

MANUAL TECNICO



EMILIO SANCHEZ DE LA ROSA

Materia: Diseño de Interfaces

2-8-2024

Contenido

Introducción	3
Requisitos del Sistema	3
Estructura del Proyecto	3
Código Fuente	4
Importaciones y Funciones Auxiliares	4
Ventana Principal y Navegación	5
Cálculo de BMI y Resultados	7
Ventana de Información Adicional.....	9
Ventana Principal y Botón de Navegación	10
Descripción de Componentes.....	10
Ventana Principal:.....	10
Segunda Ventana:.....	10
Funciones Auxiliares:	10
Cálculo del BMI:	11
Ventana de Información Adicional:.....	11
Instalación.....	11
• Instalar Python	11
• Instalar PIL (Pillow):	11
Ejecución de la Aplicación	11
• Guardar el Código	11
• Preparar Imágenes	11
• Ejecutar la Aplicación	12
Uso de la Aplicación	12
• Pantalla Principal	12
• Ventana de Ingreso de Datos	12
• Resultado del BMI	13
• Más Información	14
Errores Comunes	15

• Error en los Valores	15
• Imágenes no Encontradas	15
Convertir la Aplicación de Cálculo de BMI en un Archivo .exe	15
Requisitos del Sistema	15
Instalación de PyInstaller	15
Convertir el Código a un Archivo .exe.....	15
Ajustes Adicionales	16
Diagramas de Flujo:.....	17

Introducción

Este manual técnico describe la estructura, componentes y funcionamiento de una aplicación de cálculo de Índice de Masa Corporal (BMI) desarrollada con Python utilizando la librería tkinter para la interfaz gráfica y PIL para el manejo de imágenes.

Requisitos del Sistema

- Sistema operativo: Windows, macOS, Linux
- Versión de Python Instalado
- Librerías de Python:
 - tkinter
 - PIL (Pillow)
- Mu Editor
- Visual Studio

Estructura del Proyecto

- bmi_calculator.py: Archivo principal de la aplicación.
- images/: Carpeta que contiene todas las imágenes utilizadas en la aplicación.
- background.jpg: Imagen de fondo para las ventanas.
- calculate.png: Imagen del botón de calcular.
- calculate_hover.png: Imagen del botón de calcular al pasar el ratón.
- info.png: Imagen del botón de información.
- info_hover.png: Imagen del botón de información al pasar el ratón.
- main_button.png: Imagen del botón principal.
- bmi_table.png: Imagen de la tabla de BMI.

Código Fuente

Importaciones y Funciones Auxiliares

```
import tkinter as tk

from tkinter import messagebox, Toplevel

from PIL import Image, ImageTk


def resize_image(image_path, width, height):

    image = Image.open(image_path)

    image = image.resize((width, height), Image.LANCZOS)

    return ImageTk.PhotoImage(image)


def set_background(window, image_path):

    background_image = Image.open(image_path)

    background_photo = ImageTk.PhotoImage(background_image)

    background_label = tk.Label(window, image=background_photo)

    background_label.image = background_photo

    background_label.place(x=0, y=0, relwidth=1, relheight=1)

    background_label.bind("<Enter>", lambda event: on_enter(event,
background_label))

    background_label.bind("<Leave>", lambda event: on_leave(event,
background_label))


def make_entry_oval(entry_widget):

    entry_widget.config(relief="flat", bd=1, highlightthickness=0)

    entry_widget.bind("<FocusIn>", lambda event:
entry_widget.config(relief="groove", bd=2))
```

```
entry_widget.bind("<FocusOut>", lambda event: entry_widget.config(relief="flat",
bd=1))
```

```
def make_entry_highlighted(entry_widget):
```

```
    entry_widget.config(highlightbackground="black", highlightcolor="black",
highlightthickness=1)
```

```
def on_enter(event, widget):
```

```
    widget.config(cursor="hand2")
```

```
def on_leave(event, widget):
```

```
    widget.config(cursor="")
```

- **Importaciones:** tkinter para la interfaz gráfica y PIL para el manejo de imágenes.
- **resize_image:** Función para redimensionar imágenes.
- **set_background:** Función para establecer una imagen de fondo en una ventana.
- **make_entry_oval:** Función para estilizar las entradas de texto.
- **make_entry_highlighted:** Función para resaltar las entradas de texto.
- **on_enter/on_leave:** Funciones para cambiar el cursor al pasar sobre un widget.

Ventana Principal y Navegación

```
def open_second_window():
```

```
    second_window = Toplevel(root)
```

```
    second_window.title("Ingresar Datos")
```

```
    set_background(second_window, "images/background.jpg")
```

```
frame_style = {'bg': 'white', 'bd': 2, 'relief': 'groove'}
```

```
entry_frame = tk.Frame(second_window, bg='white', padx=20, pady=10)
```

```
entry_frame.pack(padx=20, pady=20)
```

```
tk.Label(entry_frame, text="Nombre:", font=("Arial", 12)).grid(row=0, column=0,  
padx=10, pady=5, sticky='w')
```

```
name_entry = tk.Entry(entry_frame, font=("Arial", 12))
```

```
name_entry.grid(row=0, column=1, padx=10, pady=5)
```

```
make_entry_oval(name_entry)
```

```
make_entry_highlighted(name_entry)
```

```
tk.Label(entry_frame, text="Edad:", font=("Arial", 12)).grid(row=1, column=0,  
padx=10, pady=5, sticky='w')
```

```
age_entry = tk.Entry(entry_frame, font=("Arial", 12))
```

```
age_entry.grid(row=1, column=1, padx=10, pady=5)
```

```
make_entry_oval(age_entry)
```

```
make_entry_highlighted(age_entry)
```

```
tk.Label(entry_frame, text="Peso (kg):", font=("Arial", 12)).grid(row=2, column=0,  
padx=10, pady=5, sticky='w')
```

```
weight_entry = tk.Entry(entry_frame, font=("Arial", 12))
```

```
weight_entry.grid(row=2, column=1, padx=10, pady=5)
```

```
make_entry_oval(weight_entry)
```

```
make_entry_highlighted(weight_entry)
```

```
tk.Label(entry_frame, text="Altura (m):", font=("Arial", 12)).grid(row=3, column=0,  
padx=10, pady=5, sticky='w')
```

```
height_entry = tk.Entry(entry_frame, font=("Arial", 12))
```

```
height_entry.grid(row=3, column=1, padx=10, pady=5)
```

```
make_entry_oval(height_entry)
```

```
make_entry_highlighted(height_entry)
```

- **open_second_window**: Función para abrir la segunda ventana donde se ingresan los datos del usuario.
- **set_background**: Configura el fondo de la ventana con la imagen especificada.
- **entry_frame**: Marco que contiene los campos de entrada para el nombre, edad, peso y altura.

Cálculo de BMI y Resultados

```
def calculate_bmi():
```

```
    try:
```

```
        name = name_entry.get()
```

```
        age = age_entry.get()
```

```
        weight = float(weight_entry.get())
```

```
        height = float(height_entry.get())
```

```
        bmi = weight / (height ** 2)
```

```
        bmi_status = ""
```

```
    if bmi < 18.5:
```

```
        bmi_status = "Bajo peso"
```

```
    elif 18.5 <= bmi < 24.9:
```

```
        bmi_status = "Peso normal"
```

```
    elif 25 <= bmi < 29.9:
```

```
        bmi_status = "Sobrepeso"
```

```
    else:
```

```
        bmi_status = "Obesidad"
```



```

result_window = Toplevel(root)

result_window.title("Resultado IMC")

set_background(result_window, "images/background.jpg")


result_label = tk.Label(result_window, text=f"Nombre: {name}\nEdad:
{age}\nPeso: {weight} kg\nAltura: {height} m\n\nBMI: {bmi:.2f}\nEstado:
{bmi_status}",
                        font=("Arial", 12), bg='white', padx=20, pady=20)

result_label.pack()


except ValueError:

    messagebox.showerror("Error", "Por favor, ingresa valores válidos para
    peso y altura.")


calculate_image = resize_image("images/calculate.png", 150, 80)
calculate_hover_image = resize_image("images/calculate_hover.png", 150, 80)

calculate_button = tk.Button(second_window, image=calculate_image,
command=calculate_bmi, borderwidth=0)

calculate_button.image = calculate_image

calculate_button.pack(pady=10)

calculate_button.bind("<Enter>", lambda event: on_enter(event,
calculate_button))

calculate_button.bind("<Leave>", lambda event: on_leave(event,
calculate_button))

```

Ventana de Información Adicional

```
def open_third_window():  
    third_window = Toplevel(root)  
    third_window.title("Más Información")  
    set_background(third_window, "images/background.jpg")  
  
    third_image = resize_image("images/bmi_table.png", 400, 400)  
    tk.Label(third_window, image=third_image).pack()  
    third_window.mainloop()  
  
    info_image = resize_image("images/info.png", 150, 80)  
    info_hover_image = resize_image("images/info_hover.png", 150, 80)  
    info_button = tk.Button(second_window, image=info_image,  
command=open_third_window, borderwidth=0)  
    info_button.image = info_image  
    info_button.pack(pady=10)  
    info_button.bind("<Enter>", lambda event: on_enter(event, info_button))  
    info_button.bind("<Leave>", lambda event: on_leave(event, info_button))  
  
    second_window.mainloop()
```

- **open_third_window:** Función para abrir una ventana con información adicional sobre el BMI.
- **info_button:** Botón que abre la ventana de información adicional, con efectos al pasar el ratón.

Ventana Principal y Botón de Navegación

```
root.title("Cálculo de BMI")
```

```
set_background(root, "images/background.jpg")
```

```
button_image = resize_image("images/main_button.png", 250, 250)
```

```
image_button = tk.Button(root, image=button_image,  
command=open_second_window, borderwidth=0)
```

```
image_button.pack(pady=20)
```

```
root.mainloop()
```

- **root**: La ventana principal de la aplicación.
- **set_background**: Configura la imagen de fondo de la ventana principal.
- **image_button**: Botón de imagen que abre la segunda ventana para ingresar los datos del usuario.

Descripción de Componentes

Ventana Principal:

- Contiene un botón de imagen que, al hacer clic, abre la segunda ventana.

Segunda Ventana:

- Contiene campos de entrada para el nombre, edad, peso y altura.
- Botón "Calcular" que, al hacer clic, calcula el BMI y abre una ventana con los resultados.
- Botón "Información" que, al hacer clic, abre una ventana con información adicional sobre el BMI.

Funciones Auxiliares:

- **resize_image**: Redimensiona imágenes a un tamaño especificado.
- **set_background**: Establece una imagen de fondo en una ventana.

- `make_entry_oval`: Estiliza las entradas de texto con bordes ovalados.
- `make_entry_highlighted`: Resalta las entradas de texto.
- `on_enter`, `on_leave`: Cambian el cursor al pasar sobre un widget.

Cálculo del BMI:

- Toma los valores de peso y altura ingresados, calcula el BMI y muestra los resultados en una nueva ventana.
- Maneja errores de conversión de valores para asegurar que los datos ingresados sean válidos.

Ventana de Información Adicional:

- Muestra una imagen con información adicional sobre los rangos de BMI.

Instalación

- **Instalar Python**: Asegúrate de tener Python 3.x instalado en tu sistema u otra versión. Puedes descargarlo desde python.org o desde la Microsoft Store
- **Instalar PIL (Pillow)**: Abre la terminal o el símbolo del sistema y ejecuta:

```
pip install pillow
```

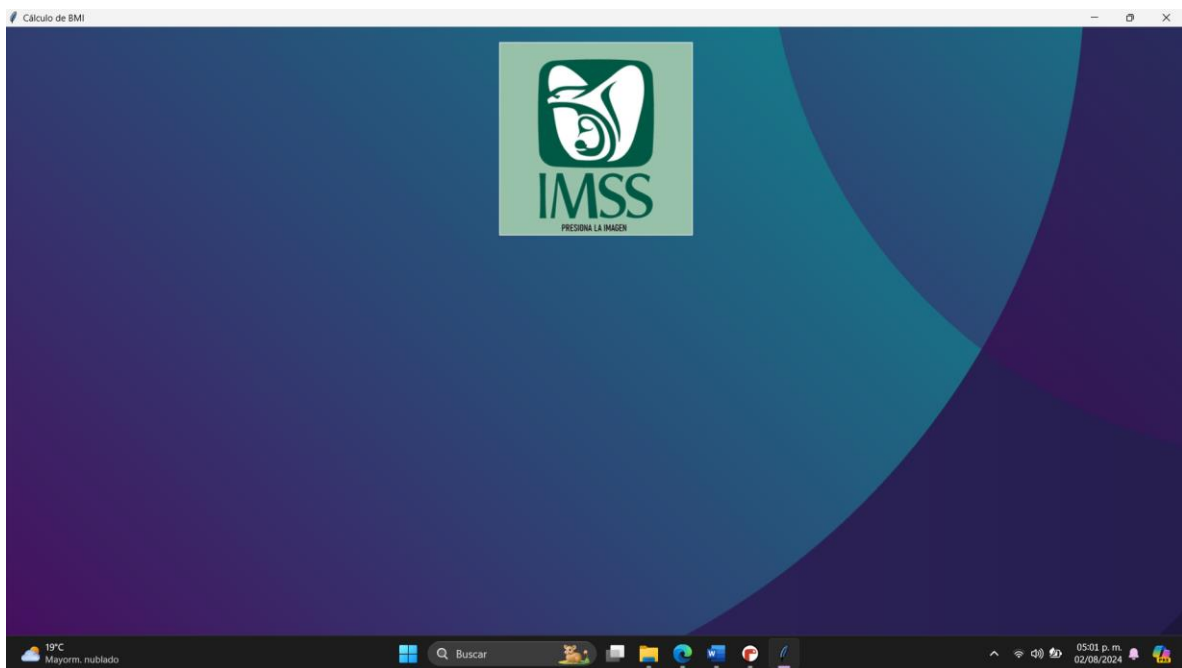
Ejecución de la Aplicación

- **Guardar el Código**: Guarda el código proporcionado en un archivo con extensión `.py`, por ejemplo, `bmi_calculator.py`.
- **Preparar Imágenes**: Coloca las imágenes necesarias en las rutas especificadas en el código:
 - `C:/Users/emili/Downloads/s.jpg` (imagen de fondo)
 - `C:/Users/emili/Downloads/C.png` (imagen del botón calcular)
 - `C:/Users/emili/Downloads/Calcular.png` (imagen del botón calcular al pasar el ratón)
 - `C:/Users/emili/OneDrive/Imágenes/Capturas de pantalla/Tablalmc.png` (imagen de la tabla de IMC)

- C:/Users/emili/Downloads/info.png (imagen del botón información)
 - C:/Users/emili/Downloads/R.png (imagen del botón información al pasar el ratón)
 - C:/Users/emili/Downloads/Imss.png (imagen del botón principal)
- **Ejecutar la Aplicación:** Navega hasta la carpeta donde guardaste el archivo bmi_calculator.py y ejecuta el siguiente comando en la terminal o símbolo del sistema:
- ```
python bmi_calculator.py
```
- O también en alguna de las aplicaciones como Mu o Visual studio ya con el programa guardado.

## Uso de la Aplicación

- **Pantalla Principal:** Al ejecutar la aplicación, verás la pantalla principal con un botón de imagen. Haz clic en el botón para abrir la segunda ventana.



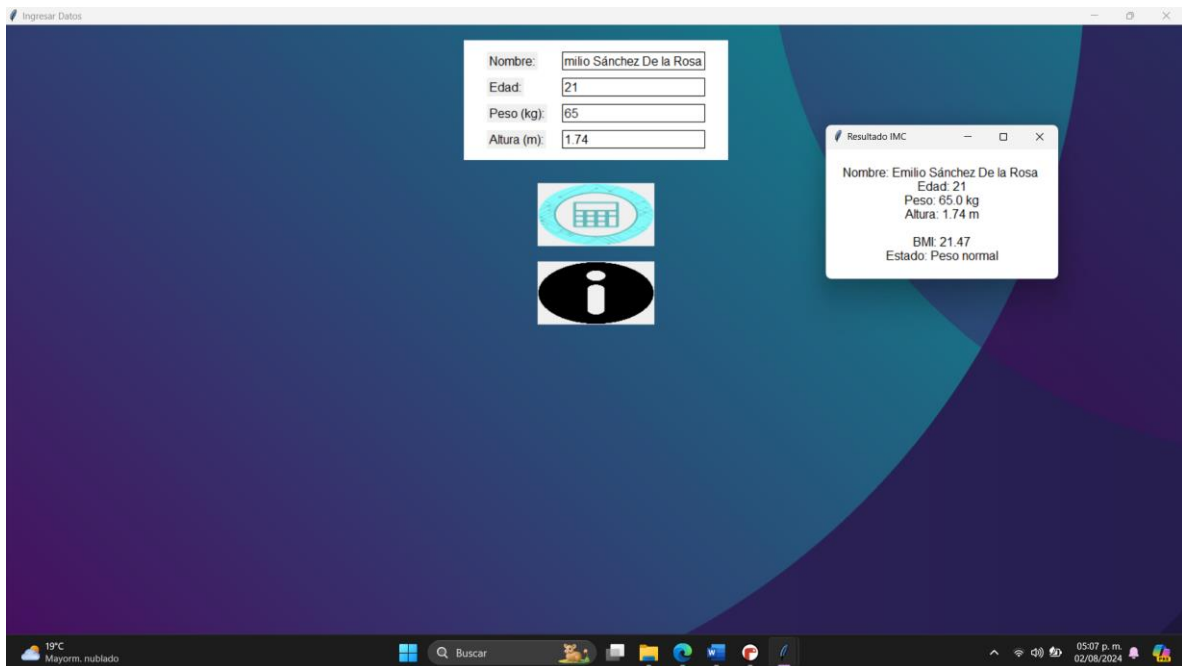
- **Ventana de Ingreso de Datos:** En la segunda ventana, podrás ingresar tu nombre, edad, peso (kg) y altura (m).

The screenshot shows a web browser window with the title 'Ingresar Datos'. The background is a dark blue and purple gradient. In the upper center, there is a white rectangular form with four input fields labeled 'Nombre:', 'Edad:', 'Peso (kg):', and 'Altura (m):'. Below the form are two icons: a light blue calculator icon and a black circular icon with a white lowercase 'i' (information). The Windows taskbar is visible at the bottom, showing the date and time as 05:03 p.m. on 02/08/2024.

- **Nombre:** Ingresa tu nombre.
- **Edad:** Ingresa tu edad.
- **Peso (kg):** Ingresa tu peso en kilogramos.
- **Altura (m):** Ingresa tu altura en metros.

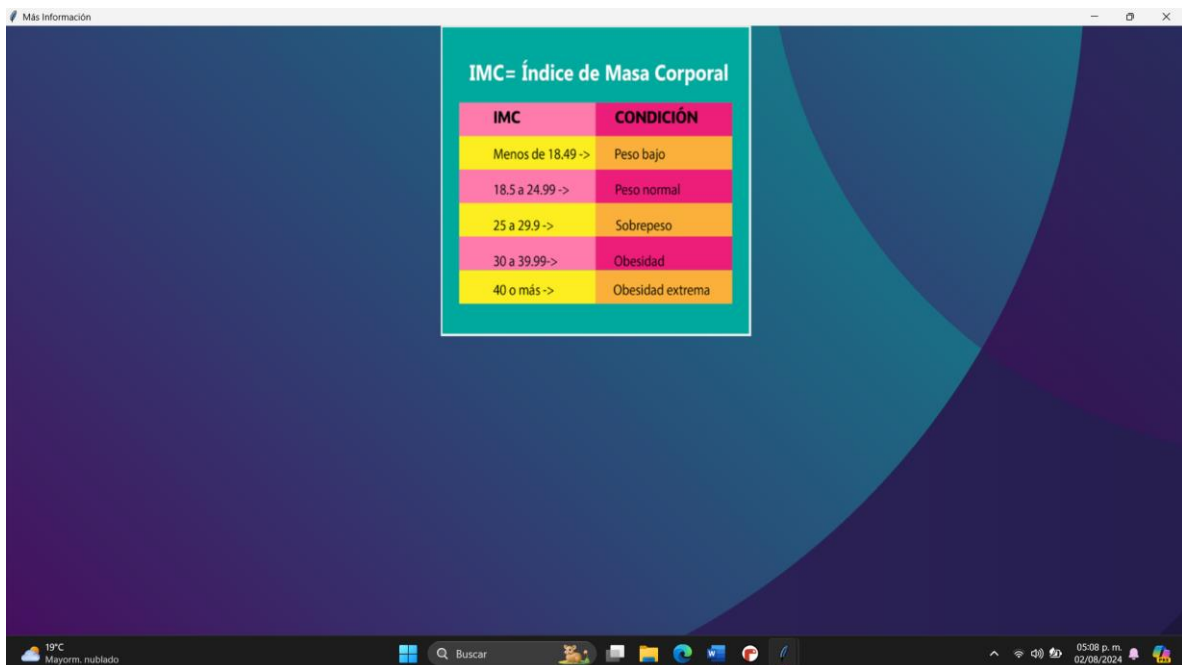
Después de ingresar los datos, haz clic en el botón "Calcular" para calcular tu BMI.

- **Resultado del BMI:** Después de hacer clic en "Calcular", se abrirá una nueva ventana mostrando tu nombre, edad, peso, altura, BMI calculado y el estado del BMI.



- **Más Información:** En la ventana de ingreso de datos, hay un botón de "Información". Haz clic en este botón para abrir una nueva ventana que muestra una tabla con información adicional sobre los rangos de BMI.

![[Más Información](C:/Users/emili/OneDrive/Imágenes/Capturas de pantalla/Tablalmc.png)]



## Errores Comunes

- **Error en los Valores:** Si ingresas valores no válidos para peso o altura, aparecerá un mensaje de error solicitando que ingreses valores válidos.
- **Imágenes no Encontradas:** Asegúrate de que las imágenes estén en las rutas especificadas. Si una imagen no se carga, verifica la ruta y asegúrate de que el archivo existe.

## Convertir la Aplicación de Cálculo de BMI en un Archivo .exe

Este manual describe cómo convertir el código de Python de la aplicación de cálculo de BMI en un archivo ejecutable (.exe) utilizando la herramienta pyinstaller.

### Requisitos del Sistema

- Sistema operativo: Windows
- Python 3.x instalado
- Librería pyinstaller instalada

### Instalación de PyInstaller

1. **Instalar Python:** Asegúrate de tener Python 3.x instalado en tu sistema. Puedes descargarlo desde [python.org](https://python.org).
2. **Instalar PyInstaller:** Abre la terminal o el símbolo del sistema y ejecuta:  

```
pip install pyinstaller
```

### Convertir el Código a un Archivo .exe

1. **Abrir la Terminal:** Navega hasta la carpeta que contiene tu archivo bmi\_calculator.py:

```
cd path\to\bmi_calculator
```

2. **Ejecutar PyInstaller:** Ejecuta el siguiente comando para crear el archivo .exe:

```
pyinstaller --onefile --windowed --add-data "images;images" bmi_calculator.py
```



- `--onefile`: Crea un único archivo ejecutable.
- `--windowed`: Ejecuta la aplicación sin una ventana de consola.
- `--add-data "images;images"`: Incluye la carpeta de imágenes en el ejecutable (la sintaxis puede variar en diferentes sistemas operativos, asegúrate de usar `:` en lugar de `;` en sistemas basados en Unix).

3. **Verificar el Resultado:** Después de que PyInstaller termine, encontrarás el archivo ejecutable en la carpeta `dist` dentro de tu proyecto:

```
bmi_calculator/
├── dist/
│ └── bmi_calculator.exe
├── build/
├── images/
└── bmi_calculator.py
```

## Ajustes Adicionales

1. **Modificar el Archivo spec:** Si necesitas ajustar configuraciones adicionales, puedes modificar el archivo `bmi_calculator.spec` generado por PyInstaller.
2. **Incluir Dependencias Adicionales:** Asegúrate de que todas las dependencias y archivos necesarios están incluidos en el ejecutable. Esto incluye cualquier biblioteca de terceros o archivos de recursos.

## Ejecución del Archivo .exe

1. **Ejecutar el Archivo .exe:** Navega a la carpeta `dist` y ejecuta el archivo `bmi_calculator.exe` haciendo doble clic sobre él.
2. **Distribuir el Archivo .exe:** Puedes distribuir el archivo `bmi_calculator.exe` junto con la carpeta `images` si es necesario para que otros usuarios puedan ejecutar la aplicación en sus sistemas.

## Errores Comunes y Soluciones

1. **Imágenes no Encontradas:** Si el ejecutable no encuentra las imágenes, asegúrate de que la ruta especificada en `--add-data` sea correcta y que las imágenes estén en la misma carpeta que el ejecutable.

2. **Problemas de Librerías:** Si faltan librerías, asegúrate de que todas las dependencias estén correctamente instaladas y configuradas en tu entorno de Python.

## Diagramas de Flujo:

