

Módulo de Conexión a Bases de Datos

Propósito General

Este módulo se encarga de gestionar las conexiones a múltiples sistemas de bases de datos utilizados en la aplicación. La arquitectura implementa una solución multi-base de datos donde cada sistema está especializado para un tipo específico de datos y casos de uso. Actualmente se integran MariaDB para datos relacionales estructurados e InfluxDB para datos de series temporales y métricas.

Conexión a MariaDB

La configuración de MariaDB utiliza un pool de conexiones para optimizar el rendimiento y la gestión de recursos. El pool se crea con el método `createPool` del driver `mariadb`, permitiendo la reutilización de conexiones y evitando el overhead de establecer nuevas conexiones para cada operación.

La configuración requiere varias variables de entorno esenciales: `MYSQL_HOST` para especificar la dirección del servidor de base de datos, `MYSQL_USER` para el usuario de conexión, y `MYSQL_ROOT_PASSWORD` para la autenticación segura. La base de datos objetivo se define como 'WebAPP-DB', que contendrá las tablas y datos estructurados de la aplicación.

Un aspecto importante es el `connectionLimit` establecido en 5, que define el número máximo de conexiones concurrentes en el pool. Esto ayuda a prevenir la sobrecarga del servidor de base de datos mientras mantiene un rendimiento adecuado para la aplicación.

El pool de conexiones maneja automáticamente la creación, reutilización y liberación de conexiones. Cuando la aplicación necesita interactuar con la base de datos, solicita una conexión del pool, la utiliza para ejecutar consultas, y luego la libera para que esté disponible para otras operaciones.

Conexión a InfluxDB

Para InfluxDB, se utiliza el cliente oficial de Node.js, configurado para conectarse a un servidor InfluxDB especificado mediante variables de entorno. La configuración incluye `INFLUXDB_HOST` para la dirección del servidor (con valor por defecto 'localhost'), `INFLUXDB_PORT` para el puerto (por defecto 8086), y `INFLUXDB_DB` para especificar la base de datos de series temporales.

La autenticación se maneja mediante `INFLUXDB_USER` e `INFLUXDB_USER_PASSWORD`, asegurando que solo usuarios autorizados puedan acceder a los datos. El protocolo de comunicación se establece como 'http', aunque en entornos de producción podría configurarse como 'https' para mayor seguridad.

InfluxDB está especialmente optimizado para manejar datos de series temporales, lo que lo hace ideal para almacenar métricas de sensores, datos de monitoreo en tiempo real, y cualquier información que tenga una fuerte componente temporal. Su arquitectura permite escrituras rápidas y consultas eficientes sobre grandes volúmenes de datos temporales.

Casos de Uso y Justificación de la Arquitectura Multi-DB

La decisión de utilizar múltiples bases de datos responde al principio de usar la herramienta adecuada para cada tarea. MariaDB se emplea para datos relacionales estructurados como información de usuarios, perfiles, configuraciones del sistema, y metadatos. Esto garantiza consistencia ACID, integridad referencial, y la capacidad de realizar consultas complejas con joins.

Por otro lado, InfluxDB se especializa en datos de series temporales, como lecturas de sensores de temperatura, métricas de rendimiento del sistema, y logs timestamped. Su optimización para este tipo de datos permite un almacenamiento más eficiente y consultas más rápidas para análisis temporales.

Manejo de Errores y Consideraciones de Robustez

Ambas conexiones implementan manejo de errores para garantizar la estabilidad de la aplicación. Para MariaDB, el pool de conexiones incluye mecanismos para manejar conexiones caídas y reconexiones automáticas. Es importante siempre liberar las conexiones después de su uso para evitar fugas de recursos.

En el caso de InfluxDB, el cliente incluye capacidades de reintento y manejo de errores de red. Para operaciones críticas, se recomienda implementar lógica de reintento y fallback apropiada.

Variables de Entorno Requeridas

El módulo depende de varias variables de entorno para su configuración. Para MariaDB: `MYSQL_HOST`, `MYSQL_USER`, y `MYSQL_ROOT_PASSWORD`. Para InfluxDB: `INFLUXDB_HOST`, `INFLUXDB_PORT`, `INFLUXDB_DB`, `INFLUXDB_USER`, e `INFLUXDB_USER_PASSWORD`. Todas estas variables deben estar definidas en el archivo `.env` del proyecto para que las conexiones funcionen correctamente.

Escalabilidad y Mantenimiento

La arquitectura está diseñada para ser escalable. El pool de conexiones de MariaDB puede ajustarse incrementando el `connectionLimit` según las necesidades de carga. InfluxDB puede escalar horizontalmente para manejar grandes volúmenes de datos de series temporales.

Para mantenimiento, se recomienda monitorear el uso de conexiones en el pool de MariaDB y implementar políticas de retención adecuadas en InfluxDB para gestionar el ciclo de vida de los datos temporales.

Expansión Futura

El módulo está estructurado para facilitar la adición de nuevas conexiones a bases de datos. La arquitectura modular permite incorporar fácilmente sistemas como Redis para caching, MongoDB para documentos, o cualquier otra base de datos que pueda ser necesaria en el futuro.

Esta aproximación multi-base de datos proporciona la flexibilidad necesaria para evolucionar la aplicación mientras mantiene un alto rendimiento y utiliza cada sistema de base de datos para lo que mejor hace.