

Tarea 06. Programa Cobro y Facturación de parqueo

Alejandra Magalí López Miranda, 201600085^{1,*}

¹Escuela de Ingeniería Mecánica Eléctrica, Facultad de ingeniería, Universidad de San Carlos, Guatemala.
(Dated: 20 de agosto de 2025)

I. INTRODUCCIÓN

El aumento de vehículos en las ciudades exige sistemas de parqueo más eficientes. Este proyecto desarrolla un programa en Python para calcular y facturar el servicio según el tiempo de uso, integrando ingreso y validación de datos, cálculo automático, generación de facturas y almacenamiento en archivos y en una base de datos PostgreSQL para un control confiable.

II. OBJETIVOS

A. General

- Desarrollar un sistema de cobro y facturación automatizado para parqueos, utilizando Python.

B. Específicos

- * Diseñar e implementar un programa que solicite y valide los datos de entrada del usuario.
- * Programar el cálculo automático del costo de estacionamiento aplicando las tarifas establecidas y generar facturas en un archivo de texto.
- * Incorporar mecanismos de control de errores y almacenamiento de datos en PostgreSQL y archivos de texto, garantizando la integridad y disponibilidad de la información.

III. MARCO TEÓRICO

A. Automatización de procesos administrativos

La automatización de procesos administrativos es simplemente usar la tecnología y la integración de sistemas y datos para realizar un proceso que antes se ejecutaba manualmente.

Sin embargo, es importante destacar que para poder hacer la automatización de procesos es necesario que el flujo de trabajo esté bien definido, es decir, tener claro quiénes intervienen en el proceso, cuál es la información que se maneja, qué clase de acciones se realizan y cuál es el resultado final.

B. Diagrama de flujo

Un diagrama de flujo en informática es una representación gráfica de un algoritmo o un proceso dentro del ámbito de la informática. Utiliza símbolos predefinidos y líneas de conexión para representar una secuencia de operaciones que conducen al resultado deseado.

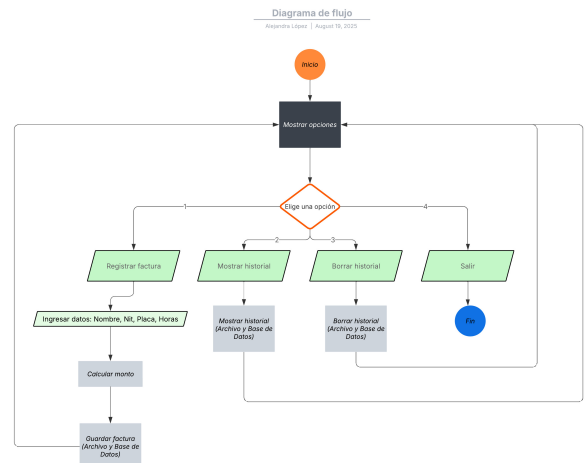
En los diagramas de flujo de informática se utilizan varios símbolos, cada uno con su significado único:

- Óvalo: Representa el inicio o final de un proceso.
- Rectángulo: Indica un proceso o acción.
- Diamante: Denota un punto de decisión.
- Paralelogramo: Significa entrada o salida.
- Flechas: muestran la dirección del flujo de trabajo.

IV. RESULTADOS

A. Diagrama de flujo

Figura 1: Diagrama de flujo



* 2816733570108@ingenieria.usac.edu.gt

B. Python

Figura 2: Creando base de datos 'parqueo'

```
Query Query History
1 CREATE DATABASE parqueo;
```

Figura 3: Registro y factura generada

```
===== SISTEMA DE PARQUEO =====
1. Registrar factura
2. Mostrar historial
3. Borrar historial
4. Salir
Seleccione una opción: 1
Ingrese nombre del cliente: Alejandra López
Ingrese NIT del cliente (9 dígitos): 114595747
Ingrese número de placa (P123456): P189CXT
Ingrese hora de entrada (HH:MM): 05:42
Ingrese hora de salida (HH:MM): 07:12

Factura generada: Cliente=Alejandra López, NIT=114595747, Placa=P189CXT, Tiempo=1:30:00, Horas facturadas=2, Monto=Q35

===== SISTEMA DE PARQUEO =====
1. Registrar factura
2. Mostrar historial
3. Borrar historial
4. Salir
Seleccione una opción: 1
```

Figura 4: Historial en archivo y base de datos

```
===== SISTEMA DE PARQUEO =====
1. Registrar factura
2. Mostrar historial
3. Borrar historial
4. Salir
Seleccione una opción: 2

Historial desde archivo:
- Alejandra López,114595747,P189CXT,2,35,1:30:00

Historial desde base de datos:
- Cliente: Alejandra López, NIT: 114595747, Placa: P189CXT, Tiempo: 1:30:00, Horas facturadas: 2, Monto: Q35
```

Figura 5: Tabla 'facturas' en base de datos

	id	nombre	nit	placa	horas	monto	tiempo
	[PK] integer	text	bigint	text	integer	integer	text
1	2	Alejandra López	114595747	P189CXT	2	35	1:30:00

Figura 6: Borrar historial y salir del programa

```
===== SISTEMA DE PARQUEO =====
1. Registrar factura
2. Mostrar historial
3. Borrar historial
4. Salir
Seleccione una opción: 3
Archivo de facturas borrado.
Historial de base de datos borrado.

===== SISTEMA DE PARQUEO =====
1. Registrar factura
2. Mostrar historial
3. Borrar historial
4. Salir
Seleccione una opción: 4
Saliendo del sistema. ¡Gracias!
```

Figura 7: Posibles errores

```
===== SISTEMA DE PARQUEO =====
1. Registrar factura
2. Mostrar historial
3. Borrar historial
4. Salir
Seleccione una opción: 1
Ingrese nombre del cliente: Daniela Morales
Ingrese NIT del cliente (9 dígitos): 123456
El NIT debe contener exactamente 9 dígitos numéricos.
Ingrese NIT del cliente (9 dígitos): 1kd256789
El NIT debe contener exactamente 9 dígitos numéricos.
Ingrese NIT del cliente (9 dígitos): 123
El NIT debe contener exactamente 9 dígitos numéricos.
Ingrese NIT del cliente (9 dígitos): 123456789
Ingrese número de placa (P123ABC): q123abc
La placa debe tener el formato P123ABC (7 caracteres, solo letras y números).
Ingrese número de placa (P123ABC): pklj245
La placa debe tener el formato P123ABC (7 caracteres, solo letras y números).
Ingrese número de placa (P123ABC): p-502-jpk
La placa debe tener el formato P123ABC (7 caracteres, solo letras y números).
Ingrese número de placa (P123ABC): p123abc
Ingrese hora de entrada (HH:MM): 0110
Formato inválido. Use HH:MM (ejemplo: 14:30)
Ingrese hora de entrada (HH:MM): 01:10
Ingrese hora de salida (HH:MM): 00:05
La hora de salida debe ser posterior a la hora de entrada.
```

V. CONCLUSIONES

- * El diagrama de flujo ayuda a visualizar la secuencia de acciones y decisiones del sistema de parqueo, desde la entrada de datos del cliente hasta el cálculo del monto y el registro en archivo o base de datos. Permite identificar claramente las validaciones, los bucles de reintento y la lógica de facturación, facilitando tanto la comprensión como el mantenimiento y la mejora del código.
- * El uso de def para definir funciones permitio modularizar el código, haciendo más fácil entender, probar y modificar cada parte del código de manera independiente.

* Se implementó un manejo de errores preventivo en cada entrada del usuario, NIT, placa, horas y tiempos. Gracias a esto, el programa no permite datos inválidos. Esto garantiza que los registros sean compatibles con la base de datos y evita problemas al momento de consultas o reportes. Esto garantiza que los registros sean compatibles con la base de datos y evita problemas al momento de consultas o

reportes.

VI. ANEXOS

Link de repositorio: <https://github.com/AlejandraMLM/980-Proyectos.git>

[1] ¿Qué es la automatización de procesos administrativos?. Disponible en: <https://ibaiscanbit.com/aplicaciones/automatizacion-de-procesos-administrativos/>

[2] Entienda a fondo los diagramas de flujo en informática. Disponible en: <https://boardmix.com/es/knowledge/flowchart-in-computer-science/>