# Módulo de Gestión de Roles y Permisos de Usuario

## Propósito General

Este módulo implementa el sistema de resolución de roles de usuario basado en direcciones de email, integrando directamente con la base de datos para determinar los privilegios de acceso de cada usuario. El servicio sigue un modelo de seguridad de tres niveles: administrador, usuario con permisos de acción, y usuario de solo lectura, proporcionando una estructura granular de permisos para la aplicación.

## Arquitectura de Roles

El sistema define tres roles principales con diferentes niveles de acceso. El rol "admin" representa usuarios con privilegios completos del sistema, incluyendo acceso a configuraciones avanzadas y operaciones administrativas. El rol "action" designa usuarios con permisos intermedios que pueden realizar ciertas operaciones específicas pero no tienen acceso completo administrativo. El rol "readonly" es el nivel por defecto que permite solo consulta y visualización de datos sin capacidad de modificación.

## Flujo de Resolución de Roles

La función principal resolveRoleByEmail implementa el proceso completo de determinación de roles. El proceso comienza normalizando el email proporcionado a minúsculas para asegurar consistencia en las búsquedas, independientemente de cómo el usuario haya ingresado su dirección de email.

El servicio luego establece una conexión con la base de datos a través del pool de conexiones y ejecuta una consulta que busca el usuario en la tabla usuarios_google. La consulta verifica no solo la coincidencia del email sino también que el usuario esté marcado como activo en el sistema, añadiendo una capa adicional de control de acceso.

## Lógica de Asignación de Roles

Cuando se encuentra un usuario en la base de datos, el sistema evalúa secuencialmente sus privilegios. Primero verifica si el usuario tiene el flag admin establecido en TRUE, asignándole el rol de administrador si es así. Si no es administrador, verifica el flag action para asignar el rol de usuario con permisos de acción. Si ninguno de estos flags está activado, asigna el rol de solo lectura por defecto.

Este enfoque secuencial crea una jerarquía clara de permisos donde el rol de administrador tiene precedencia sobre todos los demás, seguido por el rol de acción, y finalmente el rol de solo lectura como fallback.

Registro Automático de Usuarios

Un aspecto importante del sistema es su capacidad de auto-registro de usuarios. Cuando un usuario se autentica por primera vez y no existe en la base de datos, el servicio automáticamente crea un nuevo registro en la tabla usuarios_google con los flags admin y action establecidos en FALSE, efectivamente asignándole el rol de solo lectura.

Este mecanismo de auto-provisión simplifica significativamente la administración de usuarios, ya que no requiere intervención manual para agregar nuevos usuarios al sistema. El registro automático se ejecuta de manera transparente durante el primer acceso del usuario.

Manejo de Errores y Estrategia de Fallback

El servicio implementa un robusto manejo de errores que garantiza la continuidad del servicio incluso en situaciones excepcionales. Si ocurre cualquier error durante el proceso de consulta a la base de datos o durante la creación de usuarios, el sistema captura la excepción, la registra para propósitos de diagnóstico, y retorna el rol "readonly" como fallback seguro.

Esta estrategia de "fail-secure" asegura que en caso de problemas técnicos, los usuarios no obtengan privilegios elevados accidentalmente, manteniendo la seguridad del sistema. El logging de errores permite a los administradores identificar y resolver problemas subyacentes.

Integración con la Base de Datos

El módulo utiliza la tabla usuarios_google que está diseñada específicamente para gestionar usuarios autenticados a través de Google OAuth. La estructura de la tabla incluye el campo mail como identificador único, los flags booleanos admin y action para control de permisos, y el flag activo para habilitar o deshabilitar usuarios sin eliminarlos del sistema.

El uso de transacciones implícitas a través del pool de conexiones asegura la integridad de las operaciones de base de datos, mientras que la liberación explícita de conexiones previene fugas de recursos.

Consideraciones de Seguridad

El sistema sigue el principio de menor privilegio, asignando por defecto el rol más restrictivo a los nuevos usuarios. Los privilegios elevados deben ser asignados

explícitamente por un administrador modificando directamente los flags en la base de datos.

La normalización de emails a minúsculas previene posibles bypass de seguridad mediante el uso de diferentes combinaciones de mayúsculas y minúsculas en la misma dirección de email.

Escalabilidad y Mantenimiento

La arquitectura del servicio permite fácil extensión para incluir nuevos roles o modificar la lógica de asignación sin afectar el código cliente. El uso de consultas parametrizadas previene vulnerabilidades de inyección SQL y mejora el rendimiento mediante la reutilización de planes de ejecución.

Para entornos con muchos usuarios, podría considerarse la implementación de caching de roles para reducir la frecuencia de consultas a la base de datos, aunque la implementación actual prioriza la consistencia inmediata de los permisos.

Flujo de Integración con Autenticación

Este servicio se integra directamente con el proceso de autenticación de Google OAuth. Después de que un usuario se autentica exitosamente con Google, el sistema utiliza su email para consultar este servicio y determinar los permisos específicos dentro de la aplicación. El rol determinado se incluye entonces en el token JWT propio de la aplicación para su uso en autorizaciones subsiguientes.

Esta separación entre autenticación (verificar identidad) y autorización (determinar permisos) sigue las mejores prácticas de seguridad y permite una gestión flexible de permisos independiente del proveedor de identidad.

El servicio proporciona así un mecanismo robusto, seguro y auto-gestionado para el control de acceso basado en roles, fundamental para mantener la seguridad y la organización de los privilegios en una aplicación multi-usuario.