



TRABAJO PRÁCTICO DE PROGRAMACION VI

- **Nombre y apellido:** Francisco Ariel Galeano Ruiz Díaz
- **Profesor:** Ricardo Alberto Maidana Giménez
- **Materia:** Programación VI
- **Fecha:** 22/10/2024

CAACUPE, PARAGUAY
2024



INDICE

INTRODUCCION	3
APP LISTA DE COMPRAS.....	4
CODIGO FUNCIONAL DEL PROYECTO.	4
CAPTURAS DE PANTALLA.	7
CONCLUSION	12



INTRODUCCION

Este código implementa una aplicación de lista de compras utilizando la librería Flet, una plataforma basada en Python para crear interfaces gráficas (GUI) para aplicaciones web y de escritorio. La aplicación permite al usuario añadir tareas a una lista de compras, así como modificarlas o eliminarlas según sea necesario. El enfoque principal de este código es proporcionar una interfaz intuitiva donde los usuarios puedan interactuar de manera sencilla para gestionar las tareas de su lista de compras.

La estructura de la aplicación está diseñada para ser simple y funcional. Se incluyen componentes básicos como campos de texto para agregar nuevas tareas, botones para gestionar las tareas existentes y una lista que organiza visualmente cada tarea con opciones para modificar o eliminar. Además, se presenta una cabecera personalizada con un logo e información de bienvenida, lo que ofrece una experiencia más atractiva y personalizada para el usuario.



APP LISTA DE COMPRAS.

CODIGO FUNCIONAL DEL PROYECTO.

```
import flet as ft

def main(page: ft.Page):
    # Definir ancho y alto de la ventana
    page.window_width = 600 # ancho de la ventana
    page.window_height = 500 # alto de la ventana
    page.title = "Lista de Compras"

    def add_clicked(e):
        if new_task.value: # Solo agrega si el campo no está
vacío
            task = create_task(new_task.value)
            task_list.controls.append(task)
            new_task.value = ""
            new_task.focus()
            page.update()

    def create_task(task_text):
        # Crear una fila que contenga la tarea, el botón de
modificar y el botón de eliminar
        checkbox = ft.Checkbox(label=task_text)
        edit_button = ft.ElevatedButton("Modificar",
on_click=lambda e: edit_task(checkbox))
        delete_button = ft.ElevatedButton("Eliminar",
on_click=lambda e: delete_task(checkbox))

        return ft.Row([checkbox, edit_button, delete_button],
alignment="spaceBetween")

    def edit_task(checkbox):
        # Permitir modificar el texto de la tarea
        new_value = ft.TextField(value=checkbox.label, width=300)
        save_button = ft.ElevatedButton("Guardar",
on_click=lambda e: save_task(checkbox, new_value))
```



```
# Reemplazar el contenido de la fila con el campo de
texto y el botón de guardar
task_row = checkbox.parent
task_row.controls.clear()
task_row.controls.append(new_value)
task_row.controls.append(save_button)
page.update()

def save_task(checkbox, new_value):
    # Guardar el nuevo valor en el checkbox y restaurar los
    botones de modificar y eliminar
    checkbox.label = new_value.value
    task_row = new_value.parent
    task_row.controls.clear()
    task_row.controls.append(checkbox)
    task_row.controls.append(ft.ElevatedButton("Modificar",
on_click=lambda e: edit_task(checkbox)))
    task_row.controls.append(ft.ElevatedButton("Eliminar",
on_click=lambda e: delete_task(checkbox)))
    page.update()

def delete_task(checkbox):
    # Eliminar la tarea de la lista
    task_list.controls.remove(checkbox.parent)
    page.update()

# Creamos los componentes de la interfaz de usuario
new_task = ft.TextField(hint_text="¿Qué necesitas comprar?",
width=300)
task_list = ft.Column()

# Creamos una cabecera con el logo e imagen (texto o imagen)
header_text = ft.Text("Bienvenidos a la App de Lista de
Compras", size=20, weight=ft.FontWeight.BOLD)
logo = ft.Image(src="C:\\Users\\TOSHIBA\\P6 Python -
Francisco Galeano\\logo.png", width=150, height=150)
```



```
# Organizamos cabecera en una columna y centramos
header = ft.Column([
    logo,
    header_text
], alignment="center") # Alineación centrada

# Añadimos elementos a la aplicación
page.add(
    header, # Para que muestre el texto primero
    ft.Divider(height=20), # Agrega un divisor para separar
    el logo de la sección
    ft.Row([new_task, ft.ElevatedButton("Agregar",
on_click=add_clicked)]),
    task_list
)

# Establecer el color de fondo a rojo al inicio
page.bgcolor = "blue"
page.update()

ft.app(main)
```



CAPTURAS DE PANTALLA.

APP LISTA DE COMPRAS.

Al ejecutar se observa que funciona correctamente sin errores el App de lista de compras en la imagen se visualiza que tiene un logo en la parte de arriba y abajo le agregue “Bienvenidos a la App de Lista de Compras” como se ve ahí, más abajo en caja de texto le pregunte que se necesita comprar y a lado le presionamos agregar para que agregue lo que queremos y con un fondo de color blue.





Bienvenidos a la App de Lista de Compras

Agregar

<input type="checkbox"/> CARGADOR DE PC	<button>Modificar</button>	<button>Eliminar</button>
<input type="checkbox"/> MOUSE	<button>Modificar</button>	<button>Eliminar</button>
<input type="checkbox"/> TECLADO DE PC	<button>Modificar</button>	<button>Eliminar</button>
<input type="checkbox"/> CPU	<button>Modificar</button>	<button>Eliminar</button>

En esta imagen se observa que le agregue cuatro que necesito comprar como un Cargador de Pc, Mouse, Teclado de Pc y CPU. Se visualiza también los botones de modificar y eliminar.



Bienvenidos a la App de Lista de Compras

Agregar

☐ CARGADOR DE PC

Modificar

Eliminar

Guardar

☐ TECLADO DE PC

Modificar

Eliminar

☐ CPU

Modificar

Eliminar

En esta imagen se observa con el botón Modificar funciona correctamente acá modifiko en la parte mouse le modifique con Mouse Para PC.



Bienvenidos a la App de Lista de Compras

¿Qué necesitas comprar?

Agregar

☐ CARGADOR DE PC

Modificar

Eliminar

☐ MOUSE PARA PC

Modificar

Eliminar

☐ TECLADO DE PC

Modificar

Eliminar

☐ CPU

Modificar

Eliminar

Acá se ve perfectamente la modificación ya guardada que funciona perfectamente el botón Modificar.



Bienvenidos a la App de Lista de Compras

Agregar

☐ CARGADOR DE PC

Modificar

Eliminar

☐ MOUSE PARA PC

Modificar

Eliminar

☐ TECLADO DE PC

Modificar

Eliminar

En esta imagen utilicé el botón Eliminar se ve perfectamente que funciona ya se borró la CPU de la lista de Compras.

Funciona perfectamente mi código del App de la Lista de Compras.

Link de Acceso Github.

https://github.com/FranciscoGaleano551/Trabajo_Practico_ProgramacionVI



CONCLUSION

El código desarrollado ofrece una solución eficaz y fácil de usar para la gestión de listas de compras. Con características como la capacidad de agregar, modificar y eliminar elementos, la aplicación facilita al usuario organizar sus compras de manera eficiente. El uso de la librería Flet simplifica el proceso de creación de interfaces gráficas en Python, permitiendo una rápida implementación de elementos visuales interactivos.

La aplicación también incluye una sección de cabecera con un logo personalizado y un mensaje de bienvenida, mejorando la experiencia del usuario y haciendo la aplicación más atractiva. Esto la convierte en una opción versátil para aquellos que buscan una herramienta simple y efectiva para gestionar sus compras.

La aplicación nos brinda de cómo podemos manejar con una lista de compras de manera ordenada, de manera fácil y sencilla de usar es una herramienta muy efectiva para las compras.