**UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA**

**FACULTAD DE INGENIERÍA EN SISTEMAS DE INFORMACIÓN**

**CURSO: Estadística II**

**Ing. Miguel catalán**



**Proyecto de semestre**

**Víctor Alexander Siquiej Chajon 7590-23-7284**

**SAN JUAN SACATEPÉQUEZ, 24-05-2025.**

**índice**

[MANUAL TÉCNICO DEL SISTEMA DE TICKETS 3](#_Toc199022831)

[1. Descripción General del Sistema 3](#_Toc199022832)

[3. Estructura del Proyecto 3](#_Toc199022833)

[4. Configuración de la Base de Datos PostgreSQL 3](#_Toc199022834)

[5. Configuración de Conexión JDBC 4](#_Toc199022835)

[6. Compilación y Ejecución 4](#_Toc199022836)

[7. Mantenimiento 4](#_Toc199022837)

[Diagrama UML 5](#_Toc199022838)

[DIAGRAMA ENTIDAD - REALACION 6](#_Toc199022839)

# MANUAL TÉCNICO DEL SISTEMA DE TICKETS

## 1. Descripción General del Sistema

El sistema de tickets es una aplicación de escritorio desarrollada en Java SE con Swing para la interfaz gráfica de usuario. Permite a los usuarios finales crear tickets de soporte, los cuales pueden ser gestionados por técnicos asignados. Un administrador puede visualizar los tickets, usuarios, técnicos, y departamentos involucrados. Toda la información se almacena en una base de datos PostgreSQL, utilizando JDBC para la comunicación entre Java y PostgreSQL.

**2**. Requisitos del Sistema

*Hardware:*

* Procesador: Intel Core i3 o superior
* Memoria RAM: 4 GB o más
* Almacenamiento: 200 MB libres

*Software:*

* Java Development Kit (JDK) 8 o superior
* NetBeans IDE 12 o superior (u otro IDE compatible con Java)
* PostgreSQL 13 o superior
* pgAdmin 4 para administración de la base de datos

## 3. Estructura del Proyecto

El proyecto está organizado en los siguientes paquetes:

* modelo: contiene las clases como UsuarioFinal, Técnico, Administrador, Departamento, Ticket, EstadoTicket, y Rol.
* dao: contiene los CRUDs (DAO) de cada entidad.
* interfaz: contiene javafx de la interfaz gráfica (VentanaPrincipal, VentanaCrearUsuario, etc.).
* util: contiene la clase Conexion que se encarga de conectar con la base de datos PostgreSQL.

## 4. Configuración de la Base de Datos PostgreSQL

1. Crear la base de datos:

CREATE DATABASE javadb;

1. Crear tablas necesarias, por ejemplo:

CREATE TABLE usuariosfinales (

id SERIAL PRIMARY KEY,

nombre VARCHAR(100),

usuario VARCHAR(50) UNIQUE,

contrasena VARCHAR(50)

);

1. Repetir para técnicos, tickets, departamentos, roles, etc., usando los mismos atributos que en las clases Java.
2. Verifica la conexión con la clase Conexion.java.

## 5. Configuración de Conexión JDBC

Clase Conexion.java:

prívate final String HOST = "localhost";

prívate final String PUERTO = "5432";

private final String DB = "javadb";

private final String USER = "postgres";

private final String PASSWORD = "tu\_contraseña";

Usa Class.forName("org.postgresql.Driver") y DriverManager.getConnection para establecer la conexión.

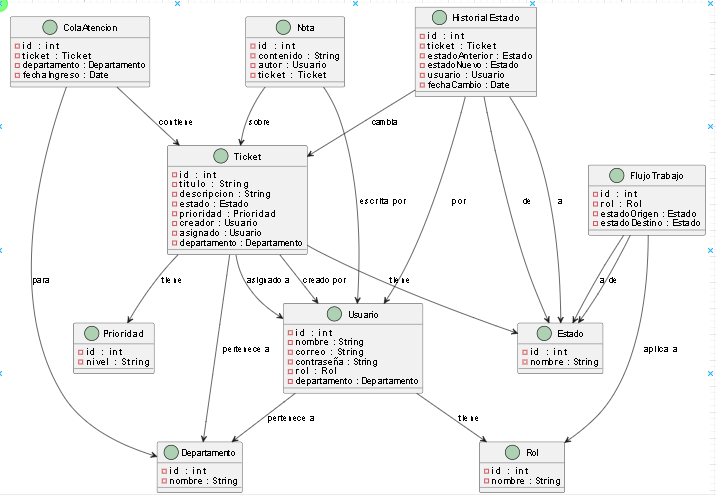
## 6. Compilación y Ejecución

1. Importar el proyecto en NetBeans.
2. Agregar la librería del conector PostgreSQL (archivo .jar).
3. Ejecutar la clase VentanaPrincipal.java o la clase main del sistema.

## 7. Mantenimiento

* Asegurar que las tablas y campos de la base de datos coincidan con los atributos de las clases.
* Verificar las rutas de los paquetes.
* Validar la conexión a la base de datos ante cualquier fallo.
* Revisar los controladores JDBC y sus versiones.

# Diagrama UML



# DIAGRAMA ENTIDAD – REALACION

