Universidad De San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ciencias y Sistemas

Lenguajes formales de programación
Sección “B-”

“MANUAL TÉCNICO”

Diego André Mazariegos Barrientos

Carné: 202003975

Objetivos

General:

Brindar al lector una guía que contenga la información del manejo de clases, atributos, métodos y del desarrollo de la interfaz gráfica para facilitar futuras actualizaciones y futuras modificaciones realizadas por terceros.

Específicos:

- Mostrar al lector una descripción lo más completa y detallada posible del SO, IDE entre otros utilizados para el desarrollo de la aplicación.

- Proporcionar al lector una concepción y explicación técnica - formal de los procesos y relaciones entre métodos y atributos que conforman la parte operativa de la aplicación.

Introducción

Este manual técnico tiene como finalidad dar a conocer al lector que pueda requerir hacer modificaciones futuras al software el desarrollo de la aplicación denominada “Proyecto 1” desarrollada durante el transcurso de las primeras semanas de septiembre, indicando el IDE utilizado para su creación, su versión, requerimientos del sistema, etc…

La aplicación tiene como objetivo cumplir con los requerimientos solicitados por la empresa pixel art debido al crecimiento en la demanda de sus productos, por lo que la aplicación posee un analizador léxico para la lectura y el análisis de los datos de entrada con un formato previamente establecido para la generación de una imagen mediante el uso de cuadriculas. Con dichos datos se generan archivos HTML y CSS en los cuales se muestran las imágenes resultantes de dicha entrada con extensión PXLA, a su vez la aplicación permite visualizar el conjunto de imágenes junto con los filtros solicitados en el archivo de entrada por medio de la interfaz de forma agradable, y como último la aplicación cuenta con una opción de generación de reportes de los errores léxicos y tokens encontrados en el archivo de entrada

Descripción de la Solución

Para poder desarrollar este proyecto se analizó lo que el cliente solicitaba y lo que el cliente realmente necesitaba, sus restricciones tanto humanas, de equipo y financieras del proyecto y empresa; y el ambiente y forma de trabajo de los futuros operadores de la aplicación.

Entre las consideraciones encontramos con mayor prioridad están:

- Realizar la lectura del archivo de entrada con el formato correcto y verificar si la sintaxis utilizada en el mismo es correcta.

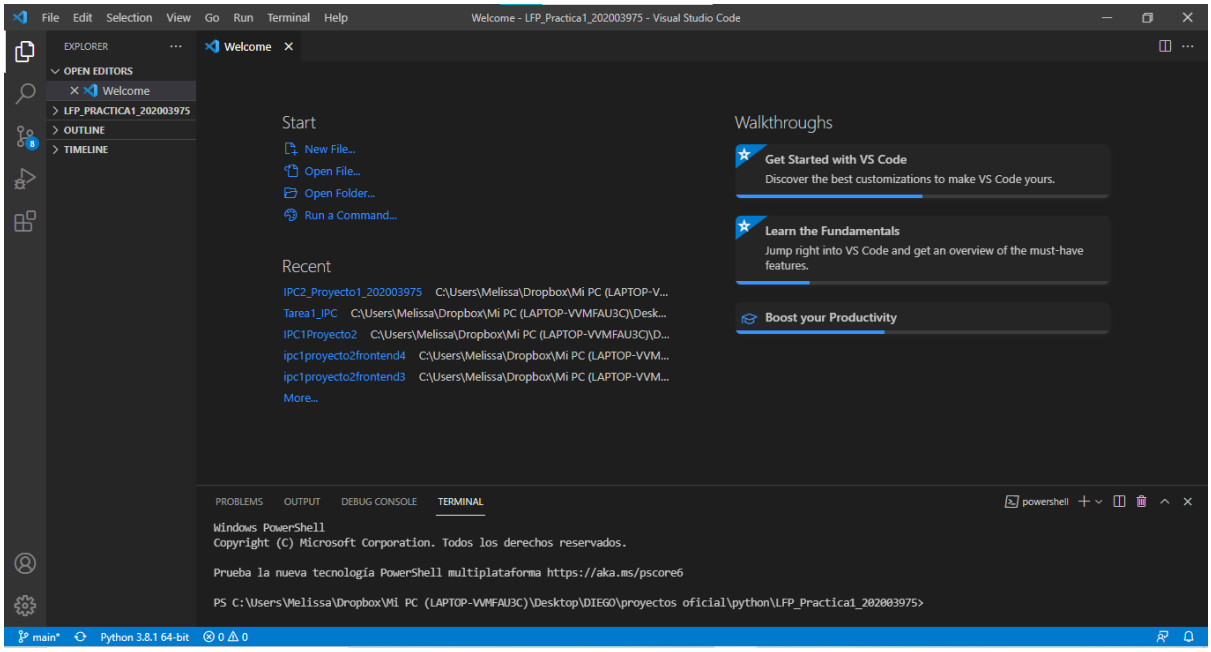
- Análisis completo del archivo de entrada una vez verificada que la entrada sea correcta procediendo a elaborar los respectivos archivos de visualización de cada imagen solicitada.

- Generación de reportes de los componentes léxicos encontrados en el documento de entrada, así como un reporte de errores léxicos si es que existiesen algunos.

- Presentación de imagen seleccionada en la interfaz gráfica de forma agradable y fácil de usar.

IDE

El IDE con el que se desarrolló el proyecto “Práctica 1” fue Visual Studio Code, debido a su apoyo al desarrollador gracias a su asistente que detecta errores semánticos, sintácticos del código por lo cual ayudan y hacen que la duración de la fase de programación sea más corta, además posee una interfaz muy agradable y fácil de entender en el modo debugging.



Requerimientos de IDE:

- **Hardware**
  Visual Studio Code es una pequeña descarga (<200 MB) y ocupa un espacio en disco de <500 MB. VS Code es liviano.

  Se recomienda:

  Procesador de 1,6 GHz o más rápido.
  1 GB de RAM.

- **Software**
  - OS X El Capitan (10.11+).
  - Windows 7 (con .NET Framework 4.5.2), 8.0, 8.1 y 10 (32 y 64 bits).
  - Linux (Debian): Ubuntu Desktop 16.04, Debian 9.
  - Linux (Red Hat): Red Hat Enterprise Linux 7, CentOS 8, Fedora 24.

- **Requisitos adicionales de Windows**

  Se requiere Microsoft .NET Framework 4.5.2 para VS Code. Si está utilizando Windows 7, asegúrese de que .NET Framework 4.5.2 esté instalado.

- **Requisitos adicionales de Linux**

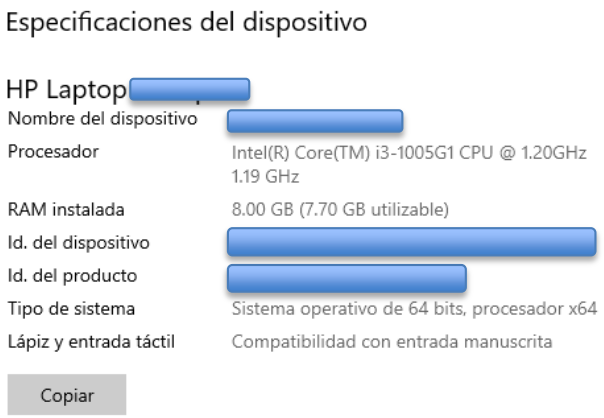
  GLIBCXX versión 3.4.21 o posterior.

  GLIBC versión 2.15 o posterior.

Requisitos del programa

Sistema operativo	Memoria RAM mínima	Memoria RAM recomendada	Espacio en disco mínimo	Espacio en disco recomendado
El programa puede ser instalado en cualquier sistema operativo.	512 MB	1 GB	10 MB	100 MB

Máquina en la cual fue desarrollado el programa



Librerías Utilizadas

Las librerías utilizadas para el desarrollo de este proyecto fueron:

```
from tkinter import Tk
from tkinter import Menu
from tkinter import filedialog
from typing import Text
from tkinter import messagebox
from tkinter import ttk
from tkinter import Button, Label, messagebox
import webbrowser
import imgkit
from PIL import ImageTk, Image
import pathlib
```

De dichas librerías la más relevante y fundamental implementada en el proyecto es la librería de tkinter puesto que fue utilizada para el manejo de la interfaz gráfica y poder implementar todos los componentes para que la solución fuese satisfactoria y cumpla con los requerimientos principales.

A su vez la librería de pathlib fue utilizada para extraer la dirección de la aplicación en donde se esté ejecutando en alguna parte de la máquina.

Y finalmente la librería PIL la cual se implementó para poder extraer las imágenes de los archivos Html generados que contienen la información de las imágenes y así poder mostrar al usuario la imagen en la interfaz gráfica.

Diagrama flujo

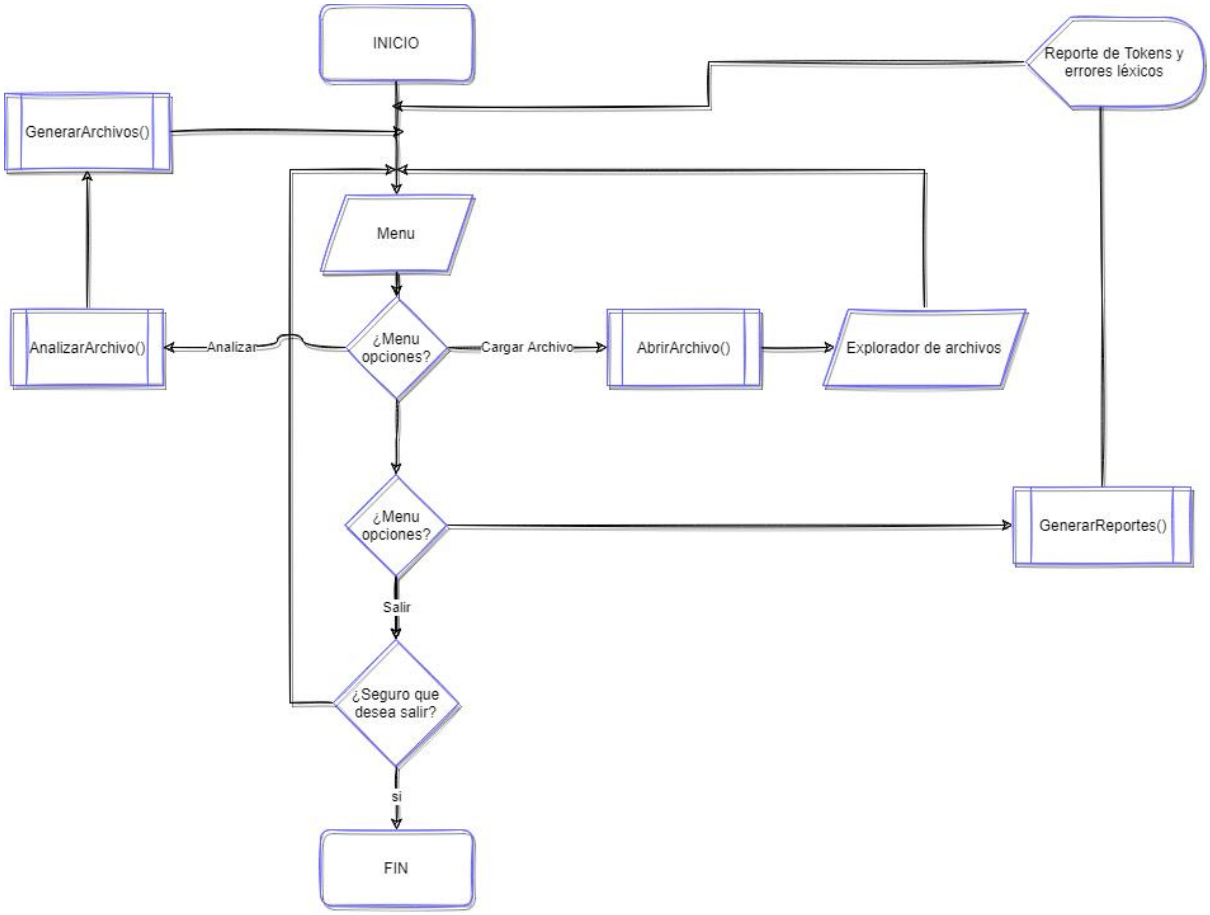


Tabla de tokens

Token	Lexema	Patrón
TITULO	TITULO	TITULO
Asignación	=	=
Cadena	"Palabra"	Le = [A_Z, a_z] -> Palabra = Le+ -> "palabra"
Punto y coma	;	;
ANCHO	ANCHO	ANCHO
Entero	Digito	Di = [0_9] -> Digito = Di+ -> Digito
ALTO	ALTO	ALTO
FILAS	FILAS	FILAS
COLUMNAS	COLUMNAS	COLUMNAS
CELDAS	CELDAS	CELDAS
Llave abre	{	{
Booleano	TRUE	TRUE
Booleano	FALSE	FALSE
Coma	,	,
Código de color	#DDDDDD	Le = [A_Z] -> Di = [0_9] -> (#(Di|Le){6})
Corchete cierra	]	]
Corchete abre	[	[
FILTROS	FILTROS	FILTROS
MIRRORX	MIRRORX	MIRRORX
MIRRORY	MIRRORY	MIRRORY
DOUBLEMIRROR	DOUBLEMIRROR	DOUBLEMIRROR
Separador imágenes	@@@@	@@@@
Llave cierra	}	}

Proceso método del árbol

Paso 1) expresión regular.

Expresión regular 2.0

Le = [A_Z, a_z]
Palabra = Le+
Di = [0_9]
Digito = Di+

Expresión regular

TITULO**|**=**|**”Palabra”**|**ANCHO**|**ALTO**|**Digito**|**;**|**FILAS**|**COLUMNAS**|**CELDAS**|**{**|**}**|**[**|**]**|**,**|**FALSE**|**TRUE**|**(#(D**|**L){6})**|**FILTROS**|**MIRRORX**|**MIRRORY**|**DOUBLEMIRROR**|**@@@@

Paso 1.1) Agregar al final de la expresión regular el $.

Expresión regular

(TITULO**|**=**|**”Palabra”**|**ANCHO**|**ALTO**|**Digito**|**;**|**FILAS**|**COLUMNAS**|**CELDAS**|**{**|**}**|**[**|**]**|**,**|**FALSE**|**TRUE**|**(#(D**|**L){6})**|**FILTROS**|**MIRRORX**|**MIRRORY**|**DOUBLEMIRROR**|**@@@@**)$**

Paso 2) Formar Árbol de sintaxis.

Para la realización del árbol se utilizó la plataforma draw.io el cual es un editor de diagramas online gratis.

Para este paso se determinó para cada nodo lo siguiente.

- Si era Anulable o no Anulable marcando con un V si es Anulable y F si no lo es.
- Se determinó para cada nodo sus siguientes.
- Se determinó para cada nodo sus últimos.

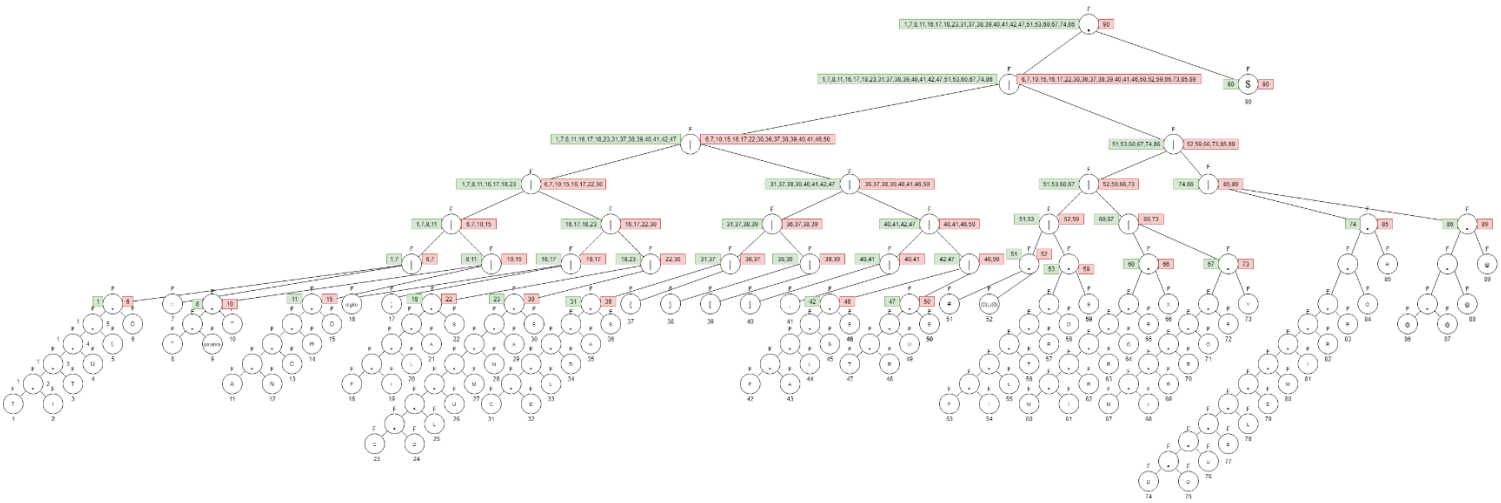


Figura 1. Diagrama del árbol binario con todos los procedimientos realizados.
Fuente: elaboración propia, 2021.

Paso 2.1) Calcular tabla de siguientes.

CALCULANDO SIGUIENTES		
VALOR	**HOJA**	**SIGUIENTES**
T	1	2
I	2	3
T	3	4
U	4	5
L	5	6
O	6	90
=	7	90
"	8	9
palabra	9	10
"	10	90
A	11	12
N	12	13
C	13	14
H	14	15
O	15	90
digito	16	90
;	17	90

F	18	19
I	19	20
L	20	21
A	21	22
S	22	90
C	23	24
O	24	25
L	25	26
U	26	27
M	27	28
N	28	29
A	29	30
S	30	90
C	31	32
E	32	33
L	33	34
D	34	35
A	35	36
S	36	90
{	37	90
}	38	90
[	39	90
]	40	90
,	41	90
F	42	43
A	43	44
L	44	45
S	45	46
E	46	90
T	47	48
R	48	49
U	49	50
E	50	90
#	51	52
(D|L){6}	52	90
F	53	54
I	54	55
L	55	56
T	56	57
R	57	58
O	58	59
S	59	90
M	60	61
I	61	62
R	62	63

R	63	64
O	64	65
R	65	66
X	66	90
M	67	68
I	68	69
R	69	70
R	70	71
O	71	72
R	72	73
Y	73	90
D	74	75
O	75	76
U	76	77
B	77	78
L	78	79
E	79	80
M	80	81
I	81	82
R	82	83
R	83	84
O	84	85
R	85	90
@	86	87
@	87	88
@	88	89
@	89	90
$	90	---

Paso 2.2) Construyendo tabla de transiciones.

	CONSTRUYENDO TABLA DE TRANSICIONES		
	ESTADO	VALORES	SIGUIENTES
Inicio/ aceptación	S0	1,7,8,11,16,17,18,23,31,37,38,39,40,41,42,47,51,53,60,67,74,86	1(T){2,48} = S1 7(=){90}=S2 8(")){9}=S3 11(A){12} = S4 16(digito){90} =S2 17(;){90}=S2 18(F){19,43,54} = S5 23(C){24,32} = S6
			37({){90} = S2 38(}){90} = S2 39([){90} = S2 40(]){90} = S2 41(,){90} = S2 51(#){52} = S7 60(M){61,68}=S8 74(D){75}=S9 86(@){87} = S10
	S1	2,48	2(I){3}=S11 48(R){49} = S12
aceptación	S2	90	---
	S3	9	9(palabra){10}=S13
	S4	12	12(N){13}=S14
	S5	19,43,54	19,54(I){20,55}=S15
			43(A){44}=S16
	S6	24,32	24(O){25}=S17
			32(E){33}=S18
	S7	52	52((D|L){6}){90}=S2
	S8	61,68	61(I){62,69}=S19
	S9	75	75(O){76}=S20
	S10	87	87(@){88}=S21
	S11	3	3(T){4}=S22
	S12	49	49(U){50}=S23
	S13	10	10(")){90}=S2
	S14	13	13(C){14}=S24
	S15	20,55	20,55(L){21,56}=S25
	S16	44	44(L){45}=S26
	S17	25	25(L){26}=S27
	S18	33	33(L){34}=S28
	S19	62,69	62(R){63,70}=S29
	S20	76	76(U){77}=S30
	S21	88	88(@){89}=S31
	S22	4	4(U){5}=S32
	S23	50	50(E){90}=S2
	S24	14	14(H){15}=S33
	S25	21,56	21(A){22}=S34
			56(T){57}=S35
	S26	45	45(S){46}=S36
	S27	26	26(U){27}=S37
	S28	34	34(D){35}=S38
	S29	63,70	63,70(R){64,71}=S39

	S30	77	77(B){78}=S40
	S31	89	89(@){90}=S2
	S32	5	5(L){6}=S41
	S33	15	15(O){90}=S2
	S34	22	22(S){90}=S2
	S35	57	57(R){58}=S42
	S36	46	46(E){90}=S2
	S37	27	27(M){28}=S43
	S38	35	35(A){36}=S44
	S39	64,71	64,71(O){65,72}=S45
	S40	78	78(L){79}=S46
	S41	6	6(O){90}=S2
	S42	58	58(0){59}=S47
	S43	28	28(N){29}=S48
	S44	36	36(S){90}=S2
	S45	65,72	65,72(R){66,73}=S49
	S46	79	79(E){80}=S50
	S47	59	59(S){90}=S2
	S48	29	29(A){30}=S51
	S49	66,73	66(X){90}=S2
			73(Y){90}=S2
	S50	80	80(M){81}=S52
	S51	30	30(S){90}=S2
	S52	81	81(I){82}=S53
	S53	82	82(R){83}=S54
	S54	83	83(R){84}=S55
	S55	84	84(O){85}=S56
	S56	85	85(R){90}=S2

Paso 2.3) Construir Tabla de transiciones

	TABLA DE TRANSICIONES																																							
	ESTADOS	T	I	U	L	O	=	"	palabra	A	N	C	H	digito	;	F	S	M	E	D	{	}	[	]	,	R	#	(D|L){6}	X	Y	B	@	$
o	S0	S1					S2	S3		S4		S6		S2	S2	S5		S8		S9	S2	S2	S2	S2	S2		S7						S10	
	S1		S11																							S12								
$	S2																																	
	S3								S13																									
	S4										S14																							
	S5		S15							S16																								
	S6					S17													S18															
	S7																											S2						
	S8		S19																															
	S9					S20																												
	S10																															S21		
	S11	S22																																
	S12			S23																														
	S13							S2																										
	S14											S24																						
	S15				S25																													
	S16				S26																													
	S17				S27																													
	S18				S28																													
	S19																									S29							
	S20			S30																														
	S21																															S31		
	S22			S32																														
	S23												S33						S2															
	S24									S34																								
	S25	S35								S34																								
	S26																S36																	
	S27			S37																														
	S28																			S38														
	S29																									S39							
	S30																														S40			
	S31																															S2		
	S32				S41																													
	S33					S2																												
	S34																S2																	
	S35																									S42							
	S36																		S2															
	S37																	S43																
	S38									S44																								
	S39					S45																												
	S40				S46																													

	S41					S2																										
	S42					S47																										
	S43										S48																				
	S44																S2														
	S45																									S49						
	S46																		S50													
	S47																S2														
	S48									S51																					
	S49																											S2	S2			
	S50																	S52														
	S51																S2															
	S52		S53																													
	S53																									S54						
	S54																									S55						
	S55					S56																										
	S56																									S2						

Paso 3) Formar el Autómata Finito Determinista (AFD).

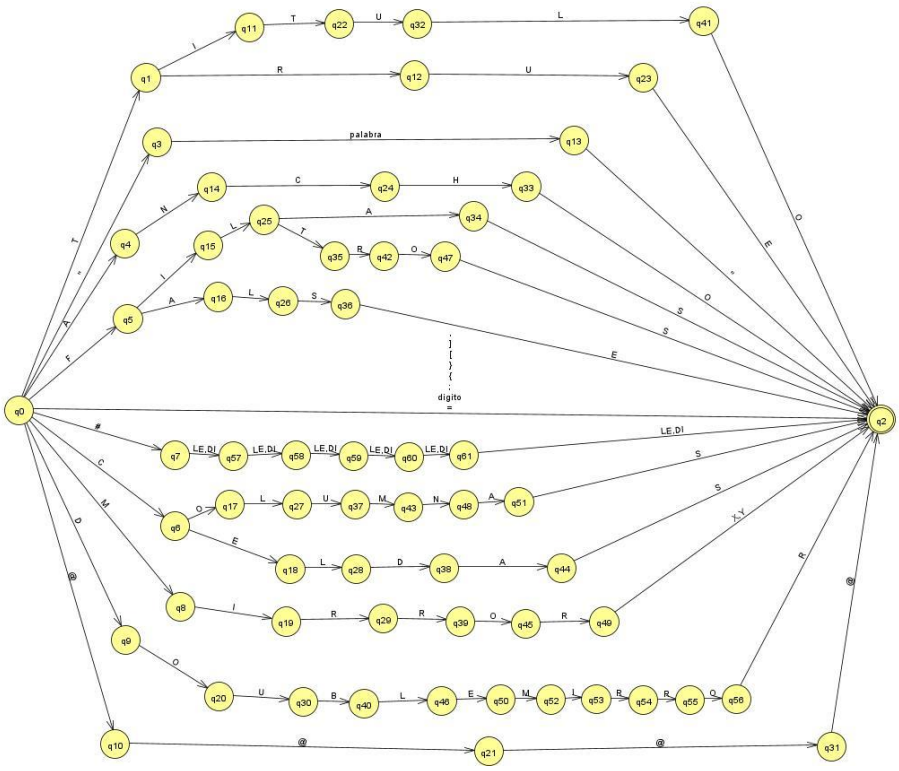


Figura 2. Autómata Finito Determinista (AFD) resultante del método del árbol.
Fuente: elaboración propia, 2021.