

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTADA DE INGENIERIA
ESCUELA DE CIENCIAS Y SISTEMAS
LENGUAJES FORMALES Y DE PROGRAMACION
PRIMER SEMESTRE 2,023

Manual Técnico

Autor: Brayan Estiben Micá Pérez

201907343

Introducción

El presente manual se presentan las especificaciones técnicas del programa de carga de archivos con extensión. .JSON, se muestra cada función por separado para poder hacer modificaciones al código posteriormente.

Objetivos

- ❑ Otorgar soporte técnico a los desarrolladores para poder modificar la aplicación.

Requerimientos

- ❑ Equipo: Dual Core o superior
- ❑ Sistema Operativo: Windows 7 o superior
- ❑ Memoria RAM: 1 Gb mínimo o superior
- ❑ Resolución gráfica: mínimo 800*600 o superior
- ❑ Tener instalado Python 3 con todas sus librerías
- ❑ Tener instalado el gestor de paquetes pip para descargar extensiones

INDICE DEL MANUAL

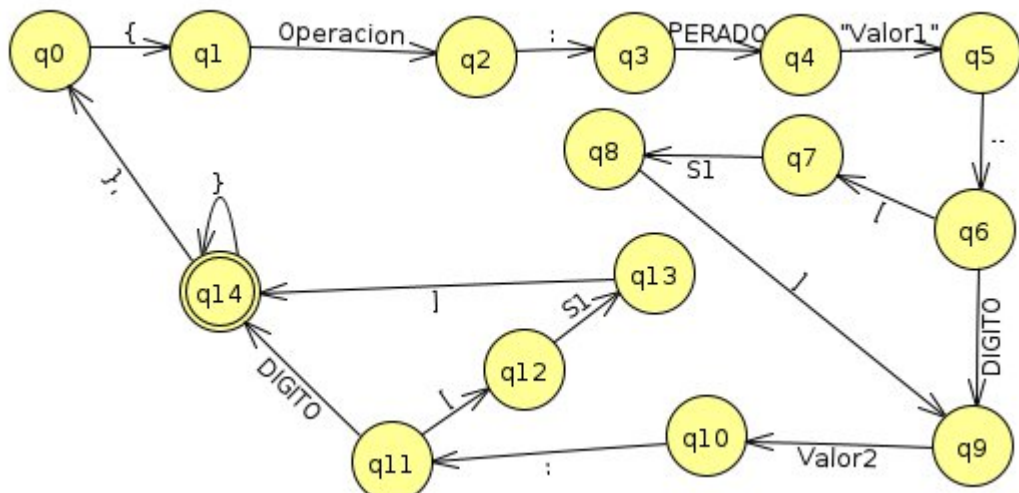
1. Gramática
2. Cargar archivo
3. Mostrar datos del archivo
4. Eliminar un registro
5. Agregar un registro
6. Modificar un registro
7. Ver el conteo de créditos

1. Gramatica:

2.

<p>S14 FINAL</p> <p>S0 -> { S1</p> <p>S1 -> "Operacion" S2</p> <p>S2 -> : S3</p> <p>S3 -> OPERADOR S4</p> <p>S4 -> "Valor1" S5</p> <p>S5 -> : S6</p> <p>S6 -> DIGITO S9</p> <p>S6 -> [S7</p> <p>S7 -> S1 S8</p> <p>S8 ->] S9</p> <p>S9 -> "Valor2" S10</p> <p>S10 -> : S11</p> <p>S11 -> DIGITO S14</p> <p>S11 -> [S12</p> <p>S12 -> S1 S13</p> <p>S13 ->] S14</p> <p>S14 -> }</p> <p> }, S0</p>	<p>OPERADOR -> "Suma"</p> <p> "Resta"</p> <p> "Multiplicacion"</p> <p> "Division"</p>	<p>DIGITO</p> <p>D0 -> [0-9] D0</p> <p> D1</p> <p> D3</p> <p>D1 -> . D2</p> <p>D2 -> [0-9] D2</p> <p> D3</p> <p>D3 -> Epsilon</p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------

Autómata:



3. Cargar archivo

La opción de cargar archivo verifica la extensión del archivo así como la selección del mismo además se tiene una función que recupera dicha respuesta si el archivo es seleccionado o no.

```
# Lectura del archivo
def Abrir(self):
    file = filedialog.askopenfilename(defaultextension=".json",
    filetypes=[("Archivos de texto", "*.json")])

    if file:
        with open(file, 'r') as archivo:
            # 2. Leer el contenido del archivo
            lineas = archivo.readlines()

            # 3. Cerrar el archivo
            archivo.close()
            self.CuadroEditor.delete(1.0, tk.END)
            for linea in lineas:
                self.CuadroEditor.insert(tk.END, linea)
            tkinter.messagebox.showinfo(title="Cargando archivo", message=("El archivo se cargo correctamente"))
    else:
        tkinter.messagebox.showinfo(title="Cargando archivo", message=("No seleccionaste ningun archivo"))
```

4. Guardar un archivo

Esta opción le permitirá al usuario guardar un archivo .json que puede ser un nuevo archivo o un archivo que se este editando.

```
# Guardar archivo
def Guardar(self):
    archivo = filedialog.asksaveasfilename(defaultextension=".json", filetypes=[("Archivos de texto", "*.json")])
    if archivo:
        with open(archivo, 'w') as f:
            text = self.CuadroEditor.get('1.0', tk.END)
            f.write(text)
```

5. Analizar

Para eliminar un registro se utilizó una llama al `getDatos` para buscar los registros en la lista y posteriormente se eliminó a través de la creación de una nueva lista cuando se encontró dicho registro, al encontrar la posición del registro a eliminar se utilizó la instrucción `pop` para eliminar el registro y posteriormente se hizo una actualización a el registro `setDatos()`.

```

def _token(self, token:str, estado_actual:str, estado_sig:str):
    if self.lineas[self.index] != " ":
        text = self._juntar(self.index, len(token))
        if self._analizar(token, text):
            self.index += len(token) - 1
            self.columna += len(token) - 1
            return estado_sig
        else:
            return 'ERROR'
    else:
        return estado_actual

```

5. Errores

El recuperador de errores permite recoger una lista de cada error de un archivo leído.

6. Salir.

La función salir permitirá salir al usuario una vez se de presionar sobre el boton o presionando sobre la x de la pantalla principal.

```

# Sub menus desplegables
# submenu de archivo
Archivo = Menu(menu, tearoff=0)
Archivo.add_command(label="Abrir", command=self.Abrir)
Archivo.add_command(label="Guardar", command=self.Guardar)
Archivo.add_command(label="Analizar", command=self.Analizar)
Archivo.add_command(label="Errores", command=self.Errores)
Archivo.add_separator()
Archivo.add_command(label="Salir", command=ventana.destroy)

```

7. Manuales de usuario y técnico

La siguiente función abre un enlace respectivo a cada manual en una pestaña del navegador predefinido en la computadora.

```

# Abriendo un pdf del manual de usuario
def abrirMusuario(self):
    absolutepath = os.path.abspath(__file__)
    Directorio = os.path.dirname(absolutepath)
    path = Directorio + r"/Manual de usuario.pdf"
    if webbrowser.open_new(path) is not None:
        tkinter.messagebox.showinfo(title="Archivo enc
    else:
        tkinter.messagebox.showinfo(title="Error al ab

# Abriendo un pdf del manual tecnico
def abrirMtecnico(self):
    absolutepath = os.path.abspath(__file__)
    Directorio = os.path.dirname(absolutepath)
    path = Directorio + r"/Manual Tecnico.pdf"
    if webbrowser.open_new(path) is not None:
        tkinter.messagebox.showinfo(title="Archivo enc
    else:
        tkinter.messagebox.showinfo(title="Error al ab

```

Para esta práctica se utilizó la programación orientada a objetos como base para el desarrollo de la aplicación, además de la combinación de la librería Tkinter para la visualización de la parte gráfica