

# MANUAL DE CÓDIGO - PARKPLUS

## 1. Introducción

Este manual describe la estructura interna del código Java utilizado para desarrollar el sistema ParkPlus. Incluye explicación de clases, subclases, ventanas (JFrame), consultas SQL y lógica principal de operaciones como ingreso, salida, reingreso y reportes.

## 2. Estructura del Proyecto

El proyecto está organizado en los siguientes paquetes:

- **ClaseVentanas:** Contiene todas las interfaces gráficas (JFrame).
- **SubClases:** Contiene clases lógicas que interactúan con la BD.
- **claseprin:** Contiene la conexión a la base de datos y la ventana principal.

## 3. Conexión a Base de Datos (dbConnection)

La clase dbConnection gestiona la conexión mediante JDBC.

Incluye:

- URL de conexión.
- Usuario y contraseña.
- Método estático conectar() que retorna un objeto Connection.

## 4. SubClases Principales

### 4.1 SubTicket

Responsable de:

- Crear tickets.
- Registrar salidas temporales.
- Obtener información de tickets activos.

### 4.2 SubParqueo

Encargado de:

- Asignar spots según el tipo de vehículo.
- Cambiar estados de los spots (FREE, OCCUPIED, TEMPORAL).

### 4.3 SubReingreso

Contiene la lógica de validación para permitir o negar reingresos dependiendo del modo (FLAT / VARIABLE) y tiempo fuera.

### 4.4 SubReportes

- Obtiene todos los tickets del día.
- Calcula el total recaudado.

## 5. Ventanas (JFrame)

### 5.1 PagoT y PagoV

Ventanas para registrar el tipo de pago. Ambas reciben:

- Placa
- Total a pagar
- Modo seleccionado

Llaman a SubTicket para crear el ticket correspondiente.

### 5.2 Reingreso

- Recibe placa.

- Usa SubReingreso para validar la entrada.
- Determina si aplica reingreso o no.

### **5.3 Reportes**

Genera una tabla con los tickets del día y suma total.  
Usa DefaultTableModel y la clase SubReportes.

### **5.4 Salidas**

Administra:

- Búsqueda de ticket activo.
- Cálculo de montos VARIABLE y FLAT.
- Procesar salida definitiva o temporal.

### **5.5 VParqueo**

Genera el mapa visual del parqueo usando JLabel y coloración por estado:

- Verde = Libre
- Rojo = Ocupado
- Amarillo = Temporal

### **6. Consultas SQL Principales**

- SELECT de tickets activos.
- UPDATE de salidas.
- UPDATE de estados de spots.
- INSERT de tickets nuevos.

### **7. Lógica del Modo FLAT y VARIABLE**

- FLAT: Monto fijo (Q15).
- VARIABLE: Q2 por hora. Se calcula en tiempo real.

### **8. Manejo de Fechas y Timestamps**

Todas las operaciones usan java.sql.Timestamp para garantizar precisión en horas, minutos y segundos.

### **9. Conclusión**

El sistema está modularizado, cada clase cumple una única función y las ventanas solo se encargan de la interacción del usuario.