
Software Design Specifications

for

Bandwidth Monitoring System (Version 1.0)

Prepared by:

V. Nithin Reddy

Document Information

Title: Bandwidth Monitoring System	
Project Manager: Panvi Tej	Document Version No: 1
Prepared By: V. Nithin Reddy	Document Version Date: 08 – 04 -2025
	Preparation Date: 08 – 04 -2025

Version History

Ver. No.	Ver. Date	Revised By	Description	Filename
1.0	01-04-2025	Kushwanth	Initial Draft Created	BMS_DesignDoc_v1.0.pdf
1.1	05-04-2025	Panvi Tej	Added Use Case	BMS_DesignDoc_v1.1.pdf
1.2	09-04-2025	Nithin	Final Version with diagrams	BMS_DesignDoc_v1.2.pdf

Table of Contents

1 INTRODUCTION	4
1.1 PURPOSE	4
1.2 SCOPE	4
1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS	4
1.4 REFERENCES	4
2 USE CASE VIEW	4
2.1 USE CASE	4
3 DESIGN OVERVIEW	4
3.1 DESIGN GOALS AND CONSTRAINTS	5
3.2 DESIGN ASSUMPTIONS	5
3.3 SIGNIFICANT DESIGN PACKAGES	5
3.4 DEPENDENT EXTERNAL INTERFACES	5
3.5 IMPLEMENTED APPLICATION EXTERNAL INTERFACES	5
4 LOGICAL VIEW	5
4.1 DESIGN MODEL	6
4.2 USE CASE REALIZATION	6
5 DATA VIEW	6
5.1 DOMAIN MODEL	6
5.2 DATA MODEL (PERSISTENT DATA VIEW).....	6
5.2.1 Data Dictionary.....	6
6 EXCEPTION HANDLING	6
7 CONFIGURABLE PARAMETERS	6
8 QUALITY OF SERVICE	7
8.1 AVAILABILITY.....	7
8.2 SECURITY AND AUTHORIZATION	7
8.3 LOAD AND PERFORMANCE IMPLICATIONS	7
8.4 MONITORING AND CONTROL	7

1 Introduction

This document outlines the software design specifications for the Bandwidth Monitoring System. It serves as a blueprint for developers, testers, and stakeholders to understand the system's architecture, components, and interactions.

1.1 Purpose

The Bandwidth Monitoring System is designed to track and analyze network bandwidth usage in real-time, providing alerts and reports for administrators. This document covers the system's design, including modules, interfaces, and data models.

1.2 Scope

[A brief description of what the **Software Design Specifications** applies to; what is affected or influenced by this document.]

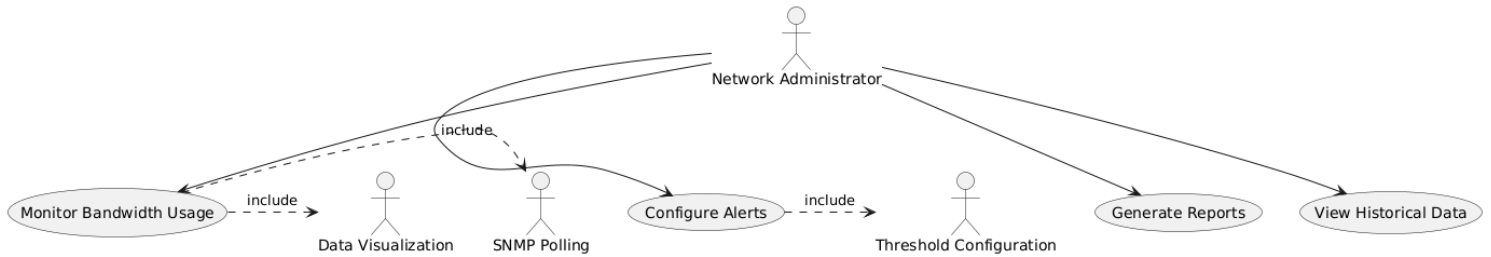
1.3 Definitions, Acronyms, and Abbreviations

- **BMS:** Bandwidth Monitoring System
- **API:** Application Programming Interface
- **SNMP:** Simple Network Management Protocol
- **UI:** User Interface
- **Mbps:** Megabits per second
- **LAN:** Local Area Network.

1.4 References

- SNMP Protocol Specifications
- IEEE SRS Template
- Flask documentation (<https://flask.palletsprojects.com/>)
- SQLite official docs (<https://sqlite.org/docs.html>)
- psutil Python Library

2 Use Case View



2.1 Use Case

UC1: Monitor Bandwidth Usage

- Trigger: System startup
- Flow: Reads current network I/O stats, stores in DB every X seconds

UC2: Set Thresholds

- Trigger: Admin sets limit via dashboard
- Flow: Threshold is stored; alerts triggered when usage exceeds it

UC3: View Reports

- Trigger: User logs in to dashboard
- Flow: Queries usage data from DB and displays interactive graphs

3 Design Overview

3.1 Design Goals and Constraints

- **Goals:** Real-time monitoring, scalability, and minimal performance impact.
- **Constraints:** Must support SNMP v2/v3 and integrate with existing network infrastructure.
- Real-time monitoring at intervals ≤ 5 seconds
- Lightweight footprint for background daemon
- Web dashboard accessible over LAN

3.2 Design Assumptions

- System has Python 3 and Flask installed
- Admin has access to configure IP restrictions
- The system includes both web-based UI and API access for integration
- Network time is synchronized across components for accurate metrics

3.3 Significant Design Packages

- **Core Monitor:** Gathers bandwidth data
- **Data Storage:** Manages data persistence (SQLite)
- **Web Dashboard:** Flask-based UI for statistics
- **Alert System:** Sends notifications if thresholds are breached

3.4 Dependent External Interfaces

External Application Module Using Interface Functionality/Description		
SNMP Service	Data Collection	Fetches bandwidth metrics from devices

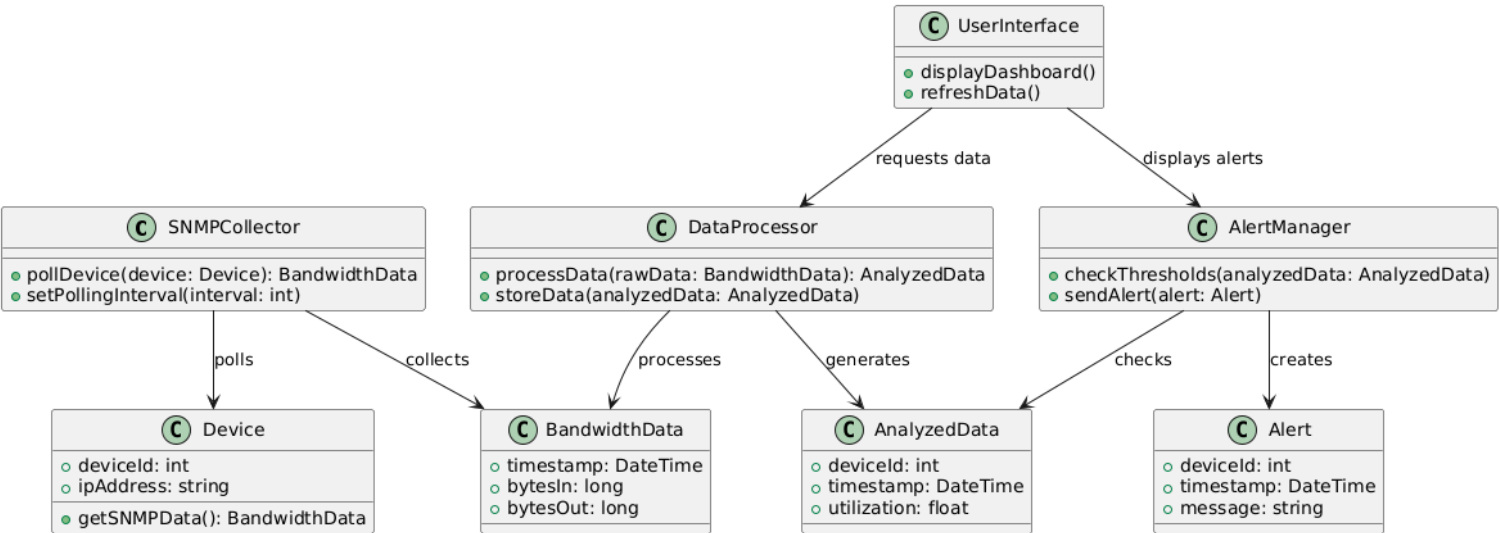
3.5 Implemented Application External Interfaces (and SOA web services)

Interface Name Module Implementing Functionality/Description		
REST API	User Interface	Provides bandwidth data to clients

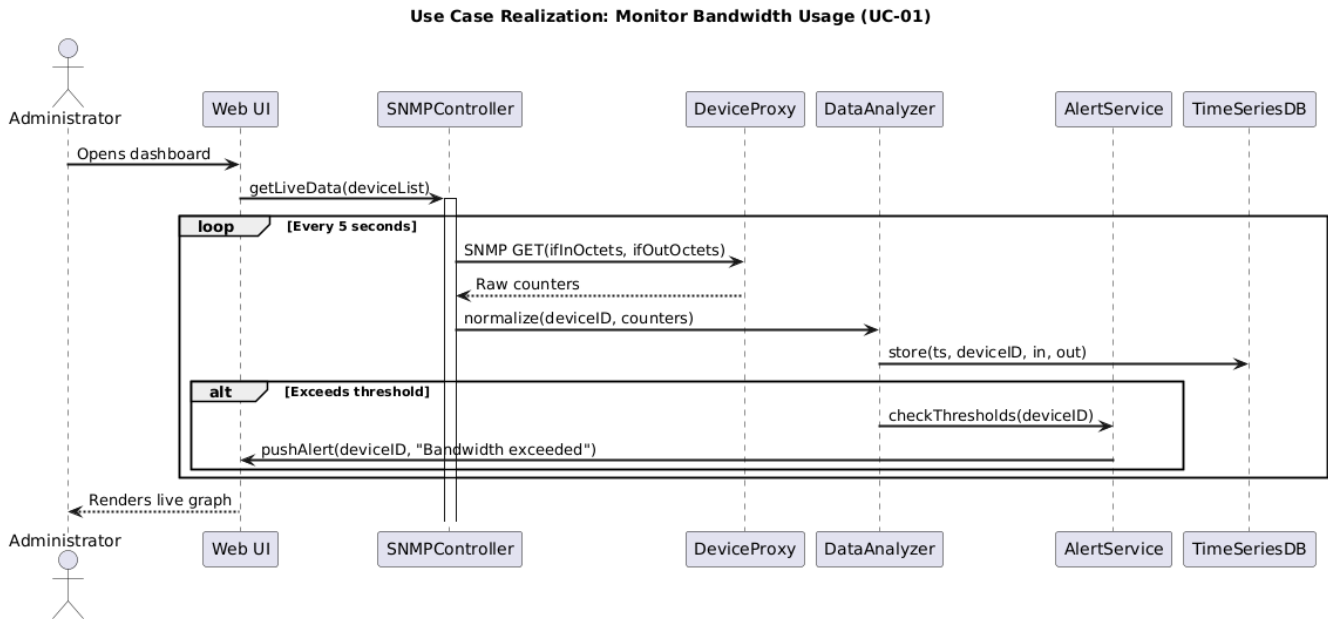
4 Logical View

4.1 Design Model

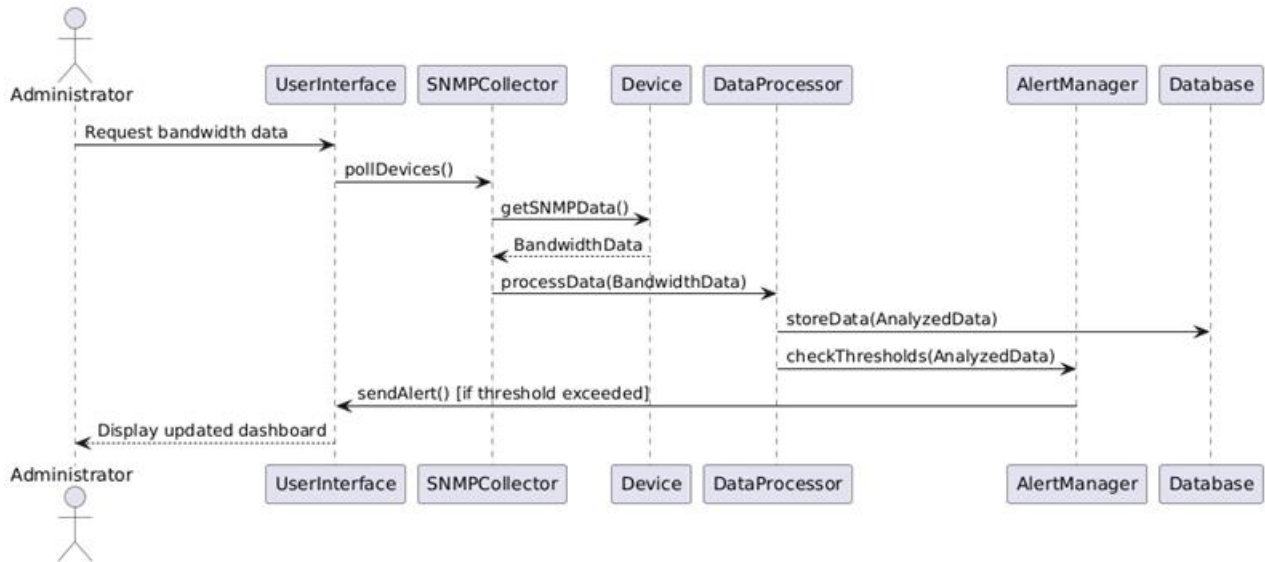
Classes:



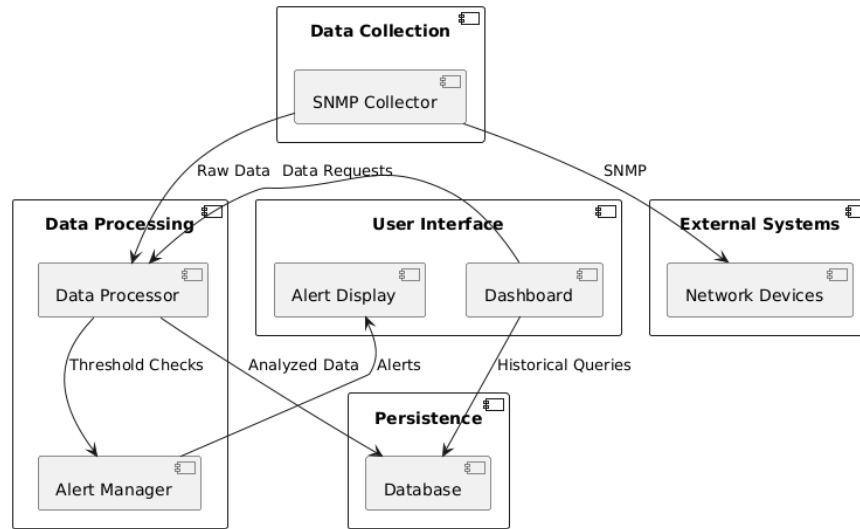
4.2 Use Case Realization



Sequence Diagram



Component Diagram



5 Data View

5.1 Domain Model

Entities:

- Device: Network device being monitored.
- BandwidthMetric: Stores usage data.

5.2 Data Model (persistent data view)

- Devices: Device details.
- Metrics: Timestamped bandwidth data.

5.2.1 Data Dictionary

Field	Type	Description
timestamp	DATETIME	Time of record
sent_MB	REAL	Data sent
recv_MB	REAL	Data received
threshold	REAL	Alert threshold
Device_id	INT	Unique Device identifier

6 Exception Handling

- **Exceptions:**
 - SNMPPollingError: Failed device communication.
 - DataStorageError: Database write failure.

7 Configurable Parameters

Parameter Name	Definition and Usage	Dynamic?
polling_interval	Time between SNMP polls (seconds)	Yes

8 Quality of Service

8.1 Availability

- System uptime target: 99.9%.
- Maintenance windows scheduled during off-peak hours.
- Designed to run as a background daemon with system auto-restart on failure
- SQLite ensures data recovery during outages

8.2 Security and Authorization

- Role-based access control for dashboard and alerts.
- Access limited to local IPs (configured in Flask)

8.3 Load and Performance Implications

- Supports up to 1,000 devices with 5-second polling.

8.4 Monitoring and Control

- Logs and alerts for system health and performance.