APPLYING THE ADDIE MODEL TO DEVELOP A CAPSMAN-BASED WIRELESS NETWORK MANAGEMENT AND TRAFFIC MONITORING SYSTEM

Tri Septianto[1)], David Nurdiansyah[2)], Hendrik Kusbandono[3)], Angger Binuko Paksi[4)]
[1, 2, 3, 4)]Jurusan Teknik, Politeknik Negeri Madiun, Madiun
e-mail: triseptianto@pnm.ac.id[1)], dafidnurdiansyah74@gmail.com [2)], h3ndrik57@pnm.ac.id[3)], angger.binuko@pnm.ac.id[4)]

ABSTRACT

The advancement of information technology demands wireless networks that are not only stable but also centrally manageable and efficient. In many medium-scale organizations, such as XYZ Company, manual access point management leads to inefficiencies, configuration inconsistencies, and difficulties in monitoring network traffic. To address these issues, this study proposes the development of a wireless network management system based on CAPsMAN (Centralized Access Point Management) with real-time traffic monitoring features, implemented using the Laravel framework. The development process follows a systematic approach using the ADDIE model (Analysis, Design, Development, Implementation, Evaluation), providing a structured methodology for identifying user requirements, designing system architecture, developing the solution, implementing it in a real-world environment, and evaluating system performance. The developed system is integrated with the MikroTik API for direct data retrieval and utilizes a MySQL database for storing traffic history. Implementation results indicate that the system successfully enables centralized management of six access points, simplifies bulk configuration, and provides an intuitive web interface for real-time monitoring of connected devices, network status, and traffic reports. Quantitative monitoring shows peak traffic of 2 MBps (upload) and 21 KBps (download), with data logged every 30 minutes for historical analysis. Prior to implementation, network administrators relied on manual, device-by-device configuration via Winbox without centralized visibilityleading to inconsistent policies and delayed troubleshooting. Post-implementation, all management tasks are unified, and real-time dashboards reduce monitoring latency to near zero. Black-box testing confirms that all system functions operate according to specifications. The application of the ADDIE model proves effective in ensuring the system's quality, reliability, and sustainability. This research demonstrates that ADDIE can be successfully adapted beyond educational contexts its novel application in network system development ensures a user-centered, traceable, and evaluable engineering process, which is rarely documented in existing literature.

Keywords*: ADDIE model, CAPsMAN, wireless network management, traffic monitoring, Laravel, MikroTik*

I. INTRODUCTION

The rapid advancement of information technology in the digital era has driven a growing demand for wireless networks that are not only stable and reliable but also efficiently manageable. Wireless networks have become the backbone of various activities in educational environments [1] , offices [2] , and industrial settings, as they provide cable-free access that supports user mobility [3].

However, as the number of devices and network coverage expands, managing distributed access points configured manually has become a significant challenge, particularly in medium-scale organizations such as XYZ Company, which manages six access points separately. The process of configuring and maintaining each device individually is not only time-consuming but also prone to technical errors and network instability caused by inconsistent settings. This situation reflects a common issue: the lack of an affordable, structured, and user-friendly centralized management system. As a solution, MikroTik's CAPsMAN (Centralized Access Point Management) offers a centralized management approach that enables administrators to control multiple access points from a single management point [4] [5] . With CAPsMAN, configuration distribution, performance monitoring, and network maintenance can be performed collectively and uniformly [6].

However, effective management must be supported by an intuitive monitoring system capable of presenting data in real time. Currently, network monitoring at XYZ Company is conducted using the Winbox application, which operates locally and lacks comprehensive data visualization capabilities. Therefore, a web-based monitoring system is needed to dynamically display traffic information, connected devices, and network performance in a way that is both accessible and user-friendly.

In this study, the system was developed using the Laravel framework, which is known for its scalability

and efficiency [7][8], integrated with the MikroTik API for direct data retrieval from network devices, and a MySQL database [9] for storing traffic history.

To ensure a systematic and controlled development process, this study employs the ADDIE model as the methodological framework. Although ADDIE was originally designed for instructional material development [10] , its five phases, namely analysis, design, development, implementation, and evaluation, demonstrate strong alignment with the software development lifecycle [11].

This approach enables the development team to thoroughly identify user needs, design an appropriate system architecture, develop the solution incrementally, test the implementation in a real-world environment, and objectively evaluate system performance. As such, ADDIE proves not only relevant but also instrumental in ensuring that the developed system meets key technical requirements [12], usability aspects [13], dan and operational sustainability [14].

The novelty of this research lies in two main aspects. First, it adapts the ADDIE model, a methodology traditionally used in instructional design for the systematic development of a wireless network management system, thereby demonstrating its applicability and effectiveness in the field of network engineering. Second, it integrates CAPsMAN with a real-time web-based traffic monitoring dashboard using Laravel and MikroTik API, offering a practical, centralized, and user-friendly solution tailored for medium-scale organizations that lack enterprise-grade network management tools. To the best of our knowledge, no prior study has combined the ADDIE framework with CAPsMAN-based network management in this manner.

II. Analysis

The analysis phase in the ADDIE model serves as the foundational step in the system development process, aiming to identify problems, user needs, and the feasibility of the proposed solution. In this study, the analysis phase was conducted through a combination of direct observation, interviews with the network administrator, and a literature review to gain a thorough understanding of the current wireless network infrastructure at the main office of XYZ Company.

We conducted observations at the company's main office, spanning four floors and operating multiple MikroTik Access Points (APs) for wireless coverage. The results revealed that all APs currently operate in standalone mode, without centralized management. This setup leads to increased complexity in configuration, maintenance, and network monitoring, particularly when policy changes or technical difficulties require rapid intervention across multiple access points simultaneously.

Interviews were conducted with a staff member responsible for network administration to gain more profound insight into operational challenges. Based on the interview, it was found that the wireless network at the office is divided into two segments: (1) a public network accessible to guests, restricted to general internet services such as Google and YouTube, and (2) an internal network used by employees to access company resources. Currently, network security relies solely on SSID passwords, without the implementation of Access Control Lists (ACL) or stricter network segmentation.

Furthermore, the network administrator reported that AP management is still performed manually via the Winbox interface, which is inefficient for a growing network environment. To address this limitation, the company plans to implement MikroTik CAPsMAN as a centralized management solution.

Regarding traffic monitoring, the company currently relies on MikroTik's built-in tools through Winbox, which provide technical data but are not user-friendly for non-technical personnel. The administrator expressed the need for an interactive dashboard capable of displaying key information in real time, such as the number of active APs, connection status, number of connected devices, and bandwidth traffic (Tx/Rx). Such a dashboard is expected to simplify network performance monitoring and support faster decision-making in network management.

Based on the analysis findings, three primary requirements were identified: (1) a centralized wireless network management system using CAPsMAN; (2) improved security mechanisms, including network segmentation and access control lists; and (3) the development of a real-time monitoring dashboard for visual and intuitive network performance tracking.

Therefore, the development of a CAPsMAN-based wireless network management and monitoring system is considered essential to enhance network efficiency, security, and service quality within the organizational environment.

III. Design

This phase comprises three interdependent stages: network topology, infrastructure diagram, and use case diagram. These stages are designed to systematically plan and organize system components to achieve optimal efficiency and effectiveness.

A. Network topology

Network topology is important because it determines the most effective way to connect devices, manage resources, and enhance security in a communication system. In this study, a star topology is employed, which is further elaborated in Figure 1.

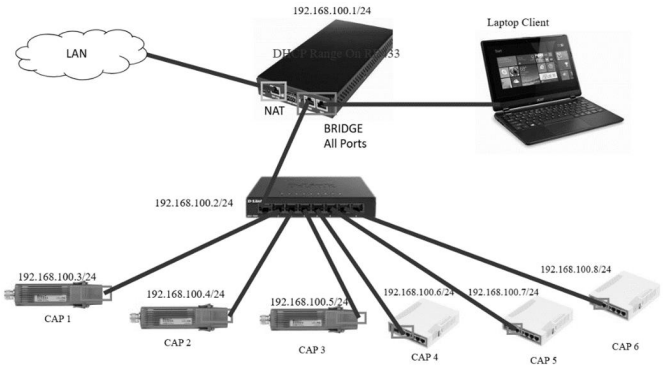

Figure. 1. Network Topology

Figure 1 illustrates the network topology implemented to deploy the CAPsMAN (Centralized Access Point Management) feature. In this topology, the network administrator configures CAPsMAN using a single system manager, specifically a MikroTik RB433AH router. This system manager plays a critical role in controlling and managing all access points that connect as Controlled Access Points (CAPs).The router acting as the system manager is connected to a D-Link DGS-108 8-Port Gigabit Desktop Switch via a bridge port. This bridge port serves to propagate the configuration from the system manager to all CAPs connected within the network.The setup includes six CAPs functioning as access points, consisting of: (1) three units of RBMetalG-52SHPacn devices, and (2) three units of RB962UiGS-5HacT2HnT devices. Each CAP device is assigned a unique IP address and is interconnected through the switch.Client devices connecting to the network receive local IP addresses distributed by the router, which performs Network Address Translation (NAT) and acts as a Dynamic Host Configuration Protocol (DHCP) server to assign IP addresses dynamically to client devices.

Figure 2 presents the infrastructure diagram of the CAPsMAN-based network management and monitoring system within a Local Area Network (LAN). In this architecture, the MikroTik API and a MySQL database facilitate communication between CAPsMAN and the web-based monitoring dashboard, which administrators use.

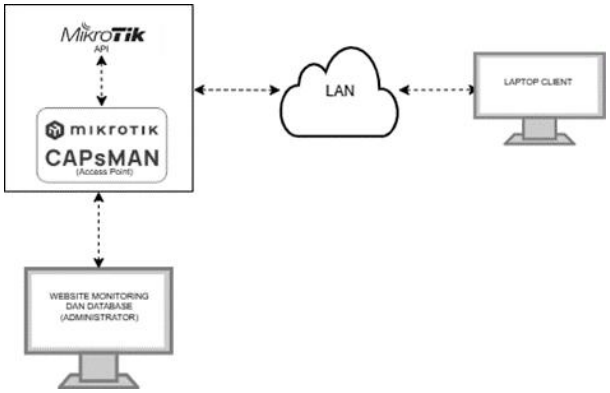

Figure. 2. The Infrastructure Design

B. *The Infrastructure Design*

The infrastructure design was developed to clearly depict the structure and components of the implemented application. The primary objective of this design is to provide a comprehensive understanding of how various application elements interact and function as an integrated system. The architectural layout of the application is illustrated in Figure 2.

The Laravel framework enables the monitoring dashboard to display the required data through this communication process. Client laptops connect to access points managed by CAPsMAN, allowing client devices to obtain network connectivity. Data exchange between CAPsMAN and the monitoring dashboard is conducted via the MikroTik API, which retrieves essential information such as access point status, bandwidth usage, and connected devices.

Laravel serves as the backend system that receives data from CAPsMAN through the MikroTik API. This data is utilized to monitor access point status, bandwidth consumption, and connected devices. The collected data is then stored in a MySQL database, which acts as the primary data repository. The monitoring dashboard, accessed by administrators through a web browser, connects to the Laravel backend and the database via a local IP address. The dashboard provides real-time network information, offering administrators a clear overview of network performance and connected devices.

C. *The Use Case Diagram*

The use case diagram is employed to identify the entities or actors that interact with the application, as

well as the access rights and functionalities available within the system. By constructing the use case diagram, the author can clearly illustrate the relationships between users, the system, and the various functions provided. This diagram plays a crucial role in defining different usage scenarios, which in turn serve as a clear guide for subsequent application development phases. The use case diagram for the proposed application is presented in Figure 3.

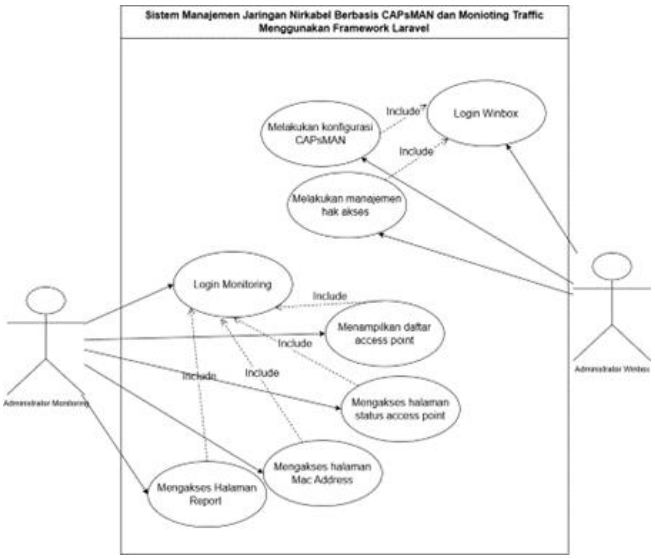

Figure. 3. The Use Case Diagram

D. Blackbox Testing Plan

In the design phase of the ADDIE model, in addition to designing the system architecture and user interface, a functional testing plan is also developed to ensure that the CAPsMAN-based wireless network management system operates in accordance with specified requirements. One of the planned testing approaches is black box testing [15], which focuses on the system's inputs and outputs without examining its internal processes. This method is selected because it effectively evaluates system performance from the end-user perspective, particularly that of the network administrator, who is the primary user of the system.

The black box testing is designed to verify seven core system features: 1) CAPsMAN activation; 2) MAC address-based access control; 3) administrator authentication process; 4) traffic monitoring via the dashboard; 5) management of MAC address lists; 6) monitoring of access point status; and 7) network traffic reporting. Each feature is evaluated using specific test scenarios to ensure that the system responds as expected.

The first test focuses on the CAPsMAN activation feature. Access points intended for centralized management must be enabled as CAPs (Controlled Access Points) through the Wireless menu. If configured correctly, the access point is expected to successfully connect to the CAPsMAN controller and receive configuration settings automatically. Conversely, if CAP mode is not activated, the access point will not communicate with the controller and thus cannot be centrally managed. This scenario ensures that only properly configured devices can join the managed network.

Next, testing is conducted on the MAC address-based access list feature. Administrators can add user device MAC addresses to the access list. Devices with registered MAC addresses should be allowed to connect to the network through CAPsMAN-managed access points, while those with unregistered addresses must be denied access. This test confirms that network access is strictly controlled in accordance with security policies.

The administrator login process is also tested. When the administrator correctly enters the IP address, username, and password, the system should redirect them to the monitoring dashboard. If any required field is left blank, the system must display a validation error message: "validation.required." If incorrect credentials are provided, the system should present a login failure page with the message "Authentication Failed." This demonstrates that the system properly handles various input conditions securely and provides appropriate user feedback.

For the dashboard monitoring feature, the system must display real-time network traffic data upon accessing the dashboard menu. The administrator can select a specific access point from the "Select AP" dropdown menu, and the system should then present graphical and numerical data on bandwidth usage for the selected access point. If no access point is selected, the system automatically displays data from the first available access point, ensuring that the dashboard remains informative even without additional user input.

Testing is also performed on the MAC address list management feature. When the administrator clicks the delete icon on a specific entry, the corresponding MAC address must be removed from both the displayed list and the access list configuration on the CAPsMAN side. This verifies that changes made through the user interface are synchronized with the backend system,

ensuring consistency between the frontend interface and network configuration.

The access point status monitoring feature is evaluated by accessing a dedicated menu that displays a list of connected access points. The system must accurately indicate the status of each access point, whether it is online or offline. This information is crucial for administrators to monitor the overall health and availability of the network infrastructure.

Finally, testing is conducted on the reporting feature. When the administrator opens the Report menu, the system should display network traffic graphs. The feature also allows the selection of a specific time range, enabling customized reporting based on desired periods. The system is expected to present historical data accurately, supporting long-term network usage analysis.

This testing plan is a critical component of the design phase, as it defines how the system will be evaluated before proceeding to further development. The test results will be used to refine the system during the development phase and will serve as the basis for performance evaluation in the final evaluation phase of the ADDIE model.

IV. DEVELOPMENT

The Development phase in the ADDIE model represents the transformation of the system design into a functional implementation of a CAPsMAN based wireless network management system. Over a period of seventy working days, development was carried out incrementally by dividing the work into seven main modules, each taking approximately ten days to complete. This modular approach enabled the team to focus deeply on each component, ensuring stability, reliability, and seamless integration between features before progressing to the next stage.

Development began with establishing the system foundation using the CAPsMAN feature on MikroTik devices. During the first ten days, initial configuration was performed on the System Manager to enable remote management and accurate time synchronization through the NTP protocol. CAPsMAN was then configured as a central controller responsible for managing all access points in the network. Through this mechanism, each access point could register automatically, receive uniform configurations, and be managed centrally. This process formed the foundation for all advanced features, ensuring consistency and efficiency in wireless network administration.

Once the core system was functioning properly, development proceeded with the implementation of security features based on access lists. Over the next ten days, the system was configured to restrict network access based on device MAC addresses. This functionality was implemented using the Security Profile in CAPsMAN, allowing administrators to specify permitted or blocked devices. Testing results indicated that the system successfully rejected connection attempts from unregistered devices, thereby enhancing network security and preventing unauthorized access.

In the following stage, the system was enhanced to monitor data traffic on each access point. For ten days, integration with the MikroTik API was performed to periodically retrieve throughput, bandwidth usage, and packet rate data. A Python based automated script was developed to poll data every five minutes, ensuring that the information remained up to date. The collected data was then processed and displayed as dynamic graphs on a web based dashboard, enabling administrators to monitor network load in real time and quickly identify traffic spikes or potential congestion.

Development also included monitoring the number of active access points in the network. Using the MikroTik API, the system accessed the CAPsMAN registration table to count connected APs at regular intervals. This information was presented as real time numerical values and supplemented with daily trend graphs, providing a visual representation of device connection stability and consistency. This feature proved highly useful for evaluating network performance, especially in environments with multiple access points distributed across various locations.

In addition to monitoring the total number of APs, the system was further developed to track the connection status of each access point individually. Over ten days, a monitoring script was created to determine whether each AP remained registered as active in the system. The results were displayed on the dashboard using color coded indicators, with green representing an online status and red indicating offline. This allowed administrators to immediately identify any malfunctioning access points without manually accessing each device, significantly accelerating fault detection and resolution.

Aligned with device management and network audit requirements, the system was also equipped with a feature to monitor MAC addresses of connected client devices. MAC address data was retrieved from the wireless registration table via the MikroTik API and presented in a sorted and searchable table format. This feature enhanced transparency regarding active network users, supported the identification of suspicious devices, and facilitated troubleshooting in cases of connectivity or performance issues.

During the final development stage, we enhanced the system with historical reporting capabilities for network traffic. We integrated a MySQL database to store bandwidth usage data from each access point at regular intervals. This historical data was used to generate daily, weekly, and monthly graphical reports, which administrators could analyze to understand usage patterns and support capacity planning.

Overall, the Development phase successfully delivered an integrated, secure, and easily monitorable prototype of the wireless network management system. The modular approach enabled structured development with minimal interference between components. The integration of the MikroTik API as the primary data source, combined with a responsive and informative dashboard, ensured that the system was ready for further testing in real world environments during the Implementation phase.

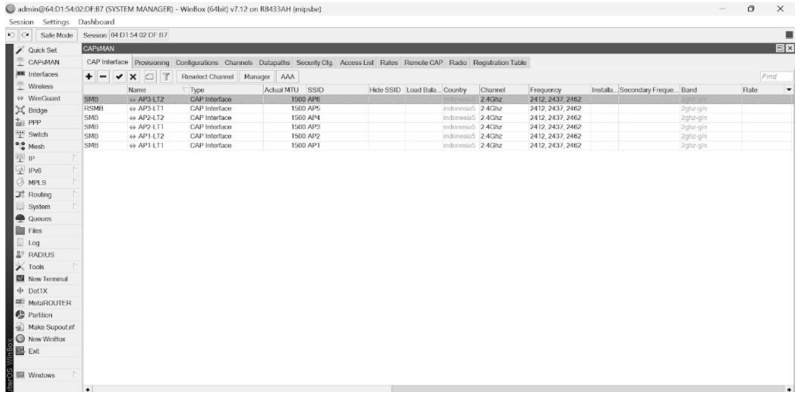

Figure. 4. CAPsMAN Configuration Results

V. IMPLEMENTATION

The implementation phase was conducted in the main office environment of XYZ Company as a practical step in deploying the CAPsMAN-based wireless network management system integrated with a monitoring dashboard. The primary objective of this phase was to realize the previously developed system design and ensure its functional operation in a real-world working environment. The implementation process included network infrastructure configuration, centralized access point management, dashboard application development, and system integration to enable real-time network monitoring by administrators.

Implementation began with the configuration of CAPsMAN on the main router, which serves as the central network management unit. The router was configured using MikroTik RouterOS, with the System Manager feature enabled to ensure centralized control over all network devices. Technical steps included assigning IP addresses, setting up DHCP Client and DHCP Server for automatic IP address distribution, configuring the gateway, and implementing firewall rules to maintain network security and integrity. Additionally, configurations were set in the Configurations and Provisioning tabs within the CAPsMAN module, enabling bulk and automatic deployment of settings to all access points. Through this mechanism, every access point connected to the network received predefined configurations consistently, eliminating the need for manual individual setup. After completing the configuration process, access points in CAP mode successfully connected to CAPsMAN, and the CAP Interface displayed their status. The results confirmed that all access points were properly connected and centrally managed, as visually demonstrated in Figure 4.

Concurrent with network infrastructure readiness, the monitoring system was developed using the Laravel framework and hosted locally via Laragon. The dashboard was designed as a visual control center for network administrators, enabling real-time monitoring of network conditions. System integration was achieved by utilizing the MikroTik API to retrieve data from network devices, which was then transmitted to a MySQL database for storage and processing. This data was subsequently visualized through the Laravel web interface, allowing administrators to access critical information without directly logging into the router. This integration serves as the backbone of the monitoring system, ensuring that network data remains accurate, up-to-date, and easily interpretable.

The developed dashboard includes several key features that support effective network management. The first is traffic monitoring, which displays real-time

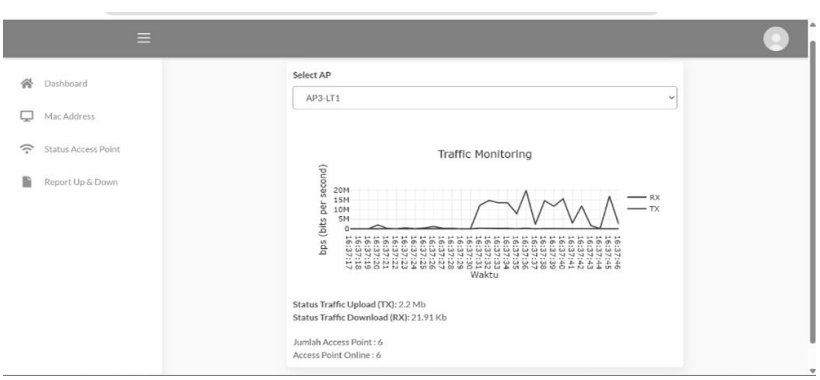

Figure. 5. Real-Time Traffic Monitoring

graphs of upload and download traffic for each access point. Through this feature, administrators can directly observe bandwidth usage patterns, including peak usage reaching 2 MBps for upload and 21 KBps for download, as shown in Figure 5.

The second feature is access point status, which presents a list of all access points connected to CAPsMAN along with their connection status. As shown in Figure 6, all six installed access points—including AP2 LT2, AP3 LT1, and AP3 LT2—are displayed as online, indicating that the network is fully operational and stable.

The third feature is MAC address management, which allows administrators to view the list of devices

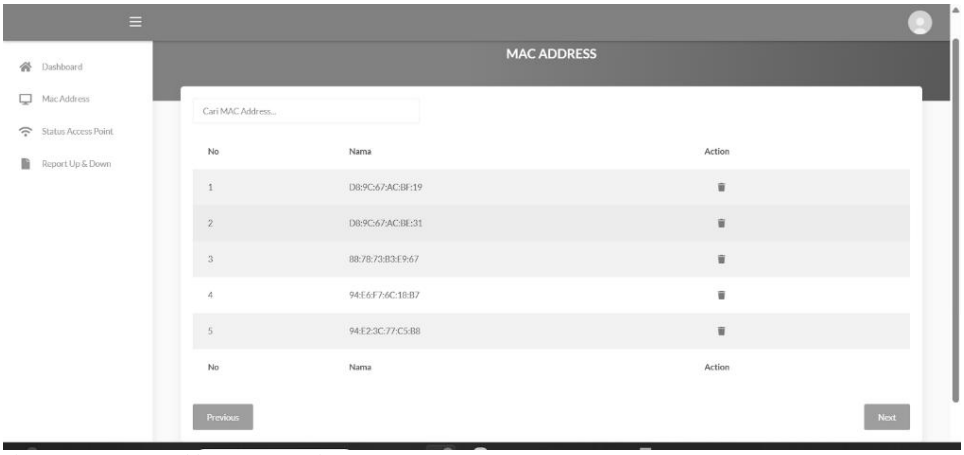

Figure. 6. MAC Address Access List

permitted to connect to the network. As illustrated in Figure 6, five MAC addresses are registered in the access list, and only devices with these identifiers are allowed to connect, thereby enhancing network security. The system also provides a delete function to remove devices that are no longer authorized.

The final feature is traffic reporting, which presents historical bandwidth usage data over a 24-hour period, from 01.00 to 24.00 WIB. Data is collected every 30

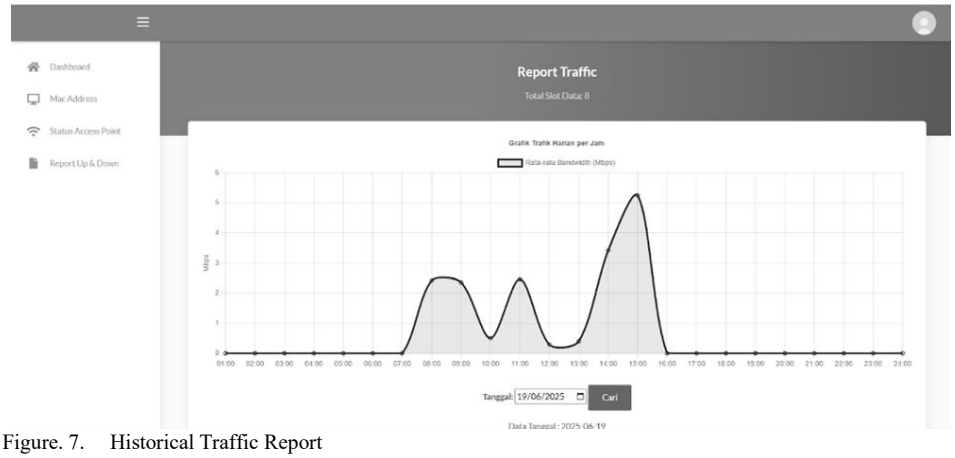

Figure. 7. Historical Traffic Report

minutes and stored in the database, resulting in two data entries per hour. This enables detailed analysis of network traffic patterns. Administrators can also select different dates to view reports from previous days, supporting capacity planning and network performance evaluation, as demonstrated in Figure 7.

The primary users of this system are network administrators who access the dashboard through a login process using predefined credentials. After successful authentication, administrators can perform various tasks such as monitoring connection status, viewing connected devices, analyzing bandwidth usage, and exporting reports in user-friendly formats. To ensure smooth adoption, a brief training session was provided to administrators covering dashboard navigation, data interpretation, and response procedures for specific network conditions.

We encountered and resolved several technical challenges during the implementation process. One issue was delayed configuration propagation from CAPsMAN to access points located at greater physical distances, resulting in slightly delayed synchronization. This was addressed by optimizing radio settings and ensuring stable signal strength between the router and access points. Another challenge involved inconsistent data retrieval from the MikroTik API, which was resolved by setting a fixed polling interval of every 30 minutes. Although the system is hosted locally using Laragon and not accessible from outside the network,

this limitation is not problematic, as the system was specifically designed for internal company use.

Overall, the implementation phase successfully realized a functional and integrated wireless network management and monitoring system. All access points are centrally managed through CAPsMAN, while the Laravel-based dashboard effectively presents both real-time and historical data in an informative and user-friendly interface. The success of this implementation serves as a solid foundation for the subsequent evaluation phase, where system performance will be assessed in terms of reliability, efficiency, and operational benefits for enterprise network management.

VI. Evaluation

The evaluation phase is a critical stage in the ADDIE model, as it ensures that the developed system operates according to its intended objectives and effectively meets user requirements. In this study, an evaluation was conducted to assess the functionality and reliability of the CAPsMAN-based wireless network management system, including its features for monitoring network traffic. The primary method employed was functional testing using a black box approach, which focuses on system inputs and outputs without examining internal structures or source code. This method was selected because it evaluates the system from the end user's perspective, providing insight into how well the system performs expected operational tasks.

Testing was carried out on several core system features, including integration with CAPsMAN, MAC address-based access control, the administrator authentication process, the monitoring dashboard display, access point status management, and network traffic reporting. Each test scenario was designed to simulate real-world activities typically performed by network administrators. Expected outcomes were defined in advance, and actual results were observed directly during testing. Overall, the system demonstrated excellent and consistent performance.

One example is the activation of the CAP feature on an access point. When enabled, the device successfully connected to the CAPsMAN controller and received centralized configuration, indicating that network management is fully integrated. Conversely, when the CAP feature was not activated, the access point failed to connect to the controller and remained undetected in the system, which aligns with the intended mechanism.

The MAC address-based access control feature also proved effective. When an administrator registered a user's MAC address in the allowed list, the device could connect to the network seamlessly. However, when a device with an unregistered MAC address attempted to connect, the system automatically rejected the connection, demonstrating that the whitelist-based security mechanism functions as intended. This capability is essential for preventing unauthorized devices from accessing the network.

Testing of the login process yielded positive results as well. When the administrator entered the correct IP address, username, and password, the system granted access and redirected the user to the dashboard. If any required field was left blank, the system displayed the warning message *"validation.required."* When incorrect credentials were entered, the user was redirected to a failure page with the message *"Authentication Failed."* These system responses provide clear feedback, helping users identify and correct input errors.

The monitoring dashboard successfully displayed network information in real time. Upon accessing the dashboard, the system immediately presented a list of access points managed by CAPsMAN, along with their connection status and traffic data. When the administrator selected a specific access point from the *"Select AP"* menu, the corresponding traffic data was displayed in both graphical and numerical formats, facilitating quick network analysis. Even when no selection was made, the system automatically displayed data from the first registered access point, ensuring the interface remained informative and never blank. This functionality supports efficient network management by providing a comprehensive and immediate overview of network performance.

The MAC address management feature also performed as expected. Administrators were able to view the list of permitted devices and remove MAC addresses no longer required. Once removed, devices with those addresses could no longer connect to the network, indicating that configuration changes were applied immediately. Additionally, the access point status feature successfully displayed the list of devices along with their connection status—online or offline—enabling early detection of malfunctioning units. The reporting feature also effectively displayed historical network traffic data in graphical form. Users could select specific dates to view bandwidth usage trends, which is highly beneficial for network performance evaluation and future capacity planning.

Overall test results indicate that all system features functioned as intended. No critical errors or significant discrepancies between expected and actual outcomes were observed. All test scenarios produced the desired results, confirming that the development process was well executed and that the system is ready for deployment in operational environments.

The success of this evaluation also reflects the effectiveness of the ADDIE model implementation, where each phase analysis, design, development, and

implementation was carried out systematically and cohesively. This phased approach enabled clear identification of requirements, precise solution design, and the development of a reliable, user-responsive system.

Although the system demonstrated excellent performance in the testing environment, several limitations should be acknowledged. Currently, the system is highly dependent on MikroTik hardware and the RouterOS operating system, which may limit its flexibility in heterogeneous network environments. Future enhancements could include integration with devices from other vendors, development of a mobile-responsive interface, and the addition of predictive analytics features to support early detection of network anomalies.

Nevertheless, the current evaluation results sufficiently demonstrate that the developed CAPsMAN-based network management system achieves its primary objectives: providing a centralized, secure, and easily monitorable solution for wireless network management.

VII. Evaluation

To better illustrate the system's impact, Table 1 compares network management conditions before and after the implementation of the proposed CAPsMAN-based system.

VIII. Conclusion

It can be concluded that the CAPsMAN-based Wireless Network Management System integrated with Network Traffic Monitoring using the Laravel framework has been successfully developed and effectively implemented to enhance centralized access point management. The configuration of the CAPsMAN feature on the main router, applied to all six access points, enables centralized network management, facilitating streamlined administration, bulk configuration, and real-time monitoring of device status.

In addition, the web-based network traffic monitoring system developed using the Laravel framework has been successfully integrated with the MikroTik API and MySQL database. This integration allows the system to display real-time information on network traffic, access point status, connected devices (MAC addresses), and traffic reports in the form of informative and easily understandable graphical representations for network administrators.

The results of black box testing indicate that all system functions operate in accordance with the predefined test scenarios. All modules—from authentication and monitoring to reporting—produce outputs that meet expectations, demonstrating that the system performs optimally and is ready for deployment in real operational environments.

Furthermore, the system's development process followed the ADDIE model (analysis, design, development, implementation, and evaluation) in a systematic and structured manner. The analysis phase helped identify user needs and existing network management challenges. The design phase produced a clear system architecture and a user-friendly interface. The development phase resulted in a fully functional system, followed by implementation in a real network environment and evaluation to ensure system reliability and performance. The successful application of the ADDIE model in this context demonstrates its relevance not only in educational or training settings but also in the development of technical and computer networks.

Thus, this study has not only succeeded in delivering an efficient and measurable network management solution but also confirms that the ADDIE model can be effectively adapted and applied in the development of network-based technology systems. It provides a structured, user-centered, and objectively evaluable approach, making it highly suitable for complex system development in real-world IT environments.

Reference

[1] M. Á. Ruiz Jaimes, J. A. Ruiz-Vanoye, J. J. Flores Sedano, and Y. Toledo-Navarro, "Design and implementation of a wireless network with mechanisms that do not violate security to meet the demand of higher education institutions," *International Journal of Combinatorial Optimization Problems and Informatics*, vol. 16, no. 2, pp. 123–129, Mar. 2025, doi: 10.61467/2007.1558.2025.v16i1.419.

[2] Jeeva. N, "Wireless Networks in Day Today Life," *International Journal of Research Publication and Reviews*, vol. 4, no. 4, pp. 1903–1906, Apr. 2023, doi: 10.55248/gengpi.2023.4.4.35848.

[3] P. Chavan, Dr. K. S. Reddy, and S. P N, "AN ANALYSIS OF WIRELESS NETWORKS," *International Journal of Innovative Research in Advanced Engineering*, vol. 9, no. 8, pp. 288–299, Aug. 2022, doi: 10.26562/ijirae.2022.v0908.25.

[4] Muhammad Donni Lesmana Siahaan, "Implementation Of Wireless Controller Using Capsman (Controller Access Point System Manager) In Computer Laboratory Of SMK Negeri 9 Medan," *International Journal Of Computer Sciences and Mathematics Engineering*, vol. 2, no. 2, pp. 289–298, Nov. 2023, doi: 10.61306/ijecom.v2i2.55.

[5] Muhammad Donni Lesmana Siahaan, "Implementation Of Wireless Controller Using Capsman (Controller Access Point System Manager) In Computer Laboratory Of SMK Negeri 9 Medan," *International Journal Of Computer Sciences and Mathematics Engineering*, vol. 2, no. 2, pp. 289–298, Nov. 2023, doi: 10.61306/ijecom.v2i2.55.

[6] Muhammad Donni Lesmana Siahaan, "Implementation Of Wireless Controller Using Capsman (Controller Access Point System Manager) In Computer Laboratory Of SMK Negeri 9 Medan," *International Journal Of Computer Sciences and Mathematics Engineering*, vol. 2, no. 2, pp. 289–298, Nov. 2023, doi: 10.61306/ijecom.v2i2.55.

[7] Z. Subecz, "Web-development with Laravel framework," *Gradus*, vol. 8, no. 1, pp. 211–218, 2021, doi: 10.47833/2021.1.CSC.006.

[8] L. A. T. Nguyen, T. S. Huynh, D. T. Tran, and Q. H. Vu, "Design and Implementation of Web Application Based on MVC Laravel Architecture," *European Journal of Electrical Engineering and Computer Science*, vol. 6, no. 4, pp. 23–29, Aug. 2022, doi: 10.24018/ejece.2022.6.4.448.

[9] Hoiriyah, H. Riadi, and Bakir, "Sistem Informasi Manajemen Produksi Dan Persediaan Bahan Baku Air Minum Dalam Kemasan (AMDK) Berbasis Web," *NJCA (Nusantara Journal of Computers and Its Applications)*, vol. 7, no. 1, pp. 28–38, Jun. 2022, doi: http://dx.doi.org/10.36564/njca.v7i1.284.g100.

[10] R. Suratnu, "THE ADOPTION OF THE ADDIE MODEL IN DESIGNING AN INSTRUCTIONAL MODULE: THE CASE OF MALAY LANGUAGE REMOVE STUDENTS," *IJIET (International Journal of Indonesian Education and Teaching)*, vol. 7, no. 2, pp. 262–270, Jul. 2023, doi: 10.24071/ijiet.v7i2.3521.

[11] M. A. Roziqin and U. Indahyanti, "Web-Based Offset Printing System Development Using ADDIE Method," *JICTE (Journal of Information and Computer Technology Education)*, vol. 8, no. 1, pp. 16–23, Apr. 2024, doi: 10.21070/jicte.v8i1.1658.

[12] A. Bakhrun, "Perancangan Sistem Pembelajaran Daring Menggunakan Model ADDIE," *Journal of Education and Instruction (JOEAI)*, vol. 4, no. 2, pp. 633–650, Dec. 2021, doi: 10.31539/joeai.v4i2.2887.

[13] H. Crompton *et al.*, "Examining technology use within the ADDIE framework to develop professional training," *European Journal of Training and Development*, vol. 48, no. 3/4, pp. 422–454, Mar. 2024, doi: 10.1108/EJTD-12-2022-0137.

[14] E. C. Asilo, J. S. Laranas, and F. L. C. Decena, "Designing Technology-Based e-Learning for Adult Education in the Philippine Agriculture Sector: The PCAARRD Advanced Learning Management System Experience," *SN Comput. Sci.*, vol. 5, no. 5, p. 620, Jun. 2024, doi: 10.1007/s42979-024-02966-3.

[15] M. Faid, A. Supriadi, M. Sukron, and M. Furqan, "Aplikasi Raport Digital Dengan Framework Codeigniter 4 Berbasis Kurikulum Merdeka Di SMKN 4 Probolinggo," *NJCA (Nusantara Journal of Computers and Its Applications)*, vol. 9, no. 1, pp. 1–7, Jun. 2024, doi: http://dx.doi.org/10.36564/njca.v9i1.289.g122.