

## Script de Inicialización de Base de Datos MariaDB

### Esquema de Base de Datos para Sistema IoT

#### Propósito General

Este script SQL constituye la base de datos completa para el sistema de monitoreo IoT, diseñando una estructura robusta y escalable que soporta autenticación de usuarios, gestión de dispositivos, almacenamiento de datos de sensores, y un sistema granular de permisos y roles.

#### Estructura de la Base de Datos

##### Creación de Base de Datos

El script inicia creando la base de datos `silodb` con codificación UTF8MB4, asegurando soporte para caracteres especiales y emojis. Se incluye también la creación de `npmdb` para el Nginx Proxy Manager, demostrando una arquitectura integrada.

##### Tablas de Gestión de Usuarios

**usuarios:** Almacena información completa de usuarios autenticados vía Google OAuth, incluyendo identificador único de Google, email, nombre, URL de foto, y campos de auditoría como fechas de creación, actualización y último acceso.

**usuarios\_google:** Tabla simplificada específicamente para el sistema de autenticación, conteniendo los flags `admin` y `action` que determinan los privilegios del usuario en el sistema.

##### Sistema de Roles y Permisos

**roles:** Define los tres roles base del sistema: Administrador (acceso completo), Operador (control operacional), y Visualizador (solo lectura).

**permisos:** Catálogo granular de permisos que especifican acciones sobre recursos específicos, permitiendo un control de acceso detallado.

**permisos\_roles:** Tabla de unión que asigna permisos a roles, estableciendo las capacidades de cada rol en el sistema.

**usuarios\_roles:** Gestiona la asignación de roles a usuarios, incluyendo fechas de expiración y auditoría de quién realizó la asignación.

##### Gestión de Sesiones y Auditoría

**sesiones\_usuarios:** Almacena tokens de sesión JWT con información de conexión como IP y agente de usuario, permitiendo gestión activa de sesiones.

logs\_actividad: Registro completo de actividades del sistema para auditoría y troubleshooting, almacenando detalles en formato JSON.

### Modelo de Dispositivos y Datos IoT

dispositivos: Catálogo de dispositivos IoT con información de identificación, ubicación, estado de conectividad, y metadatos extensibles.

datos\_sensores: Almacenamiento eficiente de lecturas de sensores con soporte para diferentes tipos de datos, unidades de medida, y metadatos adicionales.

### Gestión de Proyectos

proyectos: Permite organizar el trabajo en proyectos independientes con diferentes estados y responsables.

usuarios\_proyectos: Asigna usuarios a proyectos con roles específicos dentro de cada proyecto (administrador, operador, visualizador).

### Configuración y MQTT

configuraciones\_sistema: Almacenamiento centralizado de configuraciones del sistema con soporte para diferentes tipos de datos.

mqtt\_topics: Registro de tópicos MQTT utilizados en el sistema, con metadatos descriptivos y asociación a dispositivos.

### Inserción de Datos Base

#### Roles y Permisos Predefinidos

El script inserta los tres roles fundamentales del sistema con sus descripciones correspondientes. Luego define un conjunto comprehensivo de permisos que cubren todas las funcionalidades del sistema, desde gestión de usuarios hasta control de dispositivos.

La asignación de permisos a roles establece una jerarquía clara:

Administrador: Acceso completo a todos los permisos

Operador: Permisos operacionales y de visualización

Visualizador: Solo permisos de lectura

#### Configuraciones del Sistema

Se insertan configuraciones por defecto esenciales para el funcionamiento del sistema, incluyendo nombre del sistema, version, parámetros de seguridad, configuración MQTT, y preferencias de backup y logging.

#### Tópicos MQTT

Se definen tópicos MQTT por defecto organizados por tipo de dato (temperatura, humedad, presión, comandos), incluyendo tópicos con wildcards para flexibilidad en la suscripción.

### Optimización con Índices

El script crea índices estratégicos en todas las tablas críticas para optimizar el rendimiento:

Índices en campos de búsqueda frecuente (email, Google ID)

Índices compuestos para consultas de datos temporales

Índices en claves foráneas para optimizar joins

Índices en campos de estado para filtrado rápido

### Vistas para Reportes

Se crean vistas especializadas que simplifican consultas comunes:

vista\_usuarios\_activos: Agrega información de usuarios con sus roles y proyectos asignados

vista\_dispositivos\_estado: Proporciona resumen del estado de dispositivos con métricas de datos

### Datos de Prueba

El script incluye datos de prueba que facilitan el desarrollo y testing:

Usuarios de prueba con diferentes roles

Asignaciones de roles y proyectos

Usuarios Google preconfigurados con diferentes niveles de acceso

### Gestión de Usuarios y Permisos de Base de Datos

Se crea un usuario dedicado silo\_user con permisos específicos para la base de datos principal y acceso limitado a la base de datos del proxy, siguiendo el principio de mínimo privilegio.

### Consideraciones de Diseño

#### Escalabilidad

Uso de JSON fields para metadatos extensibles sin modificar esquema

Índices optimizados para grandes volúmenes de datos

Particionamiento implícito mediante fechas en tablas de logs

## Seguridad

Campos de auditoría en todas las tablas críticas

Manejo de expiración de asignaciones de roles

Registro completo de actividades

## Flexibilidad

Soporte para múltiples tipos de sensores y unidades

Sistema de permisos granular y extensible

Metadatos configurables para dispositivos y tópicos

Este esquema de base de datos proporciona una base sólida para un sistema IoT empresarial, balanceando rendimiento, seguridad y flexibilidad mientras mantiene una estructura organizada y mantenible.