International Journal on Science and Technology (IJSAT)

E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

Netflow Analyzer

Katuri Jahnavi Priya[1], Thota Sravanthi[2], P. Poornima[3], K. Shirisha[4], Dr.Rajitha Kotoju[5], R. Mohan Krishna Ayyappa[6]

[1, 2] Student, Department of Computer Science and Engineering, Mahatma Gandhi Institute of Technology, Gandipet, India.

[3, 4, 5, 6] Assistant Professor, Department of Computer Science and Engineering, Mahatma Gandhi Institute of Technology, Gandipet, India.

ABSTRACT

The rapid growth in wireless device usage has created an urgent need for intelligent, secure, and efficient network management systems. This paper introduces NetFlow Analyzer, an advanced Wi-Fi management solution that offers a real-time, end-to-end view of network activity with a focus on bandwidth tracking, indoor location detection, device management, and security enforcement. The system enables per-device bandwidth monitoring, allowing administrators to visualize data usage and detect unusual activity with precision. A standout feature of NetFlow Analyzer is its indoor positioning system, which uses RSSI-based Wi-Fi triangulation combined with SciPy optimization techniques to accurately track the physical location of connected devices within a defined space. Built on a Flask backend with MongoDB and a feature-rich React.js frontend, the platform provides a seamless user experience. Administrators can manage connected devices through an approval-based control system, ensuring only authorized users access the network. To safeguard against misuse, the system triggers automated alerts for data threshold breaches and unauthorized connection attempts. Incorporating Scapy for deep packet inspection and real-time network traffic analysis, NetFlow Analyzer empowers administrators with detailed visibility and control. The solution is ideal for high-traffic environments like campuses, enterprises, and public Wi-Fi zones, delivering an integrated approach to performance monitoring, location awareness, and network security.

Index Terms: Network Monitoring, Device Localization, WiFi Triangulation, Real-Time Data Analysis

1. INTRODUCTION

NetFlow Analyzer is a real-time network monitoring and device tracking system designed to improve WiFi network security, efficiency, and management across various environments like corporate offices, universities, public hotspots, and smart buildings. By leveraging IP/MAC detection, RSSI-based triangulation, and cloud-based data handling, NetFlow Analyzer provides administrators with live insights into connected devices, their approximate indoor locations, and network activity trends. Built using a Flask-based backend and scientific computing libraries like NumPy and SciPy, the system focuses on delivering accurate device localization and maintaining secure network environments. With planned expansion toward bandwidth monitoring and deep packet inspection, NetFlow Analyzer forms the backbone of a responsive, secure, and optimized WiFi ecosystem that minimizes manual monitoring and enhances operational efficiency.

1.1 Problem Definition

Traditional network monitoring tools often focus either on external threats or only on traffic volumes without offering real-time device location tracking within a network. Many systems lack the capability to estimate the physical position of connected devices, making it difficult for administrators to detect unauthorized access, optimize resource allocation, or manage security breaches effectively. Manual tracking of devices based only on IP addresses leaves room for errors and delays. Furthermore, the absence of integrated live monitoring systems with device localization leads to fragmented insights, reduced network performance, and higher security vulnerabilities. There is a critical need for a unified platform that not only tracks devices and their usage but also estimates their positions within a networked space.

1.2 Existing Applications

Existing solutions such as SolarWinds NetFlow Traffic Analyzer, Wireshark, and PRTG Network Monitor offer excellent capabilities for bandwidth usage tracking, packet inspection, or network performance analysis. However, they are often expensive, complex to configure, and primarily focus on external traffic flows rather than internal device localization. Moreover, most traditional tools lack real-time indoor device tracking based on RSSI triangulation, leaving gaps in physical security monitoring. Solutions offering precise device localization typically require specialized hardware installations, making them impractical for small- to medium-sized deployments. The absence of an affordable, lightweight, and real-time device tracking and network monitoring platform highlights a gap that NetFlow Analyzer aims to fill.

1.3 Proposed Application

NetFlow Analyzer offers a unified, real-time solution to monitor connected devices, track their approximate locations indoors using RSSI-based WiFi triangulation, and maintain historical connectivity data for future analysis. Developed using Flask for backend management and SQLAlchemy for database operations, the system utilizes NumPy and SciPy for scientific location calculations. Environment variables are securely handled through python-dotenv. While tools like psutil, netifaces, and Scapy are integrated into the project environment for future enhancements like bandwidth monitoring and packet inspection, the current focus remains on device discovery and location estimation. The modular architecture ensures easy customization and scalability, making it adaptable for campuses, corporate setups, public WiFi networks, and smart infrastructure. NetFlow Analyzer not only improves network visibility and security but also lays the foundation for future developments like AI-driven threat detection and cloud-based network management.

2. LITERATURE SURVEY

“Outdoor-Indoor Bridging for Reference-Point-Free 3D WiFi AP Localization” by Tatsuya Amano, Hirozumi Yamaguchi, and Teruo Higashino. Introduces WiSight, a GPS-anchored 3D WiFi AP localization framework that eliminates the need for indoor reference points [1]. By leveraging GPS-tagged RSS measurements and 3D building geometry, WiSight achieves 59% better accuracy than traditional methods. For our project, this presents a scalable alternative for indoor positioning, aligning with our

RSSI-based triangulation approach and enhancing location accuracy without labor-intensive calibration.

"SWIF: Smart Wi-Fi Integration Framework for Mobility and Resource Optimization-" by S. Manzoor, S. Azam, R. Wójcik, and J. Domżał. **I**ntroduces the Smart Wi-Fi Integration Framework (SWIF), a solution that uses Software-Defined Networking (SDN) for optimizing user association with access points (APs) and dynamic channel allocation [2]. By utilizing metrics like Mean Received Signal Strength Indicator (RSSI), AP workload, and Channel Utilization Level, SWIF offers real-time adjustments for mobility and resource allocation, improving indoor wireless operations. For our project, this highlights the importance of SDN in enhancing Wi-Fi network management and dynamic allocation of resources for improved performance.

"WiLoc: Encoding-based WiFi Indoor Localization" by Z. Huang, M. Valkama, J. Zhang, M. Xu, C. Yin, and M. Guan. Presents WiLoc, a Wi-Fi-based indoor localization method using a Lightweight Siamese Neural Network [3]. The Siamese network learns the similarity between inputs, utilizing a triplet loss function for feature encoding and K-Nearest Neighbors (KNN) for position prediction. For our project, this research demonstrates the potential of encoding-based localization methods for accurate indoor positioning without complex deep learning models.

"Triangulation-Enhanced WiFi-Based Autonomous Localization and Navigation System: A Low-Cost Approach" by S. A. Kharmeh, E. Natsheh, R. Nasrallah, and M. Masri. Investigates Wi-Fi signal strength for low-cost indoor navigation using triangulation techniques [4]. Mathematical models integrate logarithmic and linear regression to calculate distances for robot positioning. The research highlights the feasibility of triangulation for low-cost systems with applications in robotic navigation. For our project, this emphasizes the importance of Wi-Fi signal-based triangulation for indoor localization and navigation in cost-effective systems.

"WiFi-Crowd Spy: A Novel Crowd-Counting System" by A. Collaguazo, R. Estrada, I. Valeriano, and J. Tomala. Proposes a crowd-counting detection system that uses Wi-Fi infrastructure to estimate indoor crowd size based on access point (AP) connectivity [5]. The system sends alerts when the capacity of a building is about to be exceeded. For our project, this illustrates the use of Wi-Fi data for crowd control and monitoring, which can be incorporated into building management systems for better utilization and safety.

"Hybrid LiFi and WiFi Networks: A Survey" by X. Wu, M. D. Soltani, L. Zhou, M. Safari, and H. Haas. This survey comprehensively explores hybrid LiFi-WiFi systems, addressing challenges in resource allocation, mobility, and interference [6]. It presents state-of-the-art solutions and future research directions. The findings are highly relevant to our project, as we explore integration and coexistence strategies in VLC-WiFi heterogeneous networks.

"Real-Time Monitoring of Network Devices: Its Effectiveness in Enhancing Network Security" by W. L. Ogogo. This paper explores the effectiveness of real-time network device monitoring in enhancing network security by providing immediate alerts on network issues [7]. It highlights the advantages of proactive monitoring in detecting and resolving network vulnerabilities. For our project, this emphasizes the need for continuous network monitoring to enhance security in real-time Wi-Fi management systems.

"Attention to Wi-Fi Diversity: Resource Management in WLANs With Heterogeneous APs" by J. Saldana, J. Ruiz-Mas, J. Fernández-Navajas, J. L. Salazar Riaño, J.-P. Javaudin, and J.-M. Bonnamy. Introduces a controller-assisted framework for managing heterogeneous WiFi APs using smart handoffs and load balancing based on AP and device characteristics [8]. It shows minimal delay (10–50 ms) in real-world tests. This supports efficient resource management in indoor networks with mixed AP technologies.

"WiFi Based Indoor Positioning System Using Machine Learning and Multi-Node Triangulation Algorithms" by S. Jahagirdar, A. Ghatak, and A. A. Kumar. Presents an RFID-based security framework for IoT systems, addressing vulnerabilities in RFID communication. It ensures secure data exchange through formal security analysis, focusing on authentication and encryption mechanisms to prevent unauthorized access [9]. For our project, this reinforces the need for secure RFID integration and encryption in wireless networks for user privacy and safety.

"Evaluating Emergency Evacuation Events Using Building WiFi Data" by I. Pasquel Mohottige, H. H. Gharakheili, A. Vishwanath, S. S. Kanhere, and V. Sivaraman. Uses Wi-Fi data to evaluate both planned and unplanned emergency evacuation events, offering insights into evacuation patterns across different building layouts [10]. The study develops methods to automatically identify evacuation events from Wi-Fi data. For our project, this paper underscores the value of Wi-Fi data for real-time monitoring and emergency management in large indoor spaces.

3. METHODOLOGY OF NETFLOW ANALYZER

This section outlines the approach taken to design, develop, and implement the NetFlow Analyzer system, ensuring a robust, scalable, and user-friendly Wi-Fi management solution with bandwidth usage tracking, device management, and security alerts.

3.1 Technologies Used:

- **Programming Language:** Python 3.8 or later P was chosen for its simplicity, extensive library support, and compatibility with network analysis tools, making it suitable for implementing core features like bandwidth tracking, device management, and location tracking.
- **GUI Framework:** React.js was used for building a responsive, interactive frontend that enables real-time updates and visualizations for device management, bandwidth tracking, and security alerts. It ensures smooth interaction for both users and administrators.
- **Backend Framework:** Flask was chosen for its lightweight and flexible architecture, allowing for the seamless integration of various modules such as device management, security alerts, and location tracking through the backend.
- **Database:** MongoDB, a NoSQL database, was used for storing device information, network statistics, and user data. Its flexible schema makes it ideal for handling the varied and real-time nature of data generated by the Wi-Fi network.
- **Network Analysis:** Scapy was used for network packet analysis, helping in the detection of devices and monitoring traffic. It assists in the real-time monitoring of Wi-Fi network usage and ensures accurate device tracking and data analysis.
- **Indoor Location Tracking:** The indoor location tracking is implemented using RSSI-based triangulation. SciPy's optimization tools are used for fine-tuning the location estimates, ensuring accurate device tracking within the indoor network.

- **Alerting System:** Real-time alerts for devices exceeding data usage or attempting to connect are implemented using Flask and WebSocket for instant push notifications to administrators.
- **Other Libraries:**
  - NumPy for data manipulation and calculations.
  - Pandas for handling and analyzing large datasets from the Wi-Fi network.

3.2 Development Process:

The development of the NetFlow Analyzer followed an iterative and modular approach to ensure continuous feedback and refinement throughout the lifecycle of the project:

- **Requirement Analysis**: Stakeholder needs (network administrators, users, and security teams) were gathered and analyzed to define the system's core modules: bandwidth usage tracking, device management, location tracking, and security alerts.
- **System Design:** The architecture was designed with a focus on modularity, enabling independent functioning of key services (bandwidth monitoring, device management, location tracking, security alerts) while ensuring seamless integration through shared database access and API communication.
- **Module Implementation:**
  - **Bandwidth Usage Tracking:** Developed first to track and log real-time bandwidth usage per device.
  - **Device Management:** Implemented to register, track, and manage devices that connect to the Wi-Fi network.
  - **Indoor Location Tracking:** Integrated Wi-Fi triangulation-based indoor location tracking using RSSI data.
  - **Security Alerts:** Developed real-time alerts for abnormal activities such as new device connections or threshold exceedances in data usage.
- **Database Integration:** MongoDB collections were designed to store information about devices, network usage data, location coordinates, and alerts. This ensured efficient and secure data management and retrieval.
- **Testing:** Each module underwent unit testing and integration testing. Real-world scenarios were simulated to ensure that bandwidth usage, location tracking, and device management worked seamlessly. User acceptance testing was conducted with network administrators and test users to validate the system’s functionality and usability.
- **Deployment & Feedback:** The system was deployed in a controlled environment, and feedback was collected from users (students, mentors and project guides) for further refinement.

4. DESIGN OF NETFLOW ANALYZER

The NetFlow Analyzer is a comprehensive Wi-Fi management solution that provides bandwidth usage tracking, device management, indoor location tracking using Wi-Fi triangulation, and network security alerts. The system is designed with a modular, scalable architecture to ensure easy deployment and future expansion. It integrates advanced optimization techniques and packet analysis for real-time monitoring.

4.1. System Architecture

The system comprises multiple interconnected components responsible for capturing, analyzing, and managing network activities. The overall architecture is depicted in Figure 1

Key Components:

- **Wi-Fi Triangulation Engine:** Calculates device locations using RSSI-based optimization.
- **Bandwidth Tracker:** Monitors real-time bandwidth usage per device.
- **Device Manager:** Handles admin approvals for new device connections.
- **Security Alert System:** Triggers alerts for threshold breaches and unauthorized devices.
- **Flask Backend Server:** Handles APIs, database operations, and data processing.
- **React.js Frontend:** Provides a dynamic, real-time dashboard for admins.
- **MongoDB Database:** Stores user, device, bandwidth, and alert logs.

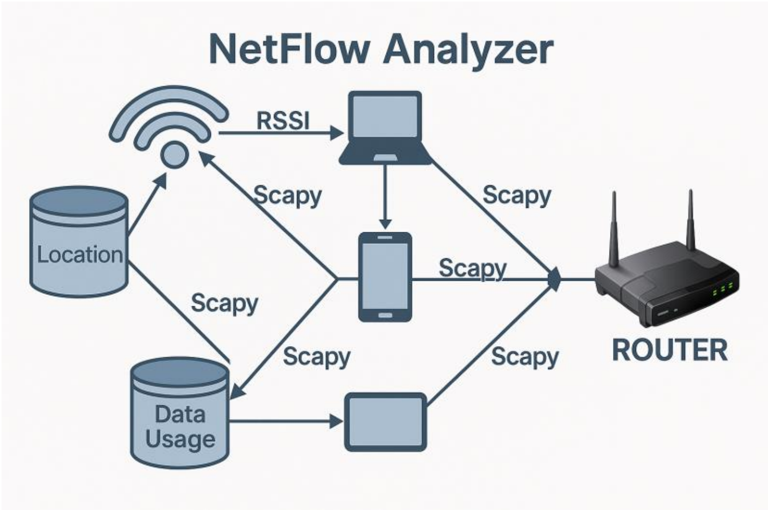

Figure 1: System architecture diagram

4.2. Database Design

The system uses a **MongoDB** database (netflow_analyzer) with the following key collections, as shown in Figure 2:

Collections:

- **devices** → Every device connected to the network is tracked here. It includes MAC and IP addresses, and a status (like active or blocked). Each device is linked to a user. Devices generate bandwidth, location, and alert data.
- **bandwidth_logs** → This records how much internet data each device consumes. It logs the data usage in MB and a timestamp. Admins have full visibility.
- **location_logs** → Devices are tracked indoors using coordinates (X, Y), calculated from RSSI signals. The timestamp tells when the location was recorded. Again, users can only see their own devices' location history.
- **security_alerts** → This captures events like suspicious data usage or unauthorized device connection attempts. Each alert includes a type, description, and timestamp. Admins see all alerts, while

users see only those for their devices.

- **users** → This entity stores all users registered in the system. Each user has a unique ID, username, password, and role (either admin or user). Admins can manage all devices and data, while regular users can only access their own devices and related data.
- **access_requests** → Logs pending connection requests for approval.

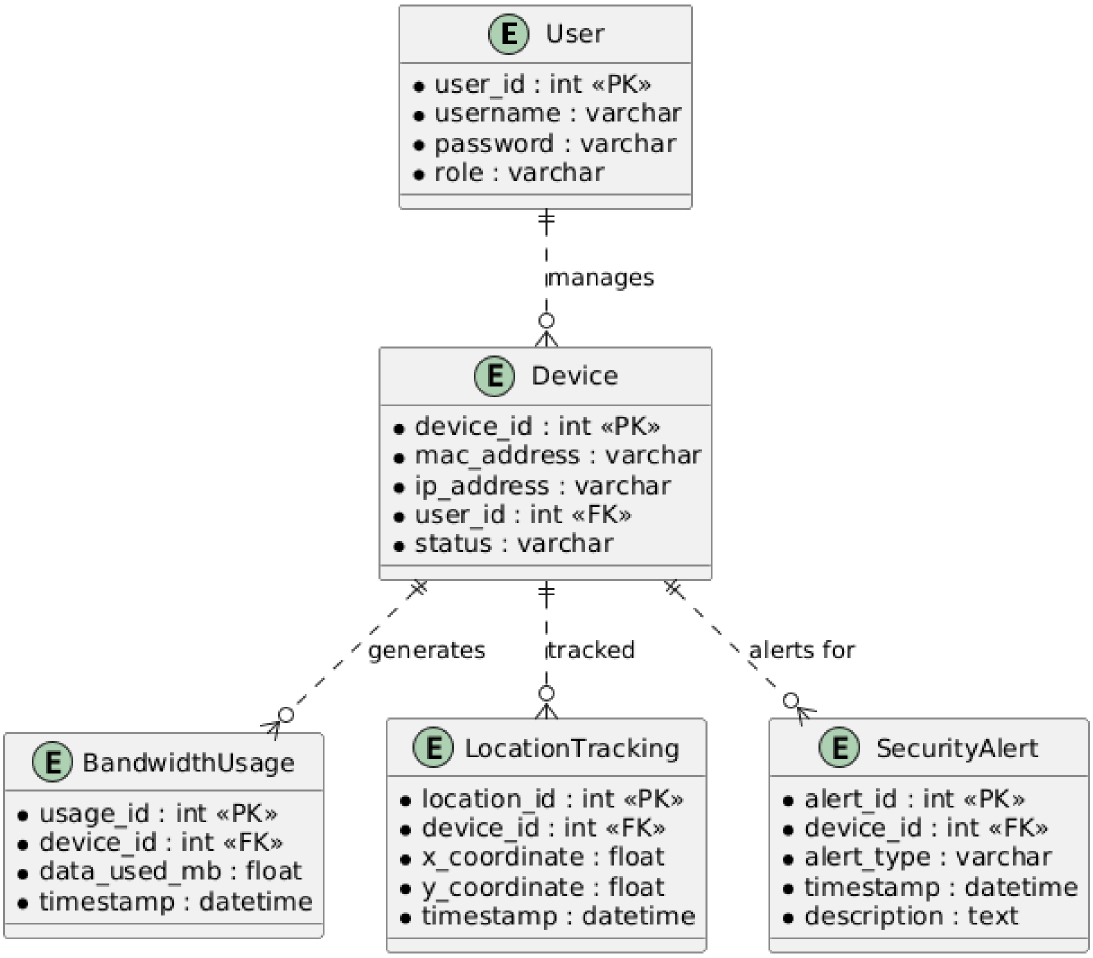

Figure 2: ER diagram

4.3. Control Flow showing User Activity

The Control Flow Diagram in Figure 3 shows the sequential process execution, including device connection management, bandwidth tracking, triangulation updates, and alert generation.

Key Process Flows:

- New device connection → Access request → Admin approval → Device entry creation.
- Active device monitoring → Bandwidth usage update → Threshold check → Alert trigger.
- Indoor location triangulation → Real-time map update.
- Admin dashboard → View bandwidth, devices, alerts, and location tracking.

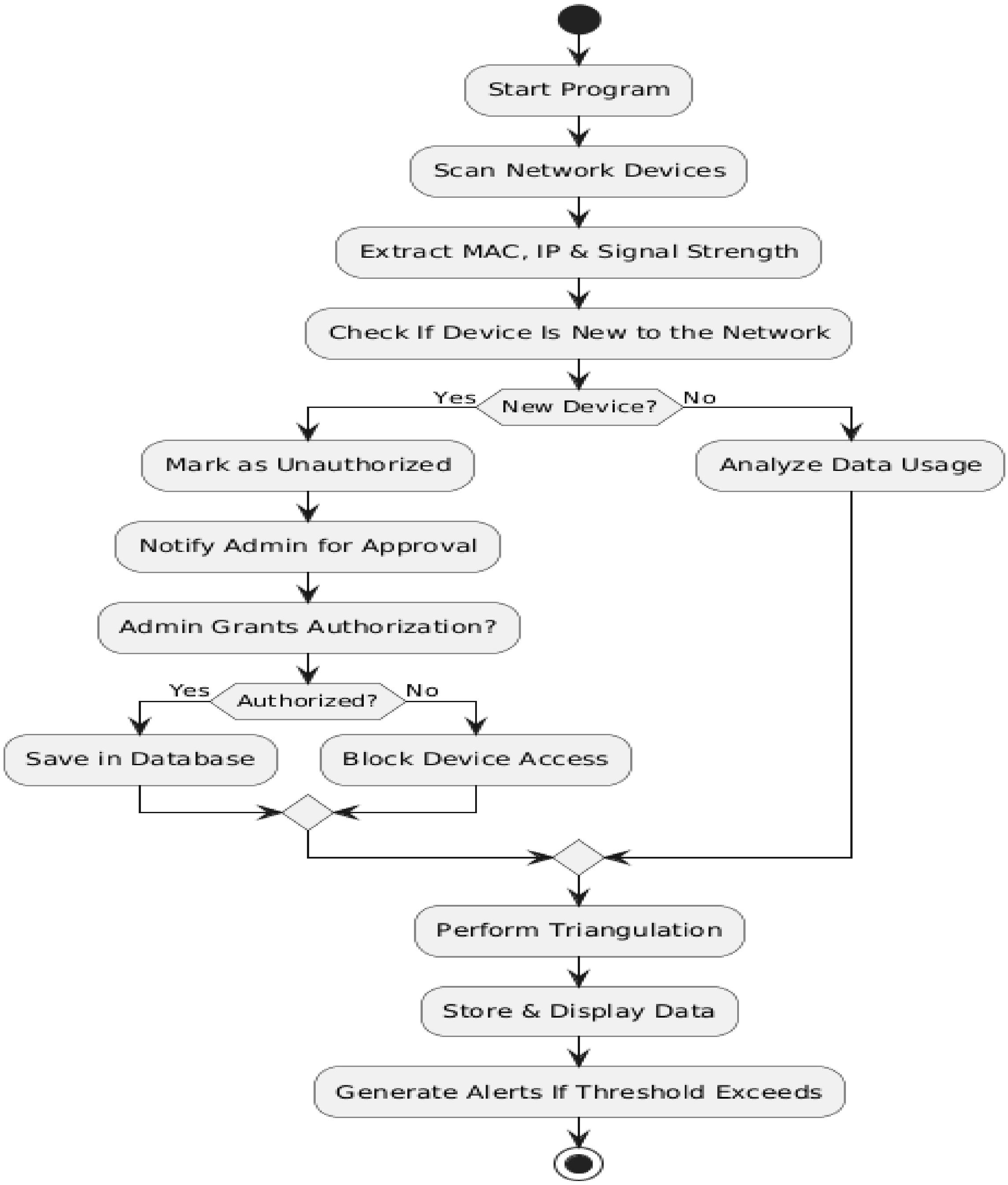

Figure 3: Control Flow diagram

4.4. Modular Architecture & Object-Oriented Structure

The NetFlow Analyzer follows a modular, object-oriented structure to maintain a clean separation between concerns, enhancing scalability and maintainability. Each module is mapped to specific functionality.

International Journal on Science and Technology (IJSAT)

E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

Key Modules & Their Corresponding Classes

Module	Class/Script	Responsibility
Device Manager	device.py	Manage device approvals, status, and details.
Bandwidth Tracker	bandwidth.py	Log and update per-device bandwidth consumption.
Location Tracker	location.py	Triangulate and update device locations.
Alert Manager	alert.py	Generate and store security alerts.
User Manager	user.py	Handle admin authentication and profile management.
API Server (Backend)	app.py (Flask)	API endpoints for devices, bandwidth, locations, and alerts.
Frontend (React)	dashboard.jsx	Admin dashboard for monitoring and management.

Class Diagram Overview

The Class Diagram, shown in Figure 4, illustrates the relationships:

- Device class links to BandwidthTracker, LocationTracker, and AlertManager.
- Admin users interact via APIs exposed by the Flask server.
- The React dashboard consumes API data for display and admin controls.

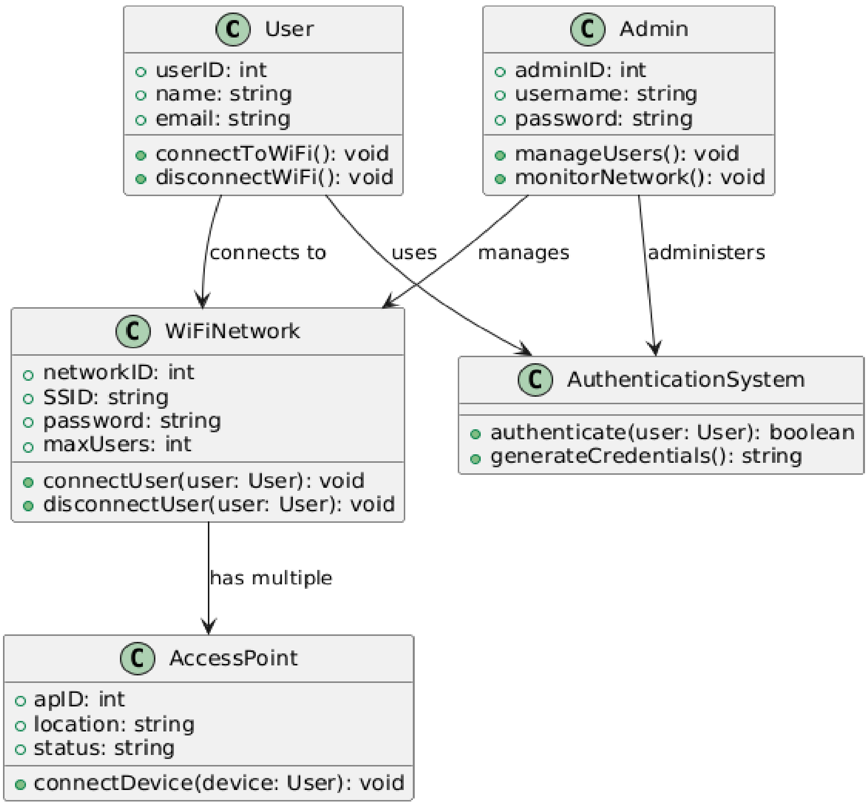

Figure 4: Class diagram

4.5. Role-Based Authentication & Access Control

The NetFlow Analyzer currently focuses on admin role management for controlling system functionalities as shown in Figure 5. Each admin accesses the system through secure login credentials.

Role	Permissions
Admin	View dashboard, approve/reject devices, monitor bandwidth, view locations, receive alerts, reset system data.

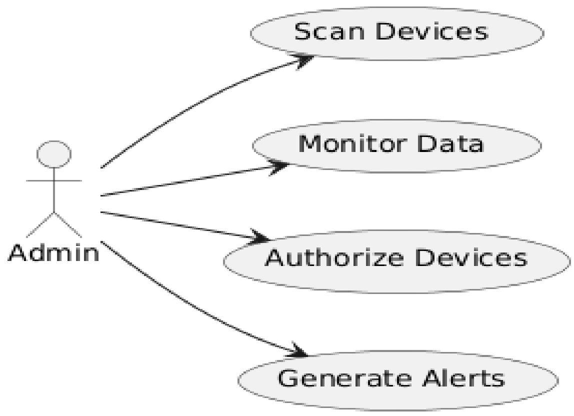

Figure 5: Use case diagram

4.6. Admin-Based Process Workflow

The following sequence diagram (Figure 6) shows the interaction flow for a typical **admin user** while managing a new device request:

Example - Device Approval Workflow:

1. New device attempts connection (detected by packet analysis).
2. Access request is created and sent to admin dashboard.
3. Admin reviews the request and approves/rejects it.
4. If approved, device is added to authorized devices list; else rejected.
5. Security logs are updated accordingly.

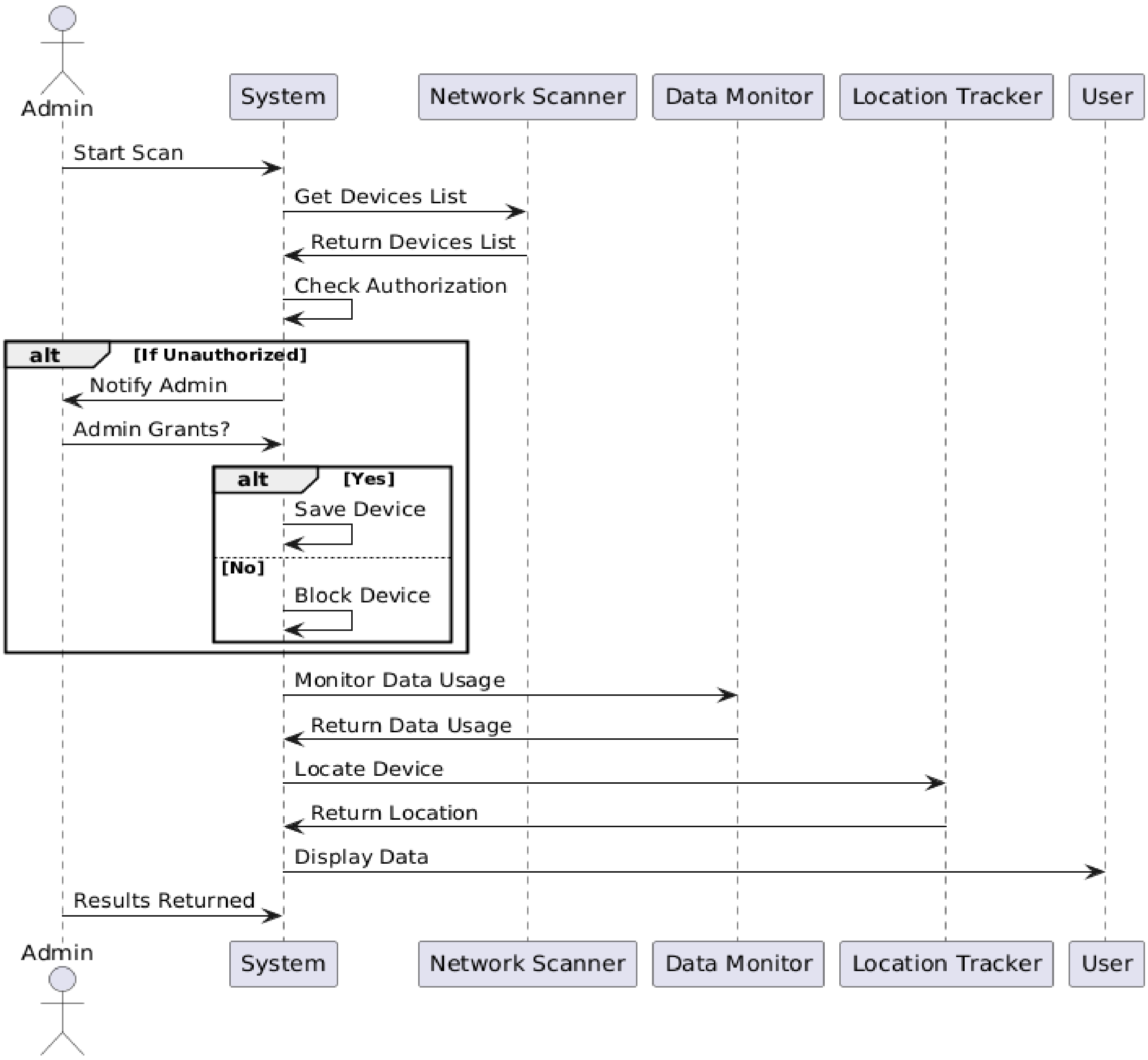

Figure 6: Sequence diagram for Admin

5. IMPLEMENTATION

NetFlow Analyzer is architected as a modular Flask-based web application for real-time Wi-Fi monitoring, location tracking, and bandwidth management. The application follows a RESTful API architecture and uses a SQLite database via SQLAlchemy for persistence. The codebase is structured into distinct modules for device detection, bandwidth tracking, location triangulation, and 3D visualization. Concurrent data collection and monitoring tasks run on daemon threads to ensure responsiveness and real-time updates. Flask serves as the backend API layer, while Matplotlib and JavaScript-based frontends handle interactive visualizations and user interaction. The system is designed to scale, with support for future database and visualization upgrades.

5.2 Database Integration

SQLite is used with SQLAlchemy ORM to manage and persist data related to:

- Connected devices (MAC, IP, hostname, signal strength)
- Bandwidth usage history
- Location coordinates
- Router metadata and signal zones
- Alert logs

CRUD operations are abstracted into utility functions in utils.py , ensuring consistent and secure data access. For example, when a student makes a wallet transaction, the transaction is recorded in the transactions collection, and the student's wallet balance is updated atomically. Real-time updates are leveraged for logs and notifications, and Firestore security rules restrict access based on user roles.

5.3 Device Detection and Real-time Monitoring

Devices are detected using a combination of:

- **Netsh**: For scanning available Wi-Fi hotspots
- **ARP Table Parsing**: To identify already connected devices
- **Ping Sweeps**: To discover devices within the IP range

Once detected, each device's signal strength, MAC/IP address, and hostname are recorded. Real-time monitoring is performed via background threads that:

- Continuously update device status and connection data
- Log bandwidth consumption
- Trigger alerts when usage crosses predefined thresholds

5.4 Indoor Location Tracking

Location tracking is based on **RSSI signal triangulation** and a path loss model. It includes:

- Signal strength data from multiple access points
- Distance estimation from routers using path loss equations
- **Kalman filtering** for reducing environmental noise
- Multi-floor consideration in building geometry

Device locations are updated in real time and mapped onto a 3D model of the building for visual interpretation.

5.5 3D Visualization

The 3D building visualization is implemented using **Matplotlib's 3D plotting** features. Key features include:

- A five-floor virtual building with configurable dimensions (100m × 60m, 3m height/floor)
- Visualization of router positions and signal zones (Excellent, Good, Fair, Poor)
- Real-time device markers showing movement across the network
- Admin interface to view router info and signal health

Visualization data is cached and updated on-demand or on periodic triggers from monitoring threads.

5.6 API Architecture

The application provides RESTful API endpoints for interaction:

- /api/devices: List all connected devices
- /api/devices/<device_id>/approve: Approve a new device
- /api/devices/<device_id>/location: Retrieve the real-time location of a device
- /api/devices/remove: Remove or block a device
- /building-visualization: Serve the frontend 3D layout
- /api/visualization/update: Fetch updated visualization data

These endpoints allow for smooth integration between frontend visual components and backend analytics.

5.7 Security

Security features are enforced through:

- MAC address-based device identification
- Device approval workflow for admin review
- Threshold-based bandwidth usage alerts
- IP-based logging and access control on backend routes

Unauthorized devices are denied access until manually approved. Alerts are generated when suspicious traffic or new device entries are detected.

5.8 User Interface

The frontend interface includes:

- **3D Visualization Page**: Shows building structure, routers, signal zones, and device positions
- **Device Approval Panel**: Interface for admins to manage newly detected devices
- **Alert Dashboard**: Displays security events and bandwidth violations
- **Router Info Display**: Shows network parameters like SSID, channel, speed, and protocol

The UI is built for usability and real-time interaction, using color-coded indicators and popup alerts for key events.

5.9 Error Handling and Reliability

Robust error handling ensures:

- Graceful fallback if network commands fail (e.g., netsh or ping)
- Logging for unauthorized access or hardware disconnections
- Thread health checks and fallback timers for thread recovery
- Resilient visualization updates even under high network churn

Background tasks are wrapped with exception handling blocks and logs are maintained for system audits and troubleshooting.

6. TESTING AND RESULTS

As shown in Figure 7, The Network Monitoring Dashboard displays a 3D layout of the building, where each device is marked with a colored dot indicating its approval status—green for approved and yellow for unapproved. The router is shown at the center in red colour with its network details, and each connected device is labeled with its hostname, IP, and MAC address. This visualization helps admins easily identify authorized and unauthorized devices, monitor their locations, and assess network coverage in real time.

International Journal on Science and Technology (IJSAT)

E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

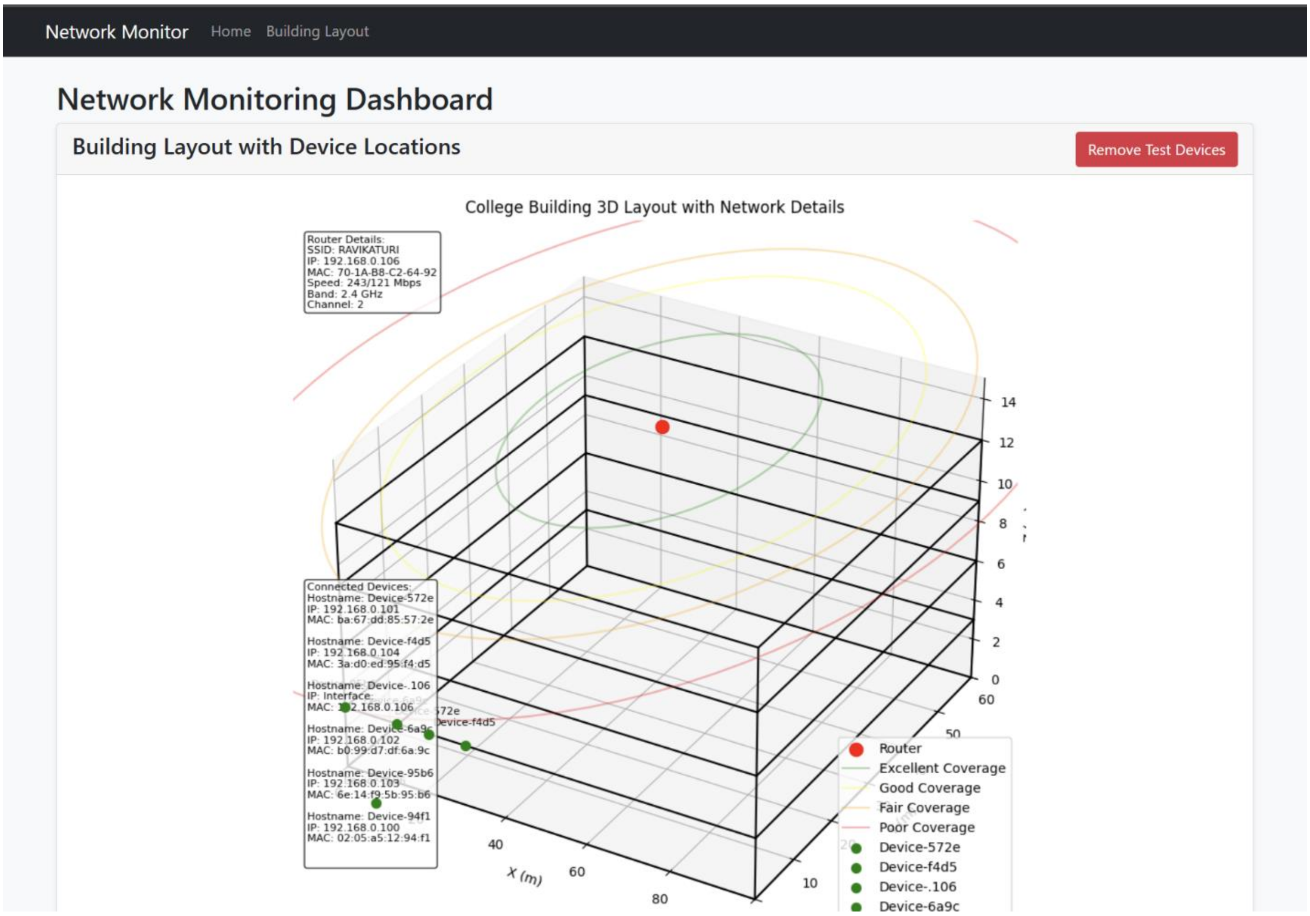

Figure 7: The Home Page

Figure 8 shows the list of connected devices to the Wi-fi network along with each device's MAC Address, IP Address, Location, Last Seen and bandwidth usage and approval status.

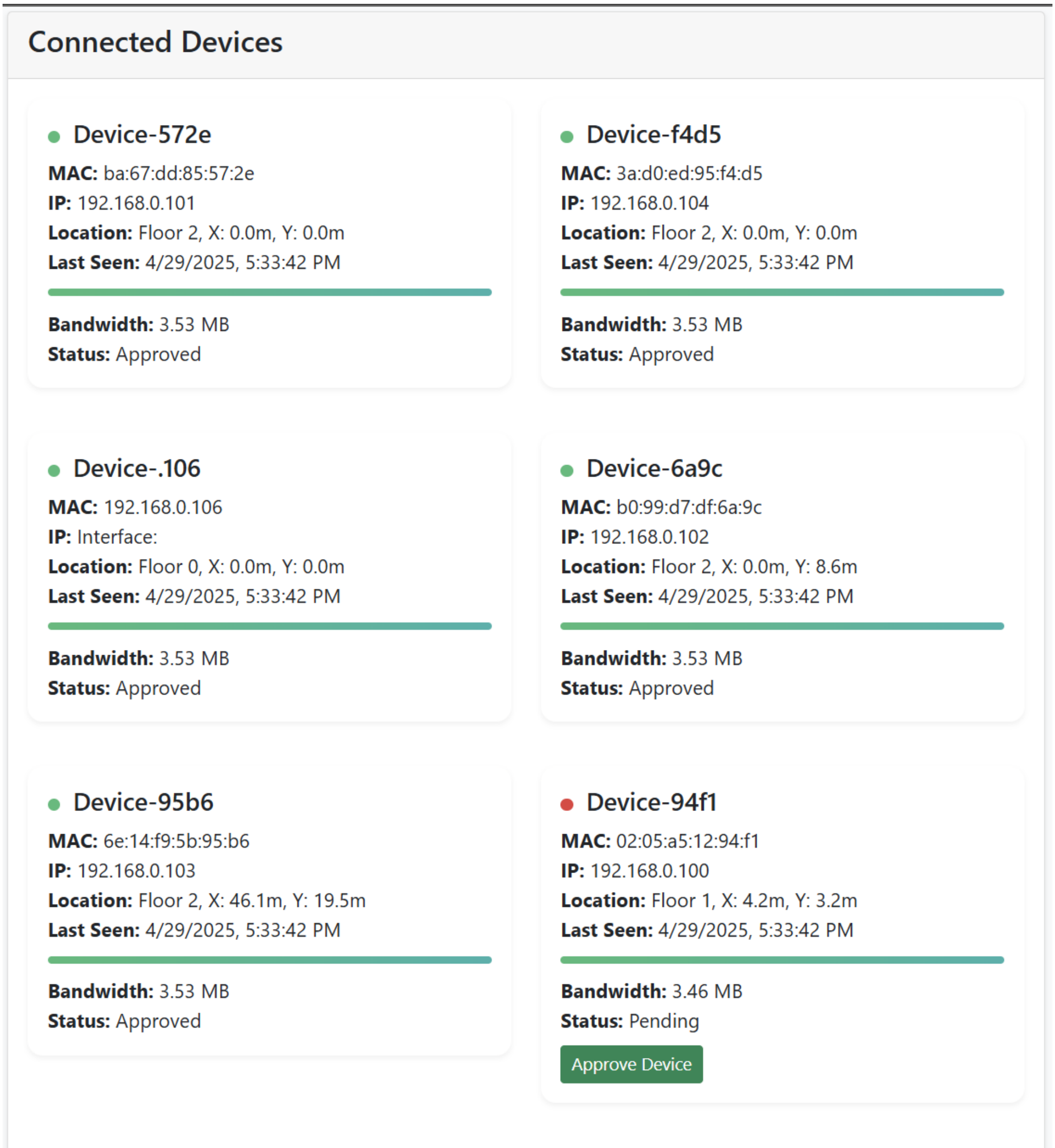

Figure 8: Connected Devices

As shown in Figure 9, recent alerts are shown, where whenever new devices are trying to connect, immediately a new alert is sent.

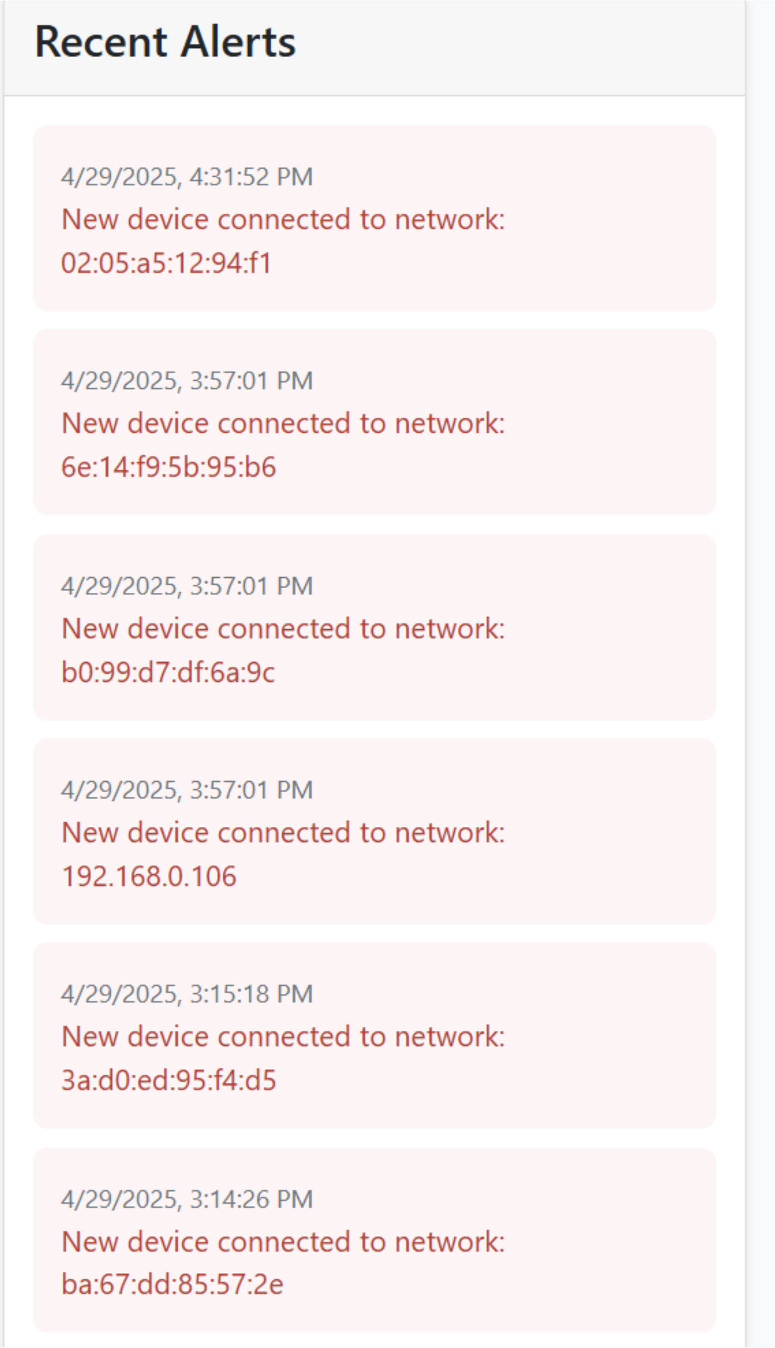

Figure 9: The Alert Interface

Figure 10 demonstrates the real time device location tracking using a 3d map where red colour represents router, green colour represents approved devices and yellow colour represents pending devices.

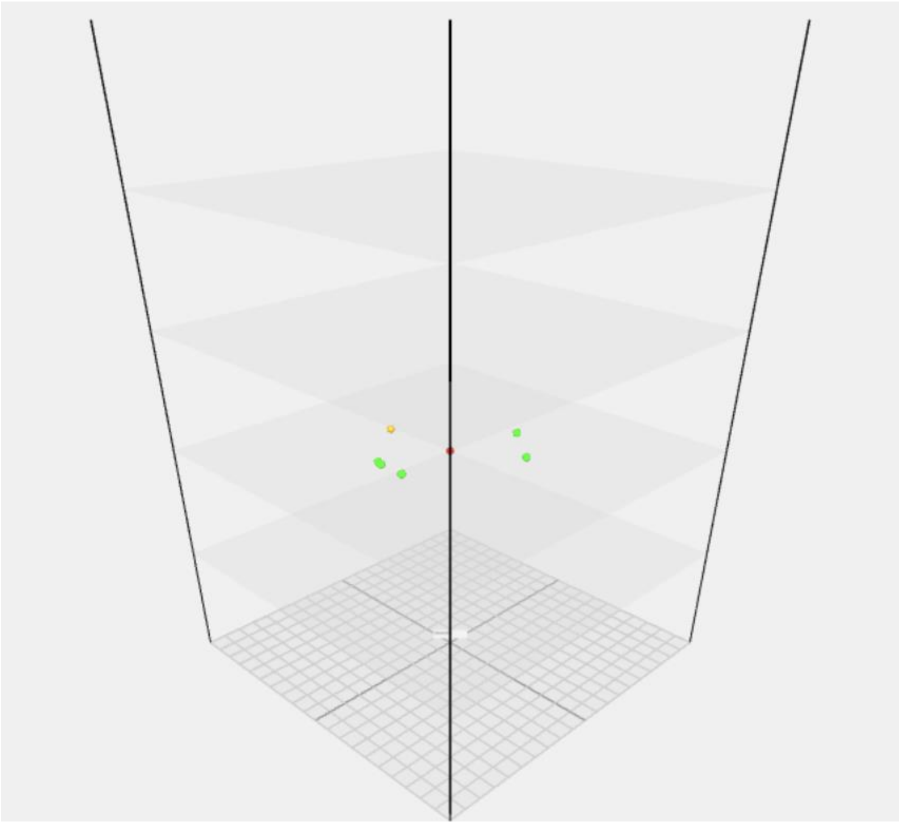

Figure 10: 3D Location Tracking

As shown in Figure 11, the complete real-time tracking of each device is shown along with the complete details of each device.

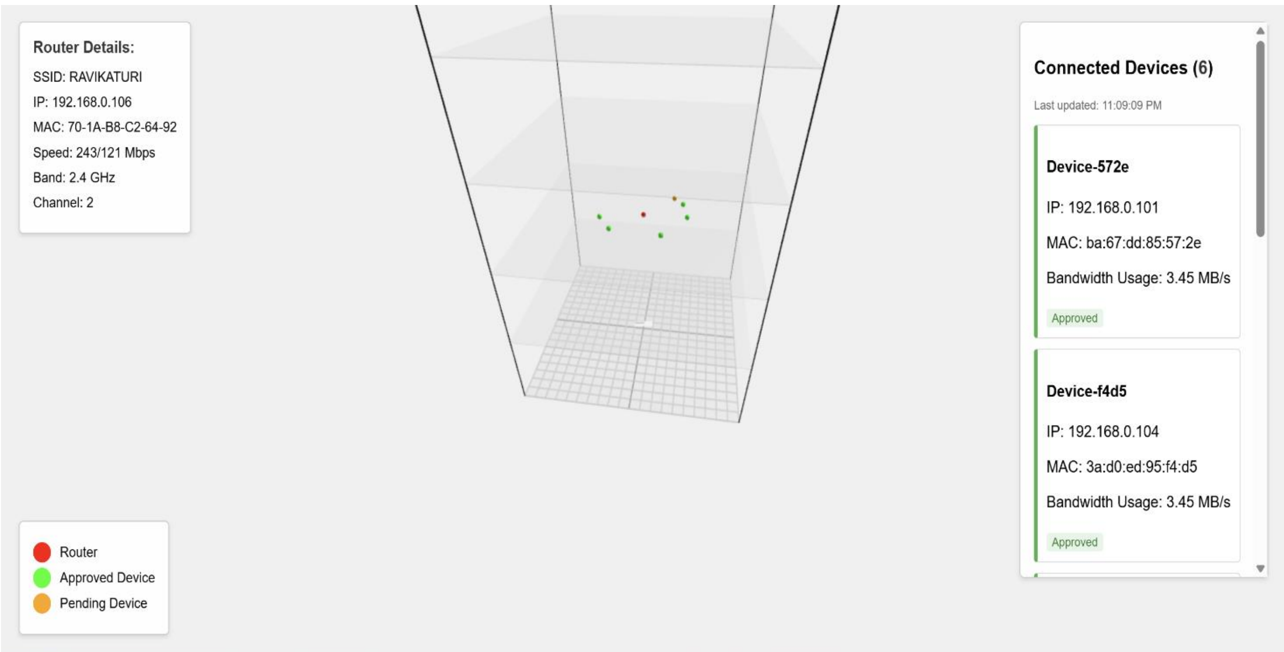

Figure 11: The complete location interface

7. CONCLUSION AND FUTURE SCOPE

7.1 Conclusion

The NetFlow Analyzer project delivers an effective and comprehensive WiFi management solution designed specifically for campus networks. By integrating bandwidth monitoring, indoor device location tracking, and security alert mechanisms, the system provides real-time insights into network usage and device behavior. Built using Python, Flask, MongoDB, Scapy, and React.js, the platform combines backend scalability with a responsive and intuitive frontend interface. The use of RSSI-based triangulation for indoor tracking and admin-controlled device management improves both transparency and security. Security alerts for abnormal data usage and unauthorized connection attempts ensure proactive threat detection. Overall, the NetFlow Analyzer fulfills its objective of providing a centralized, intelligent, and secure framework for managing campus WiFi infrastructure.

7.2 Future Scope

- **Mobile Dashboard:** Develop a mobile application version for administrators and users to monitor network activity and alerts in real-time.
- **Advanced Triangulation Models:** Integrate machine learning-based signal mapping for improved accuracy in indoor location tracking.
- **Network Health Analytics:** Incorporate dashboards for monitoring network latency, packet loss, and throughput trends.
- **Role-Based Access Control (RBAC):** Implement fine-grained access controls for different user roles (e.g., IT admin, department head).
- **Integration with Existing IT Systems:** Enable compatibility with existing network hardware, SIEM tools, and institutional login systems.
- **Automated Anomaly Detection:** Use AI models to detect and flag unusual network behavior without predefined thresholds.
- **User Notifications:** Send alerts to users when their device exceeds bandwidth limits or enters restricted zones.
- **Offline Packet Capture Support:** Store and analyze packet data locally during internet downtime with automated sync to the cloud once reconnected.

REFERENCES

1. T. Amano, H. Yamaguchi and T. Higashino, "Outdoor-Indoor Bridging for Reference-Point-Free 3D WiFi AP Localization," in IEEE Open Journal of the Computer Society, doi: 10.1109/OJCS.2025.3566774.
   Paper Link
2. S. Manzoor, S. Azam, R. Wójcik, and J. Domżał, "SWIF: Smart Wi-Fi Integration Framework for

Mobility and Resource Optimization," 2024 IEEE 21st Annual Consumer Communications & Networking Conference (CCNC), 2024, pp. 1–6.
Paper Link
3. Z. Huang, M. Valkama, J. Zhang, M. Xu, C. Yin, and M. Guan, "WiLoc: Encoding-based WiFi Indoor Localization," 2024 IEEE International Conference on Communications (ICC), 2024, pp. 1–6.
DOI: 10.1109/IPIN62893.2024.10786150
4. S. A. Kharmeh, E. Natsheh, R. Nasrallah, and M. Masri, "Triangulation-Enhanced WiFi-Based Autonomous Localization and Navigation System: A Low-Cost Approach," 2024 IEEE International Conference on Smart Computing (SMARTCOMP), 2024, pp. 1–7.
DOI: 10.1109/REM63063.2024.10735691
5. A. Collaguazo, R. Estrada, I. Valeriano, and J. Tomala, "WiFi-Crowd Spy: A Novel Crowd-Counting System," 2022 IEEE International Conference on Communications Workshops (ICC Workshops), 2022, pp. 1–6.
DOI: 10.1109/ICECET55527.2022.9873522
6. X. Wu, M. D. Soltani, L. Zhou, M. Safari and H. Haas, "Hybrid LiFi and WiFi Networks: A Survey," in IEEE Communications Surveys & Tutorials, vol. 23, no. 2, pp. 1398-1420, Secondquarter 2021, doi: 10.1109/COMST.2021.3058296.
Paper Link
7. W. L. Ogogo, "Real-Time Monitoring of Network Devices: Its Effectiveness in Enhancing Network Security," ResearchGate, 2021.
DOI: 10.37284/eajit.3.1.153
8. J. Saldana et al., "Attention to Wi-Fi Diversity: Resource Management in WLANs With Heterogeneous APs," in IEEE Access, vol. 9, pp. 6961-6980, 2021
Paper Link
9. S. Jahagirdar, A. Ghatak, and A. A. Kumar, "WiFi Based Indoor Positioning System Using Machine Learning and Multi-Node Triangulation Algorithms," 2020 IEEE International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), 2020, pp. 1–5.
DOI: 10.1109/ICCCNT49239.2020.9225350
10. I. P. Mohottige, H. H. Gharakheili, A. Vishwanath, S. S. Kanhere, and V. Sivaraman, "Evaluating Emergency Evacuation Events Using Building WiFi Data," 2020 IEEE International Conference on Computer Communications (INFOCOM), 2020, pp. 1–9.
DOI: 10.1109/IoTDI49375.2020.00018
11. R. Ahmad, M. D. Soltani, M. Safari, A. Srivastava and A. Das, "Reinforcement Learning Based Load Balancing for Hybrid LiFi WiFi Networks," in IEEE Access, vol. 8, pp. 132273-132284, 2020
DOI: 10.1109/ACCESS.2020.3007871
12. H. Liu, X. Pu, Y. Chen, J. Yang and J. Chen, "User-Centric Access Scheme Based on Interference Management for Indoor VLC-WIFI Heterogeneous Networks," in IEEE Photonics Journal, vol. 12, no. 4, pp. 1-12, Aug. 2020, Art no. 7903712
DOI: 10.1109/JPHOT.2020.3002246
13. X. Wu and H. Haas, "Load Balancing for Hybrid LiFi and WiFi Networks: To Tackle User Mobility and Light-Path Blockage," in IEEE Transactions on Communications, vol. 68, no. 3, pp. 1675-1683, March 2020
DOI: 10.1109/TCOMM.2019.2962434

14. L. Yang, Q. Zhang, W. Zhang, L. Deng and H. Yang, "Resource Allocation for Hybrid Visible Light Communications (VLC)-WiFi Networks," in IEEE Access, vol. 8, pp. 176588-176597, 2020
DOI: 10.1109/ACCESS.2020.3026388
15. P. Li, Y. Yan, P. Yang, X. -Y. Li and Q. Lin, "Coexist WiFi for ZigBee Networks With Fine-Grained Frequency Approach," in IEEE Access, vol. 7, pp. 135363-135376, 2019
DOI: 10.1109/ACCESS.2019.2941963
16. E. Markova, D. Moltchanov, I. Gudkova, K. Samouylov and Y. Koucharyavy, "Performance Assessment of QoS-Aware LTE Sessions Offloading Onto LAA/WiFi Systems," in IEEE Access, vol. 7, pp. 36300-36311, 2019
DOI: 10.1109/ACCESS.2019.2905035
17. D. Kim, K. Lee and J. -H. Kim, "Nash Bargaining Approach for Fair and Efficient LTE–WiFi Aggregation," in IEEE Access, vol. 7, pp. 117176-117187, 2019
DOI: 10.1109/ACCESS.2019.2936857
18. T. Petrovic, K. Echigo and H. Morikawa, "Detecting Presence From a WiFi Router's Electric Power Consumption by Machine Learning," in IEEE Access, vol. 6, pp. 9679-9689, 2018
DOI: 10.1109/ACCESS.2018.2797881
19. G. Z. Khan, E. -C. Park and R. Gonzalez, "M3-Cast: A Novel Multicast Scheme in Multi-Channel and Multi-Rate WiFi Direct Networks for Public Safety," in IEEE Access, vol. 5, pp. 17852-17868, 2017
DOI: 10.1109/ACCESS.2017.2749482.
20. M. Passafiume, S. Maddio, M. Lucarelli, and A. Cidronali, "An Enhanced Triangulation Algorithm for a Distributed RSSI-DoA Positioning System," 2016 IEEE Sensors Applications Symposium (SAS), 2016, pp. 1–6.
Paper Link