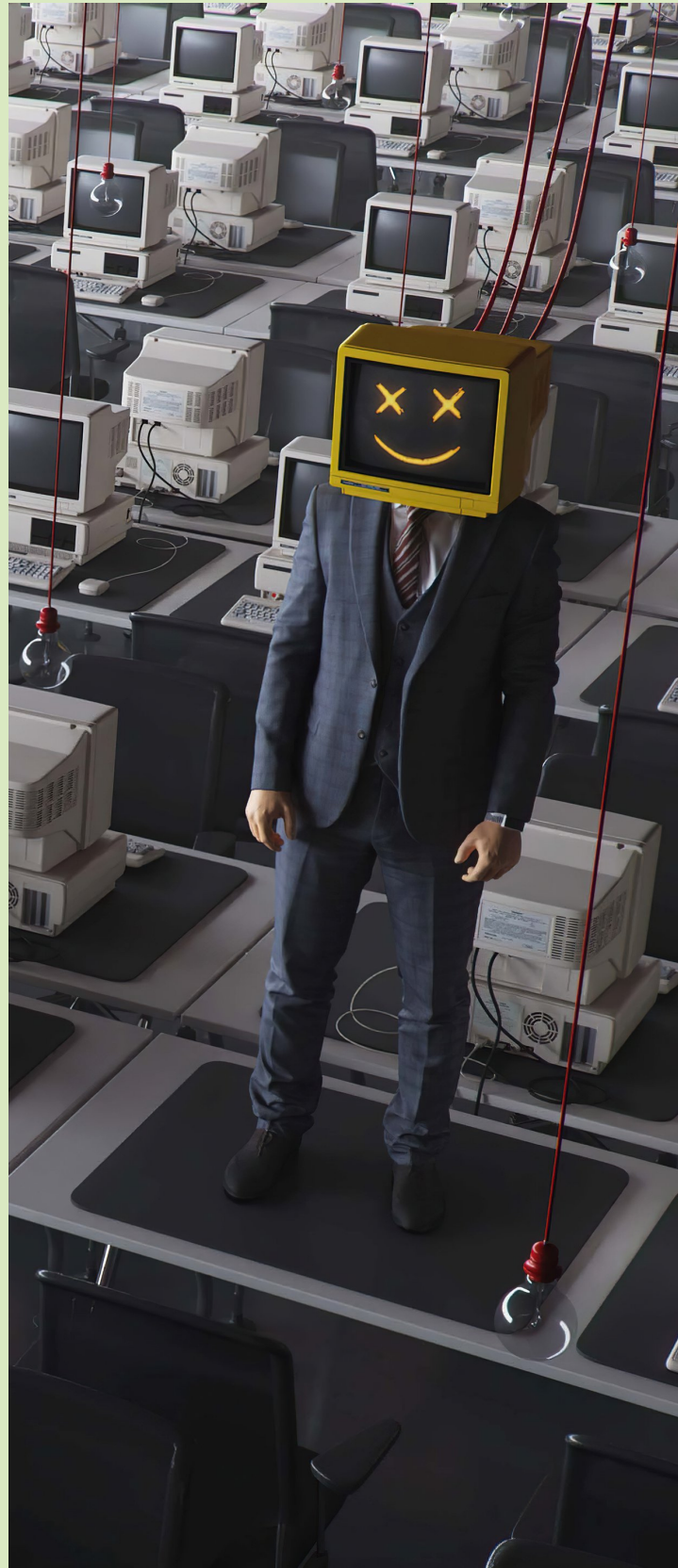


LENGUAJES  
FORMALES DE  
PROGRAMACIÓN

# MANUAL TÉCNICO



JOSÉ FUENTES  
202001228

# INTRODUCCION



---

Poder ayudar al técnico en la comprensión de la lógica que se utilizo en la realización del programa. Definiendo claramente el código del autómata usado y de igual manera realizando el diagrama para ver el funcionamiento. Por medio de un archivo de extinción .gpw el cual tendrá el contenido que se examinara para dar la salida del archivo.

# INFORME SOBRE EL PROGRAMA

---

Este es un programa realizado en el lenguaje de python versión 3.10.5 con la cual se desarrollo el software que ayudar al usuario a poder generar una pagina (html) con su respectivo estilo (css) por medio de un archivo (.pgw). Sabiendo que el programa contendrá tres contenedores los cuales son los siguientes:

- Controles: indicara los atributos que contiene la pagina que se realizara.
- Propiedades: son todas las característica de cada uno de los atributos que contendrá la pagina.
- Colocación: acá se indicara en que parte de la pagina estará cada atributo.

Este es un sistema de 64 bits, se puede utilizar en los siguientes sistemas operativos:

- Windows 7
- Windows 8
- Windows 8.1
- Windows 10
- Windows 11

requisitos para el uso del programa:

- 512 MB de memoria RAM
- espacio en disco duro de 256 MB
- Procesador: mínimo Pentium hasta la gama alta del mismo.

# IMPORTACIONES DEL PROGRAMA

---

Al iniciar la programación de este programa, se realizando varias importaciones para el mejor manejo del código e interacción con el usuario.

- La importación de tkinter es para la parte grafica, la cual ayudara a la interacción con el usuario a lo largo de la utilización del programa..
- La importación de Analizador es el autómata que contendrá todos los pasos para la lectura correcta de cada uno de los datos que vendrán en el archivo.

```
from tkinter import *  
from tkinter.filedialog import askopenfilename  
from tkinter import messagebox  
from tkinter import filedialog  
from Analizador import Automata
```

## TOKENS

---

En esta parte estarán todos los caracteres y uniones de palabras que se tomaran como tokens, los que se utilizaran para la lectura del archivo y la escritura de la misma.

```
tokens = [  
    'Controles',  
    'Propiedades',  
    'Colocacion',  
    'Etiqueta',  
    'Boton',  
    'Check',  
    'RadioButton',  
    'Texto',  
    'AreaTexto',  
    'Clave',  
    'Contenedor']
```

# GRAMÁTICA

---

Acá se examinara la parte gramática de cada uno de las números, cadenas de texto que se ingresaran para saber de que tipo es: entero, decimal, cadenas de texto, saltos de línea, espacios, etc. Se dejara el código de un numero decimal.

```
if self.estado == 0:
    if caracter == "<":
        self.estado = 1
```

Dependiendo del token se realiza diferentes acciones, ya que cada palabra reservada tiene un diferente procedimiento para la elaboración del código y redacción del mismo.

```
def setTexto(self,control,texto):
    redactar = '''<label id="''' + control + '''>''' +
    texto + '''</label>'''
    print(redactar)
```

En esta parte se muestra la escritura de un label tomando las características ingresadas en en propiedades.

```
def setColorLetra(self,control,color):
    self.css.write(control + '{\n')
    self.css.write('color: rgb(' )
    for i in color:
        self.css.write(i + ',')
    self.css.write(');\n')
    self.css.write('}\n')
```

