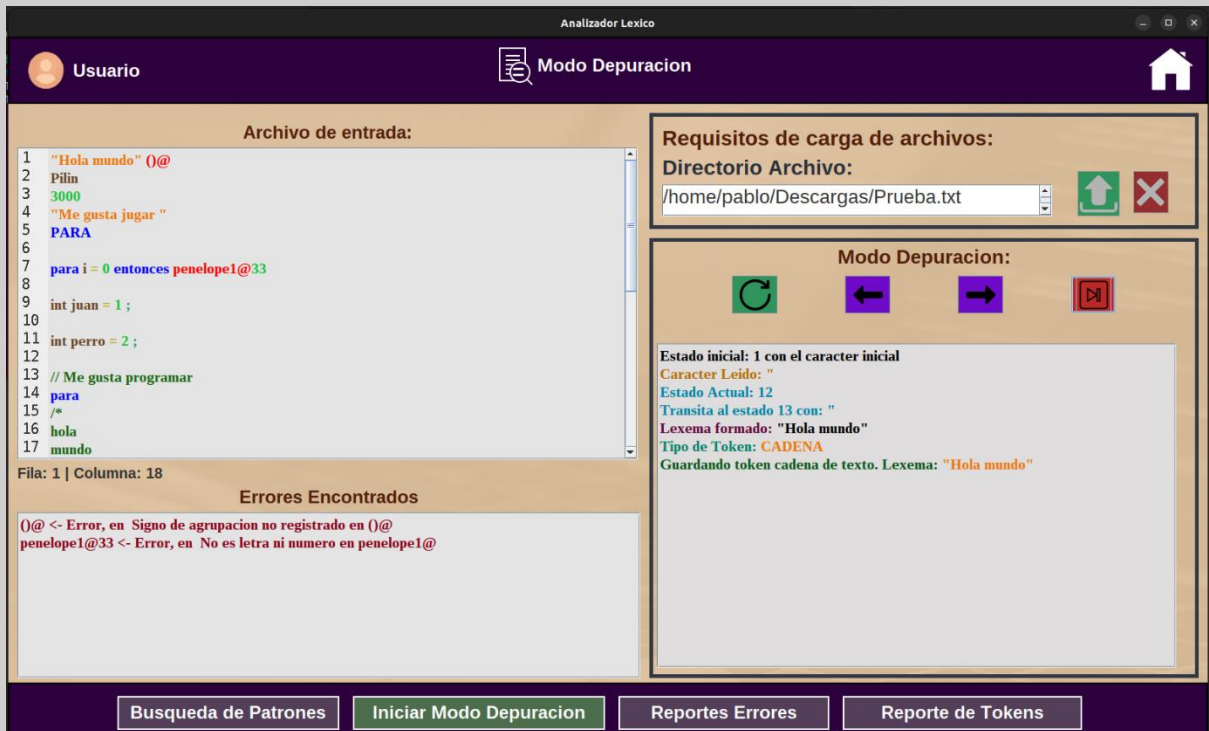
Este es uno de los modos característicos de la aplicación. Permite analizar paso a paso cada palabra escrita dentro del editor de texto. Permitiendo visualizar cada estado de transición que genera el autómata finito determinista, permitiendo avanzar paso a paso o consumir por completo el lexema. De igual forma se puede regresar paso a paso.



Avanzar paso a paso:

- **Avanzar:** Este botón con una flecha apuntando hacia la derecha, va consumiendo carácter a carácter y va mostrando las transiciones que va haciendo el AFD.
- **Regresar:** Este botón con una flecha hacia la izquierda, va regresando carácter a carácter y va saltando entre los lexemas que se van leyendo.

Visualización del fin de una depuración de un lexema:



Avanzar Consumir lexemas:

- **Reiniciar:** Este botón con una flecha en forma de círculo hacia la derecha, permite reiniciar el análisis de la lectura de la palabra regresando al estado inicial de la palabra.
- **Consumir lexema:** Este botón con una flecha hacia la izquierda, avanza hasta el estado final de análisis del lexema, permitiendo finalizar de golpe todo el análisis del lexema.

Reportes

Esta es una funcionalidad con la que se cuenta en todo momento en la que se pueden consultar los resultados del analizador lexico a su vez que estos mismos reportes permiten determinar un resumen de la cantidad de errores que existieron y si en dado caso **no hay errores, se pueden generar los reportes de lexemas.**



Detalles a tener en cuenta:

- Si el log de entrada esta vacío **NO SE PODRA GENERAR REPORTES**
- Al momento de ejecutar cualquier análisis o el simple hecho de cambiar un carácter se puede consultar el reporte general de lexemas.
- En dado caso de haber errores los reportes de **LEXEMAS no funcionaran**
- En dado caso de no haber errores los reportes de **ERRORES no se mostrarán**

Visualizacion de reportes

Reporte de Errores:

Este indicara la cantidad de errores registrados en el log de errores

The screenshot shows the 'Analizador Lexico' application interface. A modal window titled 'Reporte de Errores' is open, displaying a table of errors. The background shows a code editor with a sample program and a sidebar with file upload options.

Reporte de Errores:

Cadena de Error	Posicion	Descripcion
Signo de agrupacion no registrado en ()@	"[Fila 1 , Columna 18]"	Se esperaba un agrupacion
No es letra ni numero en penelope1@	"[Fila 7 , Columna 40]"	Se esperaba un identificador

Buttons: **Generar Reporte** (green), **Exportar Reporte** (purple).

Background elements: Code editor with lines 1-17, sidebar with 'Requisitos de carga de archivos:' and 'Archivos de entrada:', bottom bar with 'Busqueda de Patrones', 'Iniciar Modo Depuracion', 'Reportes Errores', and 'Reporte de Tokens'.

Al momento de darle exportar se creara un archivo csv que lo podrá visualizar en tablas o en cualquier editor que permita formato .csv y pueda crear las tablas como bonus de la aplicacion.

Reporte de Lexemas:

Estos permiten visualizar la ubicación y el tipo de token que compone a cada palabra que el usuario escribe.

Analizador Lexico

Reporte Conteo Lexemas

Visualizacion de Reportes

Reportes Analizador Lexico:

Reporte de Tokens:

Reporte Tokens Lexemas:

Nombre Token	Lexema	Posicion
CADENA	"Hola mundo"	"F(1) , C(1-12)"
AGRUPACION	()	"F(1) , C(16-17)"
IDENTIFICADOR	Pilin	"F(2) , C(1-5)"
NUMERO	3000	"F(3) , C(1-4)"
CADENA	"Me gusta jugar "	"F(4) , C(1-17)"
PALABRA_RESERVADA	PARA	"F(5) , C(1-4)"
PALABRA_RESERVADA	para	"F(7) , C(1-4)"
IDENTIFICADOR	i	"F(7) , C(8-8)"
OPERADOR	=	"F(7) , C(12-12)"
NUMERO	0	"F(7) , C(16-16)"
PALABRA_RESERVADA	entonces	"F(7) , C(20-27)"

Generar Reporte

Reporte Conteo Lexemas:

Lexema	Tipo de Token	Cantidad
"Hola mundo"	CADENA	1
()	AGRUPACION	1
Pilin	IDENTIFICADOR	1
3000	NUMERO	1
"Me gusta jugar "	CADENA	1
PARA	PALABRA_RESERVADA	1
para	PALABRA_RESERVADA	2
i	IDENTIFICADOR	1
=	OPERADOR	3
0	NUMERO	1
entonces	PALABRA_RESERVADA	1

Generar Reporte

Lexema Seleccionado:

Lexema:

Token:

Generar Diagrama

Busqueda de Patrones Iniciar Modo Depuracion Reportes Errores Reporte de Tokens

Reporte de conteo de Lexemas:

Estos permiten visualizar el tipo de lexema que se ingresó y que tipo de token es, a su vez que se mostrara cuantas veces se repite en el log de entrada.

The screenshot shows a web application titled 'Reporte Conteo Lexemas' with a sidebar on the left containing a code editor with the following text:

```
1 "Hola m
2 Pilin
3 3000
4 "Me gu
5 PARA
6
7 para i =
8
9 int juan
10
11 int perro
12
13 // Me gu
14 para
15 /*
16 hola
17 mundo
```

The main content area is titled 'Reportes Analizador Lexico:' and contains two reports:

Reporte de Tokens:

Nombre Token	Lexema	Posicion
CADENA	"Hola mundo"	"F(1) , C(1-12)"
AGRUPACION	()	"F(1) , C(16-17)"
IDENTIFICADOR	Pilin	"F(2) , C(1-5)"
NUMERO	3000	"F(3) , C(1-4)"
CADENA	"Me gusta jugar "	"F(4) , C(1-17)"
PALABRA_RESERVADA	PARA	"F(5) , C(1-4)"
PALABRA_RESERVADA	para	"F(7) , C(1-4)"
IDENTIFICADOR	i	"F(7) , C(8-8)"
OPERADOR	=	"F(7) , C(12-12)"
NUMERO	0	"F(7) , C(16-16)"
PALABRA_RESERVADA	entonces	"F(7) , C(20-27)"

Reporte Conteo Lexemas:

Lexema	Tipo de Token	Cantidad
"Hola mundo"	CADENA	1
()	AGRUPACION	1
Pilin	IDENTIFICADOR	1
3000	NUMERO	1
"Me gusta jugar "	CADENA	1
PARA	PALABRA_RESERVADA	1
para	PALABRA_RESERVADA	2
i	IDENTIFICADOR	1
=	OPERADOR	3
0	NUMERO	1
entonces	PALABRA_RESERVADA	1

Below the reports, there is a section titled 'Lexema Seleccionado:' with the following fields:

Lexema: "Hola mundo"

Token: CADENA

At the bottom of the application, there are four buttons: 'Busqueda de Patrones', 'Iniciar Modo Depuracion', 'Reportes Errores', and 'Reporte de Tokens'.

Generar Diagrama de transiciones:

Esta es una de las funcionalidades graficas mas bonitas que posee la aplicación. Permite ilustrar todas las transiciones que realiza el Automata finito determinista. Mostrando el estado inicial y el estado final con el que se genera el estado de aceptación del lexema. Con tan solo **dar click** sobre el lexema que se desee ver se permitirá generar el diagrama de transiciones.

