

Sistema de administración para parqueadero

Descripción del Proyecto:

La **Alcaldía de Armenia** ha identificado un problema recurrente en los parqueaderos de la ciudad: la falta de un sistema unificado que permita administrar clientes, tarifas y pagos de manera eficiente. Muchos parqueaderos operan con registros manuales o software obsoleto, lo que genera pérdidas económicas, errores en el control de pagos y dificultades para ofrecer servicios personalizados a los clientes.

En respuesta a esta problemática, la alcaldía ha lanzado una **licitación pública** para desarrollar un software de gestión de parqueaderos, el cual será **gratuito para todos los parqueaderos registrados en la ciudad**. Este sistema permitirá a los propietarios administrar sus tarifas, gestionar clientes y vehículos, y ofrecer membresías con pagos anticipados.

El objetivo del sistema es optimizar la administración de parqueaderos, reduciendo tiempos de espera y mejorando la experiencia del usuario. A través de una solución basada en **Java** y con un diseño orientado a objetos, el sistema proporcionará una herramienta eficiente sin necesidad de bases de datos, utilizando estructuras en memoria para el almacenamiento de información.

Requisitos Funcionales

1 Gestión de Clientes

- ✓ **Añadir cliente:** Registrar nuevos clientes con datos como **nombre, cédula, teléfono y correo**.
- ✓ **Eliminar cliente:** Remover clientes del sistema.
- ✓ **Buscar cliente:** Localizar clientes por nombre, cédula o número de teléfono.
- ✓ **Actualizar cliente:** Modificar información del cliente.
- ✓ **Ver vehículos del cliente:** Mostrar todos los vehículos asociados a un cliente, y si estos tienen cobertura o ya venció su membresía.

2 Gestión de Vehículos

- ✓ **Registrar vehículo:** Asociar un vehículo a un cliente, registrando **placa, tipo de vehículo (Automóvil, Moto, Camión), color y modelo**. (El tipo de vehículo se debe manejar con herencia)
- ✓ **Buscar vehículo:** Filtrar por **placa, tipo de vehículo o cliente**.
- ✓ **Actualizar vehículo:** Modificar la información del vehículo.

3 Administración, tarifas y Membresías

- ✓ **Datos del parqueadero:** El administrador del parqueadero puede definir los datos del parqueadero como nombre, dirección, representante, teléfonos o medios de contacto. También puede actualizar esta información la cual es importante para la facturación.

- ✓ **Puestos disponibles:** El administrador definirá la cantidad de espacios para motos, automóviles y camiones disponibles.
- ✓ **Configurar tarifas:** El administrador del parqueadero puede definir los precios por tipo de vehículo.
- ✓ **Pago por Período:** Los clientes pueden pagar por **1 mes, 3 meses o 1 año máximo**.
- ✓ **Historial de Pagos:** Registro de pagos realizados por vehículos con o sin membresía.

4 Registro de Ingresos y Pagos

- ✓ **Vehículos temporales:** Los vehículos que solo ingresan por horas no se deben registrar, solo hacer su ingreso y salida.
- ✓ **Registrar Ingreso y Salida:** Control de tiempo que un vehículo permanece en el parqueadero.
- ✓ **Control de espacios disponibles:** Se debe tener control sobre la cantidad de espacios disponibles del parqueadero, cada que entre un vehículo este espacio es ocupado hasta que este salga. Los vehículos con membresía activa ocupan un espacio permanente, el cual no se podrá usar por otros vehículos del mismo tipo hasta que la membresía haya vencido. Adicional un vehículo no puede ingresar si no hay espacios disponibles para su vehículo.
- ✓ **Calcular Monto a Pagar:** Basado en la tarifa y las horas que lleva en el establecimiento.
- ✓ **Factura:** Generar factura con los datos del parqueadero y del vehículo, pago y fecha de ingreso y salida (para vehículos temporales) y fecha de inscripción y terminación de la membresía para vehículos registrado.

5 Reportes y Consultas

- ✓ **Historial de Vehículos por Cliente:** Mostrar todos los vehículos asociados a un cliente.
- ✓ **Total de Ingresos del Parqueadero:** Cálculo de ganancias por día, mes o año (lo transcurrido en el año).
- ✓ **Lista de Vehículos Actuales:** Vehículos que están dentro del parqueadero en un momento determinado.
- ✓ **Clientes con Membresías Activas:** Consultar usuarios con suscripciones vigentes y próximas a vencer.

Requisitos Técnicos

1 Diseño Orientado a Objetos (POO en Java)

- ✦ **Clases y Herencia:**
 - ◆ **Cliente:** Clase base con atributos como nombre, cédula y contacto.
 - ◆ **Vehículo:** Asociado a un cliente, con atributos como placa, tipo y color.
 - ◆ **Parqueadero:** Administra clientes, vehículos y pagos.
- ✦ **Polimorfismo:** Métodos en **Membresia** para calcular tarifas según el tipo de cliente.
- ✦ **Encapsulación:** Uso de private con **getters y setters**.

2 Estructura del Proyecto

✦ Paquetes organizados:

- ♦ model → Clases de datos (Cliente, Vehiculo, Membresia).
- ♦ service → Lógica de negocio (ParqueaderoService, PagoService).
- ♦ controller → Interfaz de usuario en consola.

✦ Uso de ArrayList para almacenamiento en memoria.

✦ Entrada y salida con JOptionPane (se dará bonificación por uso de interfaz gráfica)

3 Pruebas Automatizadas

✦ JUnit para validar:

- ♦ Registro de clientes y vehículos.
- ♦ Cálculo de tarifas y pagos.
- ♦ Vencimiento de membresías.

4 Buenas Prácticas de Desarrollo

✦ Documentación: Comentarios en código y manual de usuario.

✦ Control de versiones con Git.

✦ Diagrama de clases previo al desarrollo.

✦ Conclusión

Este sistema de gestión de parqueaderos, desarrollado en **Java sin base de datos**, representa una solución innovadora para modernizar la administración de parqueaderos en **Armenia**. Con funcionalidades como membresías, gestión de clientes y tarifas personalizables, el software permitirá optimizar el negocio de los parqueaderos mientras mejora la experiencia de los usuarios.

