

Development of a Web-Based Server Monitoring System Using the PPDIOO Method

Riat Rayendra^{1*}, Aditya Kusuma Setyanegara², Muhammad Raafi Febrian Tara³

¹State Islamic University of Sultan Syarif Kasim Riau, Pekanbaru, Indonesia

²IPB University, Bogor, Indonesia

³State Islamic University of Raden Mas Said Surakarta, Sukoharjo, Indonesia

Email: ¹⁾ riatrayendra1996@gmail.com, ²⁾ adityakusuma365@gmail.com,

³⁾ raafi.febriantara@gmail.com

Received : 06 March - 2025

Accepted : 10 April - 2025

Published online : 14 April - 2025

Abstract

The Regional Revenue Agency (Bapenda) of Riau Province is a government institution responsible for planning, implementing, monitoring, and managing regional revenue in Riau Province. Since 2011, Bapenda has implemented a centralized local revenue tax collection system through the Registration Identification and Samsat System (SIRIS). In 2017, the SIRIS system handled an average of 3,606 transactions from 33 UPTUP across Riau Province, generating a total revenue of Rp 1,703,507,593,488. To maintain the quality of payment services, the SIRIS server must be continuously monitored to ensure optimal performance. One of the main issues in the current monitoring process is that Network Administrators must manually monitor the system at all times since it lacks an automatic alert feature for disruptions. This condition requires administrators to continuously monitor the server, which is not always feasible. This research focuses on developing a Web-Based Server Monitoring System that can provide real-time notifications via Instant Messaging Telegram. The system is developed using the Prepare, Plan, Design, Implement, Operate, Optimize (PPDIOO) methodology. User Acceptance Testing results indicate a 100% user acceptance rate for the system. Based on the testing results, it can be concluded that the developed server monitoring system effectively assists in server monitoring and promptly provides notifications via Telegram in case of disruptions. With a high acceptance rate, this system significantly facilitates Network Administrators in overseeing the server more efficiently.

Keywords: PPDIOO, Server Monitoring System, System Development, User Acceptance Test.

1. Introduction

The Regional Revenue Agency (Bapenda) of Riau Province is a government institution responsible for planning, implementing, monitoring, and managing regional revenue. According to Riau Governor Regulation No. 93 of 2016, Bapenda's primary functions include formulating technical policies, providing administrative support, evaluating, reporting, and supervising regional revenue activities. Riau Province's revenue sources consist of Local Own-Source Revenue (PAD), Revenue Sharing Funds, and other legitimate regional income. In 2017, Bapenda recorded a regional income of Rp 7,659,807,348,337, which has continued to increase annually (Bapenda, 2017).

PAD revenue sources include Motor Vehicle Tax (PKB), Motor Vehicle Title Transfer Fee (BBN-KB), Motor Vehicle Fuel Tax (PBB-KB), Surface Water Tax (PAP), and Cigarette Tax. Among these, PKB and BBN-KB have the highest transaction frequencies. In 2017 alone,



Bapenda recorded Rp 1,703,507,539,488 in revenue from these taxes, with an average of 3,606 transactions per day across Riau Province (Bapenda, 2017). To facilitate tax payments, Bapenda established 17 Technical Implementation Units (UPT) and 16 Implementation Units (UP) across the province, enabling taxpayers to make payments at their nearest facility without visiting Bapenda's main office.

To streamline reporting, services, and oversight—especially for PKB and BBN-KB tax collection—Bapenda implemented the Registration Identification and Samsat System (SIRIS) in 2011. This centralized system requires all UPTs and UPs to connect to Bapenda's main server, which must remain operational at all times to prevent disruptions in tax collection (Irawan, 2005). Any downtime on the main server directly impacts PAD revenue processing across the province.

Currently, server monitoring at Bapenda requires Network Administrators to manually log into the system and monitor operations continuously. However, this approach is inefficient, as the system does not provide automatic alerts for server issues. Additionally, Bapenda only employs two Network Administrators working from 08:00 AM to 04:00 PM on weekdays, meaning server issues outside working hours go undetected until the next business day. Given their other responsibilities, administrators often experience delays in identifying and resolving server downtime, which can lead to significant financial losses (Ahmad Ramdhani, personal interview, 2023).

Based on 2017 real-time revenue data, every hour of server downtime results in a revenue loss of Rp 316,580,053 and a backlog of approximately 452 taxpayers unable to process payments (Bapenda, 2017). If not addressed, this issue could lead to taxpayer dissatisfaction, decreased regional revenue, and negative public perception of Bapenda's service quality. Furthermore, it may prevent the achievement of revenue targets set by the Riau Governor.

Despite the crucial role of server reliability in tax revenue collection, the current monitoring system lacks real-time alert capabilities, leading to delayed issue detection and response. This research addresses this gap by developing a Web-Based Server Monitoring System that provides real-time notifications to administrators, ensuring system reliability and minimizing revenue loss. According to Ferdianto et al. (2018), a network monitoring system is designed to track network activity and immediately detect issues in connected devices. To improve response time, the monitoring system must send alerts through various communication channels, such as Short Message Service (SMS), email, or Instant Messaging.

This research aims to develop a Web-Based Server Monitoring System for Bapenda Riau, providing automated alerts and improving administrative response times. The final outcome is expected to enhance the efficiency of server monitoring, improve service quality for taxpayers, minimize revenue losses, and ultimately increase Riau Province's regional revenue.

2. Literature Review

2.1. The Regional Revenue Agency (Bapenda) of Riau Province

The Regional Revenue Agency (Bapenda) of Riau Province is a government institution responsible for managing regional revenue to support governance and development financing. Its establishment dates back to January 13, 1974, under Riau Governor Decree No. KPTS 29/1/1974, initially as the Provincial Tax and Revenue Agency. Over time, its structure evolved to enhance revenue collection efficiency, aligning with Article 49(2) of Law No. 5 of 1974 concerning Regional Government Principles. The agency's organizational framework was formalized through Riau Regional Regulation No. 5 of 1979, approved by the Minister of Home

Affairs under Decree No. PEM.061.341.24/127 (Ministry of Home Affairs, 1980). Bapenda operates under a strategic vision: “Realizing local revenue as the primary support for the smooth operation of Riau Province's governance in a professional manner.” To achieve this vision, three key missions are outlined: (1) optimizing the intensification and extensification of local revenue, (2) improving public services professionally, and (3) ensuring equitable revenue-sharing aligned with regional potential (Bapenda, 2020). The agency's primary functions include formulating regional revenue policies, coordinating interdepartmental activities, preparing budget plans, and executing revenue collection strategies. It also manages the allocation of local revenue to ensure balanced development across Riau's districts, sets revenue targets, and conducts technical training for revenue officers (Riau Governor Decree No. KPTS.286/XI/1980). Additionally, Bapenda plays a crucial role in maintaining financial stability through tax collection initiatives such as Motor Vehicle Tax (PKB) and Motor Vehicle Title Transfer Fees (BBN-KB), which significantly contribute to the province's annual revenue (Bapenda, 2017). Given its essential role in revenue generation, Bapenda continuously enhances its operational framework to improve service quality and efficiency. The agency's effectiveness directly impacts the realization of regional development goals and public welfare in Riau Province (Ferdianto et al., 2018).

2.2. System Definition and Development

A system is a collection of interrelated components that work together to achieve a specific goal by receiving input and producing output through a structured process (Mulyanto, 2009). According to Jogiyanto (1999), a system has several characteristics, including consisting of multiple interacting components that function as subsystems, having boundaries that separate it from the external environment, and possessing connectors that enable resource flow between subsystems. Additionally, a system includes inputs as processing energy, outputs as results, a processing mechanism to transform inputs into outputs, and a specific objective to be achieved. System development is the process of designing or improving an existing system to enhance its efficiency and effectiveness by integrating procedures, human resources, and available technology (Jogiyanto, 1999). According to Kadir (2003), system development follows several stages, beginning with planning, where a system must be well-designed before implementation. The next phase is system analysis, which involves identifying problems, understanding the existing system, analyzing requirements, and preparing reports (Ulfasari & Husin, 2022). Once the analysis is complete, system design is conducted to provide users with a comprehensive overview of the new system. The final stage is implementation, which involves deploying the system for operational use to meet organizational needs. A systematic approach to system development ensures that the resulting system effectively improves performance and supports the achievement of organizational goals (Kadir, 2014).

2.3. Monitoring

Monitoring is defined as a cycle of activities that includes data collection, review, reporting, and action on information related to an ongoing process (Mercy, 2010). Generally, monitoring is used to check the performance of a system against predetermined targets. Network monitoring refers to a system that tracks network activity, allowing administrators to quickly identify and resolve any issues (Ferdianto, 2018). Similarly, server monitoring involves the continuous supervision of a server to detect and report potential issues to the administrator (Johnson, 2011). An effective monitoring system must be simple, user-friendly, and designed with clarity and precision. It should focus on key indicators to ensure efficiency, be well-planned from a technical standpoint, and have structured procedures for data

collection and analysis to avoid inaccurate inputs or outputs (Mercy, 2010). The primary goals of a monitoring system include ensuring processes adhere to established procedures, improving data accuracy, identifying errors early, and fostering a culture of accountability and motivation among employees (Amsler, 2009). In the context of server monitoring, the goal is to gather essential information for server management and control (Kusaeri, 2010). According to Lestiawan (2011), monitoring helps oversee multiple machines, quickly detect issues before they escalate, maintain server stability, diagnose problems, and promptly notify administrators of failures. A well-implemented monitoring system enhances operational efficiency and prevents disruptions.

2.4. Computer Networks

A computer network is the integration of computer technology and telecommunications, enabling distributed data processing, shared use of databases, software applications, and hardware. Previously isolated computer systems can now be interconnected, allowing seamless communication and resource sharing (Sopandi, 2018). Based on geographical coverage, computer networks are categorized into three types. Local Area Network (LAN) covers a limited area, such as a building or a campus, facilitating efficient data transfer within a confined space. Metropolitan Area Network (MAN) spans across cities and is commonly used by Internet Service Providers (ISPs) to deliver network services. Wide Area Network (WAN) has the broadest coverage, often utilizing satellites and fiber optic cables to connect networks globally. To establish a functional network, specific hardware components are required. A Network Interface Card (NIC) connects multiple computers by converting parallel data streams into serial form for network transmission. A switch facilitates communication by directing data to the intended recipient. Network cables, such as UTP, STP, coaxial, and fiber optic, serve as transmission media for data exchange. An access point bridges wired and wireless networks, ensuring seamless connectivity between different network types. Finally, a router connects multiple networks, enabling data transmission across different network environments (Sopandi, 2018).

2.5. Internet

The term “Internet” originates from “Interconnection Networking.” According to Oetomo et al. (2007), the Internet is a limitless communication network that connects millions of personal computers worldwide. The integration and communication within this network are governed by the TCP/IP protocol. Transmission Control Protocol (TCP) ensures that all connections function correctly, while Internet Protocol (IP) handles the transmission of data between computers. TCP/IP is responsible for selecting the best data transmission route, finding alternative paths if a route is unavailable, and managing data packet delivery. The Internet facilitates virtual conferences, allowing multiple participants to engage in discussions without physical meetings, using only their computer screens. Additionally, the Internet supports the development of e-business systems, enabling transaction data integration across terminals worldwide. Several commonly used terms in Internet technology include the World Wide Web (WWW), which is a vast collection of web servers providing shared information on various topics such as politics, economics, social issues, technology, and education. A website is a storage location for data and information categorized by specific topics, while web pages refer to individual sections of a website. The homepage serves as the front page containing menus or an index of a website. Lastly, a browser is an application that allows users to navigate and access various information on the WWW (Oetomo et al., 2007).

2.6. Web

The World Wide Web (WWW), commonly known as the web, is one of the services accessible to users connected to the Internet. It is a system where information in the form of text, images, audio, and other media is stored on web servers and displayed as hypertext documents. These documents are written in HTML format (Sutanta, 2005). Initially, the web was an information space on the Internet that utilized hypertext technology, allowing users to navigate through information by following links embedded within web documents displayed in web browsers (Sidik & Pohan, 2010). The web simplifies interaction between users and facilitates information browsing on the Internet. Additionally, the adoption of web technology by businesses is driven by several factors. According to Sidik & Pohan (2010), the reasons for adopting web technology include easy access to information, simple server setup, effortless information distribution, and platform independence, as web content can be accessed by any operating system through a web browser.

2.7. Hypertext Markup Language (HTML)

According to Sidik & Pohan (2010), HTML is a plain text file that can be created using any text editor and is commonly known as a web page. It originated from a language widely used in publishing and printing, known as SGML (Standard Generalized Markup Language). HTML has become the universal standard for displaying web pages and is currently defined and regulated by the World Wide Web Consortium (W3C). W3C, established in 1994 by Tim Berners-Lee, is an international organization composed of companies involved in the Internet and the World Wide Web. Its primary goal is to regulate and establish web standards for universal accessibility and consistent formatting across different web browsers. W3C is also the main standards organization for HTTP (Hypertext Transfer Protocol) and HTML (Hypertext Markup Language). HTML enables users to control the appearance of web pages and their content, ensuring consistency in design and structure. It allows documents to be published online, making them accessible to users worldwide. Additionally, HTML facilitates the creation of online forms that can handle various tasks such as user registrations and online transactions. Moreover, HTML supports the integration of multimedia elements, including images, audio, video, and Java applets, enhancing the interactivity and visual appeal of web pages.

2.8. PHP

According to Sidik (2012), PHP (HyperText Preprocessor) is a server-side scripting language embedded within HTML and executed on the server. It is widely used to develop web-based applications and offers seamless integration with various databases, such as MySQL, PostgreSQL, and Microsoft Access. One of PHP's key advantages is its open-source nature, allowing developers to use it freely without licensing fees. By using PHP, websites can be built dynamically, making their maintenance more manageable and efficient (Peranginangin, 2006). PHP is compatible with multiple operating systems, including Linux, Unix (HP-UX, Solaris, OpenBSD), Microsoft Windows, and macOS. Additionally, PHP can function as a CGI processor, making it a versatile tool for web development. Beyond generating HTML output, PHP is also capable of handling image processing, generating PDF files, and creating Flash movies. It can also generate structured text formats such as XHTML and XML. Hidayatullah & Kawistara (2017) highlight several advantages of PHP. First, PHP operates as a server-side scripting language, meaning the execution occurs on the server rather than the client's browser. It also supports command-line scripting, allowing developers to run PHP scripts without a web browser. Furthermore, PHP enables the creation of desktop applications and is compatible with multiple operating systems. It supports various web servers and offers

flexibility in programming, allowing both object-oriented and procedural approaches. Additionally, PHP can output XHTML, HTML, and XML formats while providing extensive support for relational database management systems (RDMS). PHP also facilitates various communication protocols and offers powerful text processing capabilities, making it a robust and efficient programming language for web development.

2.9. Instant Messaging

According to Wikipedia, Instant Messaging is a fast message delivery system via the internet between computers. It originated in the early 1990s when online technology became popular. Software developers created chat rooms where individuals or groups could communicate. Quantum Link introduced a feature allowing users to send messages while online. In November 1996, Mirabilis launched ICQ, a free instant messaging service, leading to its widespread adoption. AOL became a pioneer in 1997 by integrating chat rooms and instant messaging. Today, instant messaging has evolved beyond computers to mobile devices, enabling users to stay connected anywhere. In Indonesia, instant messaging is widely used, particularly among students, with software like mig33 and eBuddy facilitating mobile communication.

2.10. Telegram

Telegram is a messaging application focusing on speed, security, simplicity, and cost-free access (telegram.org). It was launched on iOS on August 14, 2013, and on Android on October 20, 2013. Besides smartphones, Telegram is available on web and desktop (Windows, macOS, Linux). It offers several advantages over other instant messaging apps, including sending text, images, videos, and various file formats up to 1.5GB. Telegram supports group chats of up to 100,000 members and allows bot creation for automated messaging.

2.11. Simple Network Management Protocol (SNMP)

SNMP is a standard network management protocol (Purbo & Taufan, 2001). It stores captured information in the Management Information Base (MIB) and defines message exchanges between the Network Management Station (NMS) and network devices. SNMP enables large-scale network management by allowing computers within a network to report their status to NMS.

2.12. Server

According to Nugroho (2005), a server acts as a central hub that facilitates data transmission, reception, and management among connected computers, serving clients efficiently.

2.13. Database

A database is crucial for information systems as it stores and organizes data systematically. According to Oetomo et al. (2007), a database comprises interrelated data groups structured to prevent unnecessary duplication and enable fast data processing. Businesses rely on databases for efficient data organization. Oetomo et al. (2007) describes database hierarchy as follows: A database consists of related files, each file contains records with similar field formats, a record includes multiple fields representing specific data, a field stores an item such as a name or address, bytes form the characters within a field, and bits represent the smallest data unit, often in ASCII format (0s and 1s).

2.14. Database Management System (DBMS)

Rosa & Salahuddin (2011) defines DBMS as an application system for data storage, management, and retrieval. A DBMS must provide data access management, ensure data integrity, and support data backup. Due to its importance, most businesses use DBMS, managed by specialists known as Database Administrators (DBA). Common open-source DBMS include MySQL, PostgreSQL, Firebird, and SQLite.

2.15. MySQL

MySQL is a cross-platform, open-source database server (Gunawan & Setiabudi, 2004). It operates efficiently on Linux and Windows. Web developers prefer MySQL due to its ease of use and seamless integration with PHP. It is an open-source Relational Database Management System (RDBMS) distributed under the GPL license. RDBMS allows users to create, manage, and use relational data models, linking multiple tables (Prasetyo, 2004). According to Prasetyo (2004), MySQL is fast, reliable, and user-friendly, operating up to three to four times faster than commercial database servers while requiring minimal administrative expertise.

2.16. PPDIOO Network Development Methodology

The PPDIOO methodology, used for LAN and WLAN development, consists of six phases: (1) Prepare, which defines organizational needs and high-level network strategies; (2) Plan, which identifies initial requirements, analyzes existing networks, and conducts a GAP analysis; (3) Design, which details infrastructure architecture, ensuring reliability, security, scalability, and performance; (4) Implement, which tests and deploys the network without disrupting existing systems; (5) Operate, which monitors system health, minimizes downtime, and ensures operational efficiency; and (6) Optimize, which proactively resolves emerging issues and enhances system reliability through corrective actions and potential upgrades.

2.17. Black Box Testing

Black box testing evaluates a software interface by focusing on functional testing rather than implementation. This technique verifies whether software inputs, outputs, and functions meet the required specifications (Pressman, 2010). The test success rate is calculated as follows:

$$\text{Success Percentage} = \left(\frac{\text{Successful Answers}}{\text{Total Questions}} \right) \times 100\%$$

2.18. User Acceptance Test (UAT)

According to Perry (2007), UAT is conducted by end-users, often employees, to verify system functionality. Lewis (2012) describes acceptance testing as a black-box testing method that ensures software meets specifications. Black (2002) states that acceptance testing confirms system compliance with predefined requirements. In software and hardware development, it includes alpha tests (internal users) and beta tests (external users) to determine market readiness.

2.19. Related Work

Several previous studies have been conducted on network monitoring systems using various approaches and technologies. Taftazanie et al. (2017) developed a web-based server monitoring application that provides outage notifications via SMS and email. This research was further enhanced by adding instant messaging notifications through Telegram. Segara et al. (2018) built a network monitoring system using the SNMP protocol, displaying CPU, load, and memory information in graphical form on a web interface. Further improvements were

made by integrating instant messaging notifications and monitoring running services. Ferdianto et al. (2018) developed a network activity monitoring system using a Raspberry Pi microcomputer, where monitoring results were displayed as graphs on a web application. This study was later expanded by adding notification features and service monitoring. Nugroho et al. (2014) designed a network monitoring application using SNMP with an early warning system via SMS and a MySQL database. This research was further improved by implementing Telegram notifications and additional service monitoring features. Wun (2014) created a system for visualizing network traffic, managing bandwidth usage, and blocking websites via SNMP on a web-based platform. This system was further developed by integrating instant messaging notifications and monitoring uptime and hardware usage. Building upon these previous studies, this research contributes by expanding the network monitoring system through instant notification integration and a more comprehensive service monitoring approach.

3. Methods

3.1. Research Methodology Flow

To clarify the stages of the research process, the author has created a research flowchart, which can be seen in Figure 1. This flowchart outlines the systematic steps taken throughout the study, from initial analysis to implementation and evaluation.

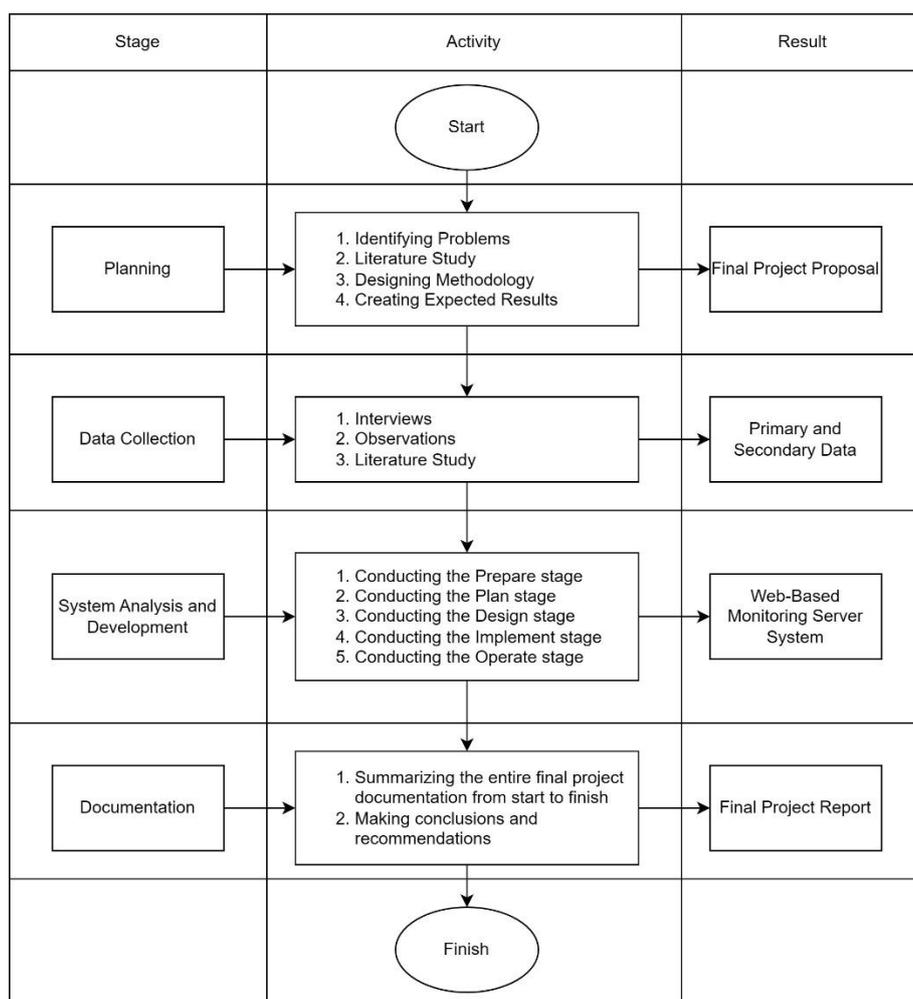


Figure 1. Research Methodology Flow

3.2. Planning and Data Collection

In this stage, the author conducted observations on the situation and conditions at the Regional Revenue Agency (Bapenda) of Riau Province, particularly regarding the existing system processes. The author visited the Bapenda office and interviewed Ahmad Ramdhani, a staff member in the Data Management and Revenue Development Division, and Ranra, a Functional Computer Operator at Bapenda Riau. The interview data, which can be found in Appendix A, helped identify issues related to the monitoring of Bapenda's main server. Based on these identified problems, the author conducted a literature review by studying relevant books and journals. This served as the foundation for planning the research, which involved defining the research background, methodology, and expected outcomes. To ensure the completeness of the research data, three data collection techniques were employed: interviews, observations, and literature review. The interviews involved a structured set of questions to assess whether the system met operational needs and to identify common challenges faced by Bapenda Riau. Observations were conducted at the Bapenda office, allowing the author to examine existing server monitoring practices and detect any operational issues, with photo documentation available in Appendix D. Additionally, a literature review was conducted to gather insights from books and journals on server monitoring, providing supporting references for this research.

3.3. Data Collection Tools and Procedure

The System Analysis and Development phase is the most critical stage of this research. The development of the server monitoring system follows the PPDIIO model, consisting of five stages: Prepare, Plan, Design, Implement, and Operate. The process flow of this system development can be seen in Figure 2.

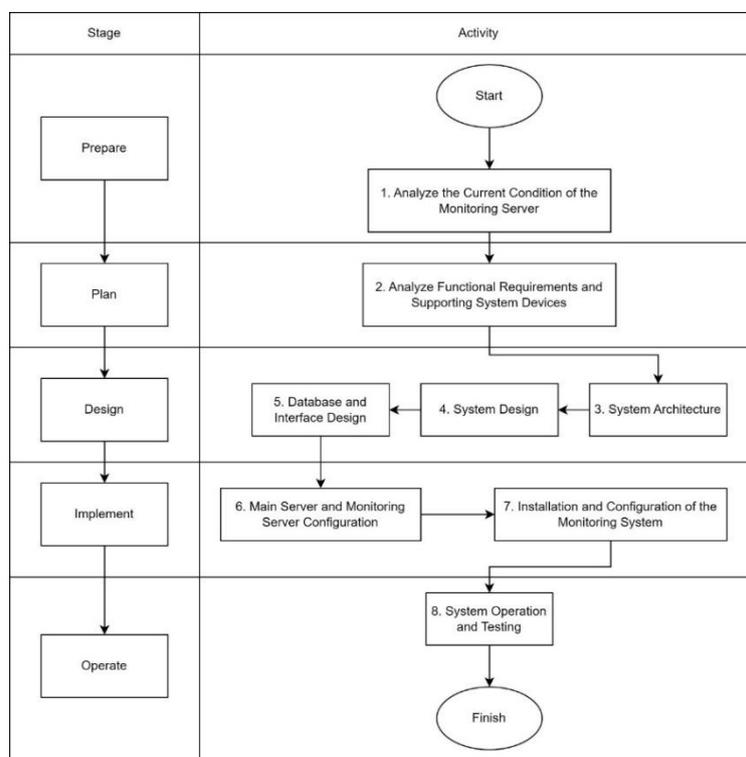


Figure 2. System Monitoring Development Flow

In the Prepare phase, an initial analysis was conducted to assess the main server's condition and the current monitoring system. This was followed by the Plan phase, where the

system requirements were identified, including functional requirements, hardware specifications, and supporting software for the web-based server monitoring system. The Design phase involved structuring the system architecture, defining monitoring workflows, creating system flow diagrams, designing a suitable database, and developing the system's interface. Next, in the Implement phase, preparations were made to configure both the main server and the monitoring server to ensure system stability. Once configured, the server monitoring system was installed. The final stage, Operate, involved testing and evaluating the functionality of the web-based server monitoring system to ensure it met the predefined requirements. Tests included SNMP testing, system notification testing, black-box testing, and User Acceptance Testing (UAT). Following the development and testing, the Documentation phase was carried out to compile all research findings into a final report. This stage summarized the results, drew conclusions, and provided recommendations, forming the basis of the final research report. The documentation process follows the guidelines and includes all research results obtained throughout the study.

4. Results and Discussion

The proposed web-based server monitoring system for Bapenda Riau aims to enhance efficiency by replacing the slow manual monitoring process. By utilizing SNMP and Telegram notifications, the system facilitates real-time and automated monitoring. It requires two servers, supporting devices, and software such as Debian 8.1, Apache, MySQL, and PHP. The SNMP architecture collects server performance data and sends notifications in case of failures. The system's database consists of seven tables to store critical information, while a user-friendly interface allows administrators to monitor server and service status in real-time, reducing downtime and improving service reliability.

4.1. Current Condition Analysis

The Regional Revenue Agency (Bapenda) of Riau Province collects regional revenue, particularly Motor Vehicle Tax (PKB) and Transfer of Motor Vehicle Title Fee (BBN-KB), through 33 UPT and UP offices spread across the province. To facilitate reporting, service management, and supervision, especially for PKB and BBN-KB revenue, Bapenda has developed the *Sistem Informasi Registrasi Identifikasi dan Samsat* (SIRIS) since 2011. This centralized system requires all UPT and UP offices to connect to the main server located at Bapenda Riau Province. Since the tax payment system is centralized, the main server must be continuously monitored to ensure optimal performance and uninterrupted service.

The main server at Bapenda Riau Province is an HP ProLiant DL380 running CentOS 7.x. Linux was chosen as the operating system due to its open-source nature, reducing operational costs. The server was upgraded in 2018 to improve service performance and operates 24/7, shutting down only during maintenance. To support its operation, the server is equipped with UPS and an air conditioning system. The server specifications include CentOS 7.x as the operating system, English as the system language, HP ProLiant DL380 Gen9 3.5GHz as the processor, 128GB DDR4 memory, and a 3TB disk. The main server also runs various software, including OpenSSH Server 6.4, Apache 2.4.3 as the web server, PHP 5.4.7 as the programming language, MySQL 5.5.27 as the database, and PHP 5.6.33.

Currently, the Network Administrator must be physically present near the main server and log into the system to run monitoring commands. If the main server crashes, administrators can only detect it when actively monitoring since there is no automatic notification system. This manual monitoring process is inefficient because Bapenda's IT team consists of only two network administrators, working Monday to Friday from 8 AM to 4 PM.

If the server fails outside working hours, the issue is discovered late, often through reports from UPT and UP offices rather than real-time monitoring. This delay slows down server recovery.

4.2. System Requirements Analysis

Before designing the monitoring system, a requirement analysis was conducted, covering functional, hardware, and software requirements. To address the existing monitoring challenges, the proposed Web-Based Server Monitoring System must collect server status data every 15 seconds, store server logs at 15-second intervals, display real-time server status through a local web interface, send alerts if the main server or its services fail, notify administrators if any UPT or UP loses connection, and send instant notifications via Telegram in case of disruptions.

The proposed system requires two computers, one as the main server and one as the monitoring server. Three laptops will be used, one for the web monitoring display and two as simulated UPT and UP clients. A Switch Hub is needed to interconnect all devices within the local network, along with UTP cables for data transmission and RJ45 connectors for connecting UTP cables to network devices.

The software components needed for the Web-Based Server Monitoring System include Debian 8.1 (Jessie) as the operating system for both monitoring and simulated main servers, SNMP 5.7.2.1 for collecting server status and performance metrics, Apache 2.4.10 as a web server to provide monitoring dashboards, MySQL 5.5.50 as a database to store monitoring logs, PHP 5.6.33 as a server-side scripting language for web development, PuTTY 0.70 as a remote access tool for server administration, Mozilla Firefox 61.0.1 as a browser for monitoring web display and system testing, Windows 8.1 64-bit as the operating system for client-side monitoring tools, and Telegram 4.8.11 for real-time notification alerts on mobile devices.

4.3. Server Monitoring System Design

The system design phase defines the architecture, workflow, process diagrams, database structure, and user interface of the monitoring system. Based on the analysis, the proposed system architecture consists of two key servers: the Main Server as the monitored device and the Monitoring Server as the observer. Both servers use SNMP, with the Main Server acting as the SNMP Agent and the Monitoring Server as the SNMP Manager. The monitoring server not only collects server performance data but also sends alerts via Telegram if failures occur. Figure 3 presents the overall system architecture, showing the relationship between the Main Server, Monitoring Server, and client devices.

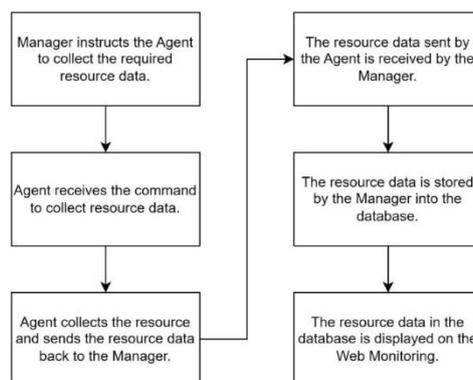


Figure 3. System Workflow Process

The system operates by having the Monitoring Server send SNMP requests to the Main Server for status updates. The Main Server responds with CPU, RAM, disk usage, and service status data. The Monitoring Server saves the data into a database and displays it on a web-based dashboard. If the Main Server crashes, the Monitoring Server detects the failure and sends a Telegram alert. The monitoring process flow, including how data is collected, processed, and stored, is detailed in Figure 4.

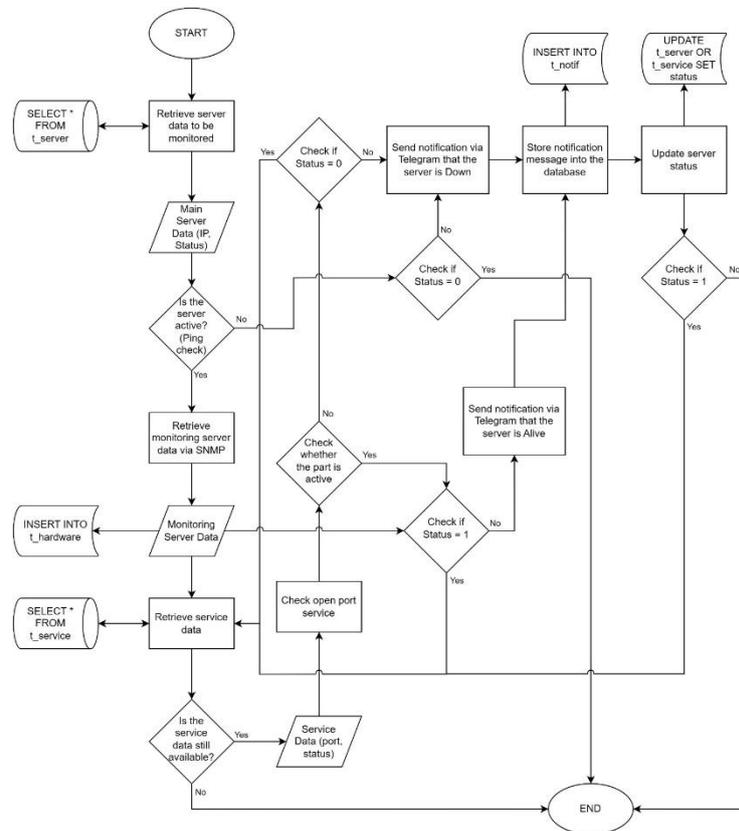


Figure 4. Main Server Monitoring Flowchart

The system flowchart describes the monitoring process, including Main Server Monitoring and UPT & UP Connection Monitoring. For Main Server Monitoring, the system checks if the server is active using ping commands. If no response is received, the system updates the server status to “offline” and sends a Telegram alert. If the server is active, it retrieves CPU, RAM, disk, and service status. If any service fails, a notification is sent, and the status is updated. For UPT & UP Monitoring, the system checks UPT & UP connectivity via ping requests. If a UPT/UP fails to respond, an alert is sent via Telegram. These processes ensure real-time monitoring and immediate failure detection. Figure 4.5 displays the user interface of the web-based monitoring dashboard, highlighting its key features and functionalities.

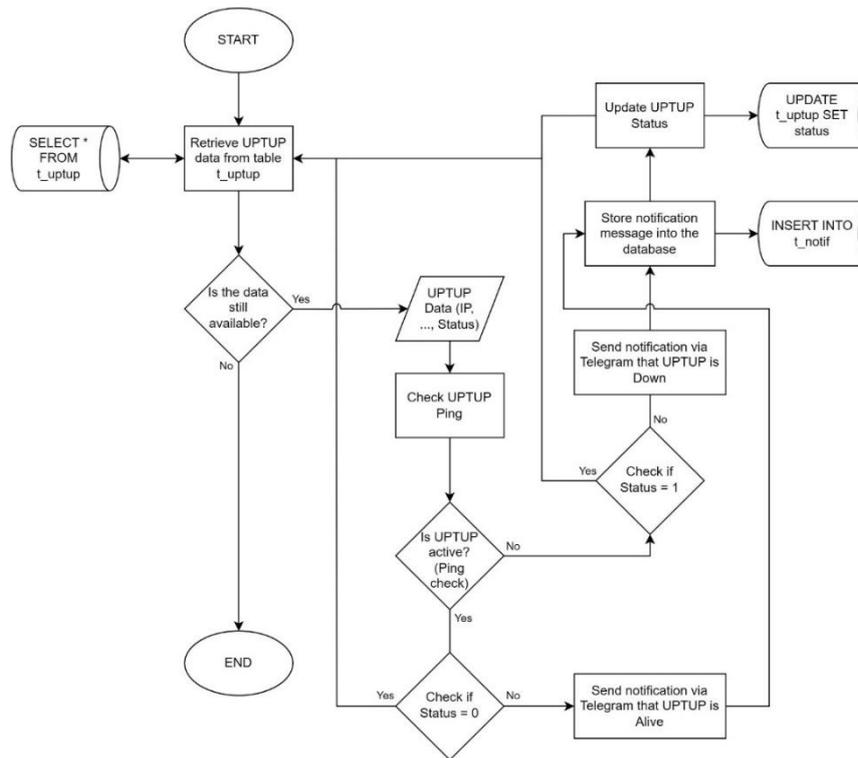


Figure 5. UPT and UP Monitoring Flowchart

4.4. Database Design

Database design is a crucial stage in the development of the server monitoring system. This database functions to store all the data needed in the system, such as server information, hardware, services, uptime, notifications, and users. The database used in this system is db_simover, which consists of seven main tables. The database design can be seen in Tables 1 to 8 below:

Table 1. Database Tables Overview

No	Table Name	Primary Key	Description
1	t_server	ip_server	Stores monitored server data.
2	t_hardware	no	Stores hardware monitoring information (CPU, RAM, disk, etc.).
3	t_service	id_service	Stores data on services running on the server.
4	t_uptup	id_uptup	Stores UPTUP monitoring data.
5	t_notif	comment_id	Stores system notification log messages.
6	t_telegram	tele_id	Stores Telegram administrator data for system notifications.
7	t_user	id_user	Stores application user login information.

The details of each table in the db_simover database are as follows:

Table 2. Table t_server

No	Field Name	Data Type	Length	Default
1	id_server	Varchar	15	NOT NULL
2	hostname	Varchar	40	NOT NULL
3	community	Varchar	40	NOT NULL
4	n_status	Int	1	NOT NULL

Table 3. Table t hardware

No	Field Name	Data Type	Length	Default
1	no	Int	100	NOT NULL
2	ip_server	Varchar	15	NOT NULL
3	cpu_perc	Float	-	NOT NULL
4	ram_tot	Int	10	NOT NULL
5	ram_aval	Int	10	NOT NULL
6	ram_used	Int	15	NOT NULL
7	ram_perc	Float	-	NOT NULL
8	disk_tot	Int	10	NOT NULL
9	disk_aval	Int	10	NOT NULL
10	disk_used	Int	11	NOT NULL
11	disk_perc	Float	-	NOT NULL
12	uptime	Varchar	40	NOT NULL
13	date	Datetime	-	NOT NULL

Table 4. t_service

No	Field Name	Data Type	Length	Default
1	id_service	Int	5	NOT NULL
2	ip_server	Varchar	15	NOT NULL
3	service_name	Varchar	20	NOT NULL
4	port	Int	5	NOT NULL
5	status	Int	1	NOT NULL

Table 5. t_uptup

No	Field Name	Data Type	Length	Default
1	id_uptup	Int	5	NOT NULL
2	ip_uptup	Varchar	15	NOT NULL
3	h_uptup	Varchar	40	NOT NULL
4	status	Int	1	NOT NULL

Table 6. t_notif

No	Field Name	Data Type	Length	Default
1	comment_id	Int	11	NOT NULL
2	comment_subject	Varchar	250	NOT NULL
3	comment_text	Text	-	NOT NULL
4	warning	Int	1	NOT NULL
5	comment_status	Int	1	NOT NULL
6	date	Datetime	-	NOT NULL

Table 7. t_telegram

No	Field Name	Data Type	Length	Default
1	tele_id	Int	10	NOT NULL
2	tele_name	Varchar	20	NOT NULL

Table 8. t_user

No	Field Name	Data Type	Length	Default
1	id_user	Int	5	NOT NULL
2	username	Varchar	20	NOT NULL
3	password	Varchar	40	NOT NULL
4	w_login	Datetime	-	NOT NULL

This database design is structured to store essential information in the server monitoring system. Each table has a specific function to ensure the system runs optimally and

efficiently. With this database structure, the monitoring system can store and manage server information effectively and provide accurate notifications to administrators. If any modifications or future developments are needed, this database schema can be updated accordingly.

4.5. Interface Design

The interface design serves as a guide in developing the monitoring system, ensuring users understand the structure and functionality before implementation. The login page contains two text boxes for username and password, along with a login button to access the system. The home page displays server information, service status, and graphs of CPU, RAM, and Disk usage. The Server Description section provides details of the monitored server, including hostname, IP Address, uptime, and status. The Services Table lists monitored services, including name, port, status, and edit/delete buttons. Hardware Percentage displays hardware usage data in graphical form. Notification Bell provides alerts for new messages. The UPT and UP page is used for monitoring connected units, while the Notification page logs incoming messages with options to print reports and configure Telegram message recipients. Lastly, the Settings page allows administrators to change their passwords. The interface design details are outlined in the tables below.

Table 9. Login Interface Design

No	Name	Description
1	URL	Address to access the monitoring system
2	Background Color	Change system background color
3	System Name	System title
4	Username	Input field for username
5	Password	Input field for password
6	Login Button	Button to access the system

Table 10. Home Interface Design

No	Name	Description
1	System Name	Monitoring system title
2	Dashboard Menu	Main menu
3	UPT/UP Menu	Monitoring UPT/UP page
4	Notifications	Notification messages page
5	Logout Menu	Logout from the system
6	Notification Bell	Indicator for new notifications
7	User Settings	Admin settings
8	Server Description	Server information
9	Service Monitoring	Service status
10	CPU Percentage	CPU usage
11	RAM Percentage	RAM usage
12	Disk Percentage	Disk usage

Table 11. Server Description

No	Name	Description
1	Title	Section title
2	Edit Button	Edit monitored server
3	Hostname	Server name
4	IP Address	Server IP Address
5	Uptime	Duration of uptime
6	Media	Total RAM and Disk
7	Rocomunity	Rocomunity SNMP
8	Status	Server status ON/OFF

Table 12. Services

No	Name	Description
1	Title	Section title
2	Add Button	Add new service
3	Service Name	Monitored service name
4	Port	Monitored service port
5	Status	ON/OFF status
6	Edit	Edit button
7	Delete	Delete button

Table 13. Hardware Percentage

No	Name	Description
1	Percentage	Hardware usage percentage
2	Time	Data storage timestamp
3	Chart	Hardware usage graph

Table 14. Notification Bell

No	Name	Description
1	Notification Logo	Indicator for new messages
2	Notification Content	Message content

Table 15. UPT and UP Interface Design

No	Name	Description
1	Title	Table title
2	UPT and UP Name	Monitored unit name
3	IP Address	Unit IP Address
4	Status	ON/OFF status
5	Actions	Edit/Delete options
6	Add Button	Add new unit

Table 16. Notification Interface Design

No	Name	Description
1	Title	Table title
2	Notification Messages	Log of received messages
3	Print Button	Print reports
4	Settings Button	Configure Telegram recipient

Table 17. Settings Interface Design

No	Name	Description
1	Title	Form title
2	Username	Admin username
3	Old Password	Input for old password
4	New Password	Input for new password
5	Confirm New Password	Confirm new password
6	Change Password	Button to update password
7	Reset	Clear input fields

The research findings indicate that the proposed web-based server monitoring system effectively enhances the efficiency and effectiveness of server performance monitoring at Bapenda Riau, utilizing SNMP integration and Telegram notifications for real-time, automated monitoring. This system supports the application of SNMP for real-time monitoring and automation, which improves server reliability and reduces downtime, aligning with previous studies on network management. The two-server architecture and database structure with seven key tables contribute to system efficiency. However, the study is limited

by its reliance on specific software and hardware configurations and the stability of the network infrastructure. The implications of this research suggest that the development of such monitoring systems could accelerate advancements in network management within the public sector and provide valuable insights for future research on the optimization of server management systems.

5. Conclusion

Based on the research conducted at the Badan Pendapatan Daerah Provinsi Riau, it can be concluded that the developed Web-Based Server Monitoring System is effective in facilitating server monitoring processes. The User Acceptance Testing (UAT) results show that the monitoring system provides quick notifications of any disruptions via instant messaging through Telegram to the Network Administrator. The system is functioning well, reducing the workload of the network administrator, and is deemed suitable for implementation at Badan Pendapatan Daerah Provinsi Riau.

Due to time constraints, the development of the Web-Based Server Monitoring System using the PPDIIO method is limited to the “operate” phase and has not yet reached the “optimize” phase. It is hoped that future researchers will continue to develop the system to the optimization phase, introducing system updates such as adding server locations, UPTUP locations, and incorporating a module for detecting server attacks, which were not covered in this study.

6. References

- Black, R. (2002). *Managing the testing process*. John Wiley & Sons.
- Ferdianto, F. F., Yahya, W., & Dewi, R. K. (2018). Pengembangan Sistem Monitoring Aktivitas Jaringan pada Mikrokomputer Raspberry Pi. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 2(2), 768–775.
- Gunawan, I., & Setiabudi, D. H. (2004). *Cara Mudah Mempelajari PHP, Apache, dan MySQL*. Graha Ilmu.
- Hidayatullah, P., & Kawistara, J. K. (2017). *Pemrograman Web*. Universitas Komputer Indonesia.
- Irawan, B. (2005). *Jaringan komputer*. Graha Ilmu.
- Jogiyanto, H. M. (1999). *Pengenalan komputer: Dasar ilmu komputer, pemrograman sistem informasi dan intelegensia buatan*. Andi Offset.
- Johnson, M. (2011). *Network Performance Management: What You Need to Know for IT Operations Management*. Emereo Pty Limited.
- Kadir, A. (2014). *Pengenalan Sistem Informasi*. Andi Offset.
- Kusaeri, A. (2010). *Monitoring Jaringan*. In *Yogyakarta: Andi*. Andi Offset.
- Lewis, J. R. (2012). Usability testing. *Handbook of Human Factors and Ergonomics*, 1267–1312.
- Mulyanto, A. (2009). *Sistem Informasi konsep dan aplikasi*. Pustaka Pelajar.
- Nugroho, A. (2005). *Rational Rose untuk Pemodelan Berorientasi Objek*. Informatika Bandung.
- Nugroho, M., Affandi, A., & Rahardjo, D. S. (2014). Rancang Bangun Aplikasi Monitoring Jaringan Menggunakan SNMP (Simple Network Management Protocol) dengan Sistem Peringatan Dini dan Mapping Jaringan. *Jurnal Teknik Pomits*, 3(1).
- Oetomo, B. S. D., Wibowo, E., Hartono, E., & Prakoso, S. (2007). *Pengantar Teknologi Informasi Internet: Konsep dan Aplikasi*. Andi Offset.

- Peranginangin, K. (2006). *Aplikasi WEB dengan PHP dan MySQL*. Andi Offset.
- Perry, W. E. (2007). *Effective Methods for Software Testing, CafeScribe: Includes Complete Guidelines, Checklists, and Templates*. John Wiley & Sons.
- Prasetyo, D. D. (2004). *Aplikasi Database Client/Server Menggunakan Delphi dan MySQL*. PT. Elex Media Komputindo.
- Purbo, O. W., & Taufan, R. (2001). *Manajemen Jaringan TCP/ IP*. PT. Elex Media Komputindo.
- Rosa, A. S., & Salahuddin, M. (2011). *Modul pembelajaran rekayasa perangkat lunak (terstruktur dan berorientasi objek)*. Modula.
- Segara, A. P., Primananda, R., & Akbar, S. R. (2018). Implementasi MQTT (Message Queuing Telemetry Transport) pada Sistem Monitoring Jaringan berbasis SNMP (Simple Network Management Protocol). *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 2(2), 695–702.
- Sidik, B. (2012). *Pemrograman web dengan PHP*. Informatika Bandung.
- Sidik, B., & Pohan, H. I. (2010). *Pemrograman Web dengan HTML Edisi Revisi Kedua*. Informatika Bandung.
- Sopandi, D. (2018). *Instalasi dan konfigurasi jaringan komputer*. Informatika.
- Sutanta, E. (2005). *Pengantar teknologi informasi*. Graha Ilmu.
- Taftazanie, S., Prasetijo, A. B., & Widiyanto, E. D. (2017). Aplikasi Pemantau Perangkat Jaringan Berbasis Web Menggunakan Protokol SNMP dan Notifikasi SMS. *Jurnal Teknologi Dan Sistem Komputer*, 5(2), 62–68.
- Ulfasari, & Husin, A. I. (2022). Decision Support System for Selecting the Best Teacher Based on Performance Appraisal at Smpit as Su'adaa Using the Simple Additive Weighting (SAW) Method. *INTERNATIONAL JOURNAL ON ADVANCED TECHNOLOGY, ENGINEERING, AND INFORMATION SYSTEM*, 1(2), 57–74. <https://doi.org/10.55047/ijateis.v1i2.372>
- Wun, Y. A. O. (2014). *TA: Monitoring Trafik Jaringan dan Pengaturan PC Router Berbasis Web (Studi Kasus: Labkom Stikom Surabaya)*. STIKOM Surabaya.