

Database Schema - Resource Monitoring and Management System

Overview

Our project database has been designed using normalization principles to ensure data integrity and reduce redundancy. It consists of six related tables that work together to provide comprehensive system monitoring and management capabilities.

Tables

1. users

Purpose: Manages administrator accounts and serves as the root entity for authentication.

Attributes: - `id` (PK) - Integer, primary key, auto-incremented - `email` - String, unique, indexed, not null - `hashed_password` - String, not null (securely hashed using bcrypt) - `is_active` - Boolean, default: True - `created_at` - DateTime with timezone, server default: current timestamp

2. systems

Purpose: A central registry storing detailed information about each monitored system/device, preventing data duplication. This table represents the physical or virtual machines being monitored.

Attributes: - `id` (PK) - Integer, primary key, auto-incremented - `hostname` - String, unique, indexed, not null - `ip_address` - String, nullable - `mac_address` - String, nullable - `os_info` - String, nullable (operating system name and version) - `os_build` - String, nullable -

windows_edition - String, nullable - **user_label** - String, nullable (custom label for identification)
 - **agent_version** - String, nullable - **cpu_name** - String, nullable - **cpu_cores** - Integer, nullable -
cpu_threads - Integer, nullable - **architecture** - String, nullable (x64, x86, etc.) -
total_memory_gb - Float, nullable - **total_disk_gb** - Float, nullable - **disk_model** - String,
 nullable - **gpu_name** - String, nullable - **manufacturer** - String, nullable (system manufacturer) -
model - String, nullable (system model) - **serial_number** - String, nullable - **bios_version** -
 String, nullable - **username** - String, nullable (logged-in user) - **domain** - String, nullable -
timezone - String, nullable - **network_adapter** - String, nullable - **battery_percent** - Float,
 nullable - **is_plugged_in** - Boolean, nullable - **python_version** - String, nullable - **is_active** -
 Boolean, default: True - **last_seen** - DateTime with timezone, nullable - **drivers** - JSON, nullable
 (list of installed drivers) - **created_at** - DateTime with timezone, server default: current timestamp

Relationships: - One-to-many with **alerts**, **metrics**, and **tickets** (cascade delete)

3. alerts

Purpose: Stores system-generated alerts based on threshold violations or critical events. Each alert is linked to a specific system and tracks its resolution status.

Attributes: - **id** (PK) - Integer, primary key, auto-incremented - **system_id** (FK) - Integer, foreign key to **systems.id** - **alert_type** - String, not null (e.g., "CPU", "MEMORY", "DISK") - **severity** - String, not null (e.g., "CRITICAL", "WARNING", "INFO") - **message** - String, not null (alert description) - **is_resolved** - Boolean, default: False - **created_at** - DateTime with timezone, server default: current timestamp

Relationships: - Many-to-one with **systems**

4. metrics

Purpose: The central time-series data store. Each row represents a snapshot of system performance metrics at a specific timestamp, enabling trend analysis and historical tracking.

Attributes: - **id** (PK) - Integer, primary key, auto-incremented - **system_id** (FK) - Integer, foreign key to **systems.id** - **timestamp** - DateTime with timezone, indexed, server default: current timestamp - **cpu_usage** - Float (percentage) - **memory_percent** - Float (percentage) -

`memory_used` - BigInteger (bytes) - `disk_free` - BigInteger (bytes) - `disk_usage` - Float (percentage) - `disk_read_bytes` - BigInteger, default: 0 (cumulative) - `disk_write_bytes` - BigInteger, default: 0 (cumulative) - `network_sent` - BigInteger (bytes) - `network_recv` - Float (bytes) - `process_count` - Integer - `boot_time` - String, nullable - `uptime_seconds` - Integer, nullable - `uptime_human` - String, nullable (human-readable format) - `top_processes` - JSON, nullable (list of resource-intensive processes)

Relationships: - Many-to-one with `systems`

5. tickets

Purpose: Manages support tickets or issues reported for specific systems. Tracks ticket lifecycle from creation to resolution.

Attributes: - `id` (PK) - Integer, primary key, auto-incremented - `system_id` (FK) - Integer, foreign key to `systems.id` - `message` - String, not null (ticket description) - `status` - String, default: "OPEN" (values: "OPEN", "RESOLVED") - `resolved_at` - DateTime with timezone, nullable - `created_at` - DateTime with timezone, server default: current timestamp

Relationships: - Many-to-one with `systems`

6. alert_settings

Purpose: Stores customizable threshold configurations for alert generation. Allows per-system or global threshold settings for proactive monitoring.

Attributes: - `id` (PK) - Integer, primary key, auto-incremented - `system_id` (FK) - Integer, foreign key to `systems.id`, nullable (null = global settings) - `cpu_threshold` - Float, default: 90.0 (percentage) - `memory_threshold` - Float, default: 90.0 (percentage) - `disk_threshold` - Float, default: 90.0 (percentage)

Relationships: - Many-to-one with `systems` (optional)

Database Relationships Summary

```
users (1)
└─ Authentication only (no direct FK relationships)

systems (1)
├─ alerts (Many)
├─ metrics (Many)
├─ tickets (Many)
└─ alert_settings (Many)
```

Key Design Features

1. **Normalization:** Data is properly normalized to reduce redundancy and maintain integrity.
2. **Cascade Deletion:** When a system is deleted, all related alerts, metrics, and tickets are automatically removed.
3. **Indexing:** Strategic indexes on frequently queried fields (`email` , `hostname` , `timestamp`) for optimal performance.
4. **Timezone Awareness:** All timestamps use timezone-aware DateTime fields for accurate global monitoring.
5. **Flexible Storage:** JSON columns for complex nested data (drivers, top_processes).
6. **Soft Deletes:** `is_active` flags allow deactivation without data loss.
7. **Audit Trail:** `created_at` timestamps on all tables for historical tracking.