

## HITO 2 DEL 1º TRIMESTRE DE Programación

Yordan Rosenov Matov  
20/11/2024

## ÍNDICE

Fase 1. ....	3
Fase 2. ....	5

<https://github.com/CR1PTAS/programaci-n.git>

## Fase 1.

En este primer paso, se van a definir los pasos necesarios para resolver el registro de un nuevo cliente y cómo realizar una compra, así como diseñar para cada uno de ellos el diagrama de flujo correspondiente. En este primer paso, no es necesario desarrollar la aplicación en ningún lenguaje de programación. Se trata en todo caso, de analizar la

### HITO INDIVIDUAL PROGRAMACIÓN

Mostrar "Registro de Cliente"

identificador <- Solicitar entrada al usuario

Si identificador en clientes:

Mostrar "Error: El cliente ya está registrado."

Salir

nombre <- Solicitar entrada al usuario ("Ingrese el nombre: ")

telefono <- Solicitar entrada al usuario ("Ingrese el teléfono: ")

direccion <- Solicitar entrada al usuario ("Ingrese la dirección: ")

clientes[identificador] = {"nombre": nombre, "telefono": telefono, "direccion":  
direccion}

Mostrar "Cliente registrado exitosamente."

Mostrar "Lista de Clientes Registrados:"

Para cada identificador,

Mostrar identificador, Nombre, Teléfono Dirección

Mostrar "Realizar una Compra"

identificador - Solicitar entrada al usuario

Si identificador no en clientes:

Mostrar "Error: Cliente no encontrado."

Salir

Mostrar "Productos disponibles:"

Para cada id\_producto

seleccion - Solicitar entrada al usuario

id\_productos - Convertir seleccion en lista de enteros

total = 0

Para cada id\_producto en id\_productos:

Si id\_producto en productos:

total += productos

Longitud de pedidos + 1  
pedidos= {"cliente": identificador, "productos": items, "total": total}  
Mostrar "Compra realizada exitosamente. Número de pedido "

Mostrar "Seguimiento de Compra"  
numero\_pedido - Solicitar entrada como entero  
Si numero\_pedido no en pedidos:  
    Mostrar "Error: Pedido no encontrado."  
    Salir  
pedido = pedidos[numero\_pedido]  
cliente = clientes  
Mostrar Productos

Mientras Verdadero:  
    Mostrar - Menú Principal  
    Mostrar - 1. Registrar Cliente  
    Mostrar - 2. Visualizar Clientes  
    Mostrar - 3. Realizar Compra  
    Mostrar - 4. Seguimiento de Compra  
    Mostrar 5 - Salir  
    opcion - Solicitar entrada  
Si opcion == 1  
    Pedir datos de los clientes  
Si no:  
    Mostrar "Opción no válida. Intente de nuevo."

Si opcion = 2  
Mostrar clients

Si opcion = 3  
Mostrar la compra realizada

Si opcion = 4  
Mostrar la ruta de la compra realizada

Si opcion = 5  
Entonces Terminar

## Fase 2.

A continuación, nos dedicamos a la implementación de la aplicación. En este punto, sí es necesario utilizar Python para su desarrollo, usando las estructuras de datos que consideres más adecuadas. Debemos tener presente que no se nos pide realizar el aplicativo visual, únicamente la funcionalidad de cada caso. La selección de cada una de las funcionalidades se realizará mediante el uso de un menú.

```
# Estructuras de datos
clientes = {}
pedidos = {}
productos = {
    1: {"nombre": "Producto A", "precio": 10},
    2: {"nombre": "Producto B", "precio": 20},
    3: {"nombre": "Producto C", "precio": 30},
}

# Función para registrar cliente
def registrar_cliente():
    print("Registro de Cliente")
    identificador = input("Ingrese un identificador único (correo o ID): ")
    if identificador in clientes:
        print("Error: El cliente ya está registrado.")
        return
    nombre = input("Ingrese el nombre: ")
    telefono = input("Ingrese el teléfono: ")
    direccion = input("Ingrese la dirección: ")
    clientes[identificador] = {"nombre": nombre, "telefono": telefono, "direccion": direccion}
    print("Cliente registrado exitosamente.")

# Función para visualizar clientes
def visualizar_clientes():
    print("Lista de Clientes Registrados:")
    for identificador, datos in clientes.items():
        print(f"ID: {identificador}, Nombre: {datos['nombre']}, Teléfono: {datos['telefono']}, Dirección: {datos['direccion']}")

# Función para realizar una compra
def realizar_compra():
```

```

print("Realizar una Compra")
identificador = input("Ingrese el identificador del cliente: ")
if identificador not in clientes:
    print("Error: Cliente no encontrado.")
    return
print("Productos disponibles:")
for id_producto, info in productos.items():
    print(f"{id_producto}: {info['nombre']} - ${info['precio']}")
seleccion = input("Ingrese los IDs de los productos separados por
comas: ")
ids_productos = map(int, seleccion.split(","))
total = 0
items = []
for id_producto in ids_productos:
    if id_producto in productos:
        total += productos[id_producto]["precio"]
        items.append(productos[id_producto]["nombre"])
numero_pedido = len(pedidos) + 1
pedidos[numero_pedido] = {"cliente": identificador, "productos":
items, "total": total}
print(f"Compra realizada exitosamente. Número de pedido:
{numero_pedido}")

# Función para seguimiento de compra
def seguimiento_compra():
    print("Seguimiento de Compra")
    numero_pedido = int(input("Ingrese el número de pedido: "))
    if numero_pedido not in pedidos:
        print("Error: Pedido no encontrado.")
        return
    pedido = pedidos[numero_pedido]
    cliente = clientes[pedido["cliente"]]
    print(f"Cliente: {cliente['nombre']}, Teléfono: {cliente['telefono']},
Dirección: {cliente['direccion']}")
    print(f"Productos: {' , '.join(pedido['productos'])}, Total:
${pedido['total']}")

# Menú principal
def menu():
    while True:
        print("\n--- Menú Principal ---")
        print("1. Registrar Cliente")
        print("2. Visualizar Clientes")

```

```
print("3. Realizar Compra")
print("4. Seguimiento de Compra")
print("5. Salir")
opcion = input("Seleccione una opción: ")
if opcion == "1":
    registrar_cliente()
elif opcion == "2":
    visualizar_clientes()
elif opcion == "3":
    realizar_compra()
elif opcion == "4":
    seguimiento_compra()
elif opcion == "5":
    print("Saliendo del sistema.")
    break
else:
    print("Opción no válida. Intente de nuevo.")

# Ejecutar el menú
menu()
```

