

Server Network Monitoring Notifications and Cluster Analysis through Web Application

Eidelbert Suherianto Sinaga^{1,*}, M Zaky Erdiansyah¹, Aziz Mustika Aji¹ & Atthar Luqman Ivansyah^{1,2,3}

¹Master Program in Computational Science, Faculty of Mathematics and Natural Sciences, Institut Teknologi Bandung, Jalan Ganesha No. 10, Bandung, 40132, Indonesia

²Inorganic and Physical Chemistry Research Group, Department of Chemistry, Faculty of Mathematics and Natural Sciences, Institut Teknologi Bandung, Jalan Ganesha No. 10, Bandung, 40132, Indonesia

³Instrumentation and Computational Physics Research Group, Department of Physics, Faculty of Mathematics and Natural Sciences, Institut Teknologi Bandung, Jalan Ganesha No. 10, Bandung, West Java 40132, Indonesia

*Email: eidelbertsinaga@gmail.com

Abstract. A web-based local server monitoring application using Zabbix monitoring tools was developed to fulfill the requirement of monitoring the status of network downtime and network uptime. One of the main requirements is the ability to get real-time notifications if the server has a network problem so that mitigation actions can be taken quickly. In addition, this application is also completed with the feature of sending server downtime and uptime network status to mobile applications (Telegram) and website applications built using the PHP programming language and MySQL database. By sending the status of server network problems to the website that was built, it is useful to illustrate trends and analyze patterns of server network problems, as well as analyze cluster servers based on total network disruption. With this recapitulation, network administrators can make better evaluations and planning to improve server reliability and reduce potential downtime in the future. The Zabbix monitoring tool was chosen because of its comprehensive and flexible ability to monitor various server parameters and the ease of managing and displaying data through an intuitive web interface. The results of this implementation are expected to improve operational efficiency and maintain the continuity of services offered by the server.

Keywords: *server network monitoring, notification, cluster, zabbix, web application, telegram.*

1 Introduction

In the digital era, monitoring the availability and performance of servers is a crucial aspect of IT infrastructure management. An undetected downtime can cause significant losses to the organization, both in terms of finance and reputation. Monitoring server availability and performance includes network

monitoring. Network monitoring is a procedure used to monitor computer network systems and notify network administrators in the event of a disruption [1]. Network monitoring is very important to keep the network functioning properly and know the current condition of the network [2]. Therefore, the development of an effective monitoring system is essential to ensure smooth operations and reduce the risk of disruptions. One of the widely used tools in server monitoring is Zabbix, which is known for its comprehensive and flexible monitoring capabilities. Based on a survey, the most popular open-source monitoring solutions are Nagios, Zabbix, Pandora FMS, Cenos Core, and Cacti [3]. Zabbix allows real-time monitoring of various server parameters and provides immediate notification when a fault occurs. One of the monitoring is network monitoring which can be done in real-time [4].

Zabbix is a distributed monitoring server based on a Web interface and is a free and open-source enterprise-level monitoring system solution [5]. It collects data from various sources and presents it in the form of easy-to-understand graphics and reports. One of Zabbix's outstanding features is its ability to communicate with web applications via webhooks. A webhook is a mechanism that allows Zabbix to send data to other web applications when an event occurs, such as a network outage. By using webhooks, notifications from Zabbix can be sent automatically to web applications or mobile applications to provide real-time information to administrators. With this notification, administrators or network operators do not need to open the network monitoring application every time, because if there is a disruption in the network, Zabbix will automatically send a notification to Telegram so that technicians can find out without opening the network monitoring application [2].

The choice of application platform for the purpose of sending the server network fault status from Zabbix is first to use a web application. Desktop applications are now converted or transformed into web applications for ease of use, and its core features delivered through a browser make it an efficient choice for simplifying functionality to a wide range of devices [6]. The web application developed using PHP and MySQL serves as a monitoring application that displays the *downtime* and *uptime* status of the server and network. The application also stores historical data about network disruptions, which can be used for further analysis. With an intuitive interface, network administrators can easily access and analyze monitoring data to detect disruption patterns and trends that may emerge. This historical data is critical in helping administrators perform strategic evaluation and planning to improve network reliability and performance.

Currently, many social media offer APIs so that customized applications can be developed, Telegram is one of them through Telegram-API [7]. The API used in this research is the Telegram API bot instantly to the network administrator via a

mobile device. Telegram is an essentially better messaging platform because it offers several features such as synchronization, fast service, reliable backup, and better security [8]. With this integration, whenever a disruption or status change occurs on the server, a notification will be sent by Zabbix directly to Telegram, so that the administrator can respond to the disruption immediately.

Cluster analysis has been widely studied and used in various fields [9], [10]. The most representative centroid-based clustering method is k-Means [11], [12] which is well-known for its high efficiency. For the analysis of network disruption data, the k-Means clustering method is used to group servers based on the frequency and type of disruptions that occur. K-Means clustering is a data analysis technique that groups data into several clusters based on the closeness of each data. By using k-Means, administrators can identify which servers experience the most frequent disruptions and look for certain patterns that may indicate the cause of server disruptions. With this analysis, more effective preventive measures and better planning can be taken to improve the stability and performance of the server network.

2 Methodology

The method used in this research uses the Agile method with the Kanban approach. Agile is a software development approach that focuses on iteration, team collaboration, response from users, and adjusting plans according to project needs. The Agile method provides flexibility to an organization or team to adopt a subset of principles and practices selected based on their culture, values, and the type of system they are developing [13]. One of the popular Agile methods is Kanban, which focuses on workflow visualization and continuous improvement. Kanban is a lean production method from the Toyota Production System, which was adapted in the early 2000s for software engineering [14]. The Agile approach with Kanban offers an effective way to manage software development projects, especially in contexts that require high flexibility and rapid response to change. Visualizing workflows, limiting work in progress, and focusing on continuous improvement, improves efficiency, quality, and stakeholder satisfaction. The use of Kanban in Agile allows for more transparent, and adaptive project management, which is critical in the dynamic world of software development. Using a Kanban board is simply equated to a more complex project board with certain rules for each queue [15].

2.1 Project Initiation

In the project initiation phase, our main objective was established: to monitor server downtime and uptime status in real time using Zabbix. In the initial phase,

the researcher created the system design to be developed, as illustrated in Figure 1. Additionally, the researcher set further objectives: enabling Zabbix to send notifications to the developed website and to Telegram in case of network issues, and storing historical data on server network disruptions in the website for further analysis. The researcher also identified the necessary hardware and software, including the servers to be monitored, the Zabbix Server operating system, Zabbix configurations, Zabbix Server version, and website development using PHP and MySQL databases. Integration with the web application and Telegram bot API was also planned in this phase. Additionally, the research schedule and system requirements were identified to ensure the research could be carried out effectively and on time.

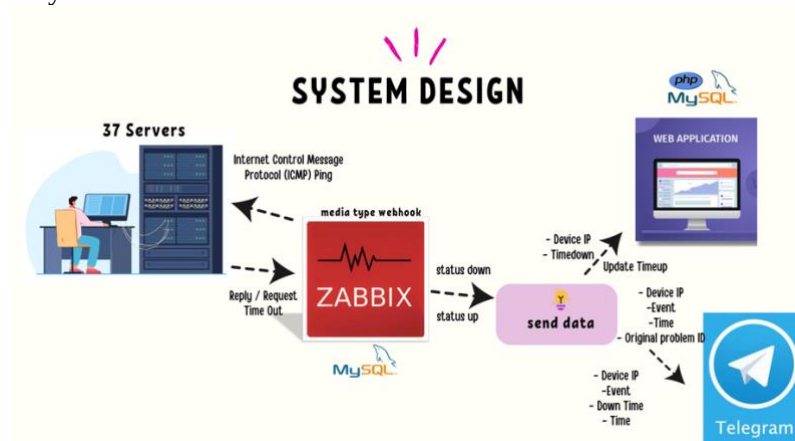


Figure 1 Research system design.

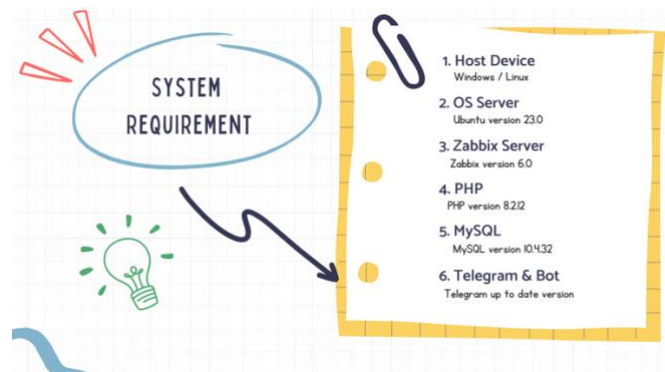


Figure 2 Research system requirements.

2.2 Planning

At the planning stage, the project was broken down into manageable, targeted tasks. These tasks were organized on a Kanban board with columns labeled "To Do," "In Progress," and "Done." Tasks were prioritized based on their importance and order of completion. Initial tasks included installing the Zabbix Server, configuring webhooks in Zabbix, developing an API for Telegram integration, and creating a web-based server disruption monitoring application. Using the Kanban board, the researcher could quickly view and manage the tasks that needed to be completed. An illustration of the Kanban board used in this study is shown below.

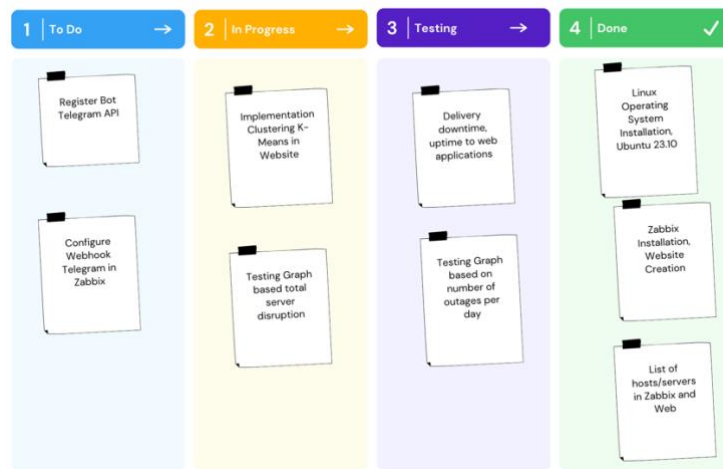


Figure 3 Implementation of research kanban board.

2.3 Implementation

In the implementation phase, the researcher began working on previously identified tasks and moved them to the "In Progress" column on the Kanban board. Development was carried out in stages and continuously, with the researcher focusing on completing small tasks and testing each developed feature. The researcher limited the number of tasks in progress to maintain focus on the tasks at hand. These tasks included adding hosts (servers) in Zabbix, configuring Webfig for web and Telegram, building the website using PHP and MySQL database, implementing K-Means Clustering, registering the Telegram Bot API, and developing features such as monitoring server downtime and uptime status, disruption notifications, and storing historical data on server network disruptions.

The screenshot shows the 'New host' configuration form in Zabbix. The 'Host name' field is set to 'Server Zoom Distribusi'. The 'Visible name' field is also 'Server Zoom Distribusi'. The 'Templates' field has 'Template: WebService (v.3.8)' selected. The 'Groups' field has 'Group: Servers (v.3.8)' selected. The 'Interfaces' table shows one interface with 'Type' as 'Agent', 'IP address' as '192.168.51.8', 'DNS name' as empty, 'Connected to' as 'IP', 'Port' as '161', and 'Default' as checked. The 'Description' field is empty. The 'Monitored by proxy' dropdown is set to '(no proxy)'. The 'Enabled' checkbox is checked. 'Add' and 'Cancel' buttons are at the bottom right.

Figure 4 Adding hosts (servers) in zabbix.

The screenshot shows a 'Data Server' form in a website application. The 'Nama Server' field contains 'Server Zoom Distribusi'. The 'Alamat Server' field contains '192.168.51.8'. At the bottom, there are 'Save' and 'Clear' buttons.

Figure 5 Adding hosts (servers) in website application.

The following is the webhook media type configuration for sending data to the website.

```
JavaScript
1 var Website = {
2   ipAddress: null,
3   statusserver: null,
4
5
6   sendMessage: function() {
7     var params = {
8       text: Website.ipAddress,
9       text2: Website.statusserver,
10      disable_web_page_preview: true,
11      disable_notification: false
12    },
13    data,
14    response,
15    request = new XMLHttpRequest(),
16    url = "http://192.168.3.9/web_analisa/typopost_dataip?" + Website.ipAddress + "&status=" + Website.statusserver,
17    request.setRequestHeader("Content-Type: application/json");
18    data = JSON.stringify(params);
19    response = request.Post(url);
20
21    try {
22      response = JSON.parse(response);
23    }
24    catch (error) {
25      response = null;
26    }
27
28  }
29
30
31
32 }
33
34 try {
35   var params = JSON.parse(value);
36
37   Website.ipAddress = params.Message;
38   Website.statusserver = params.Subject;
39   Website.sendMessage();
40 }
41
64457 characters remaining
```

Figure 6 Zabbix javascript webhook for sending data to website applications.

Next, a bot request was made to Telegram's BotFather, along with the configuration of the webhook media type for Telegram using the Telegram bot API.



Figure 7 Request telegram API bot.

A screenshot of the Zabbix webhooks configuration page. The 'Name' field is set to 'Telegram' and the 'Type' is 'Webhook'. The 'Parameters' section includes fields for Message (ALERTMESSAGE), ParseMode (html), Subject (ALERT.SUBJECT), To (ALERT.SENDTO), and Token (700678). The 'Script' field contains a shell script snippet: 'var Telegram = [...]'. The 'Timeout' is set to 10s. The 'Enabled' checkbox is checked. At the bottom, there are buttons for 'Update', 'Clone', 'Delete', and 'Cancel'. A description at the bottom provides instructions for registering the bot and sending messages.

Figure 8 Zabbix webhook configuration for sending data to telegram with entering token access from telegram's bot.

In K-Means clustering, a query is made to the MySQL database to obtain the server IP, server name, and the number of disruptions for each server. The number of clusters (K) was determined using the **Elbow Method**, where the optimal value of K is chosen at the point where the decrease in inertia starts to slow down significantly.

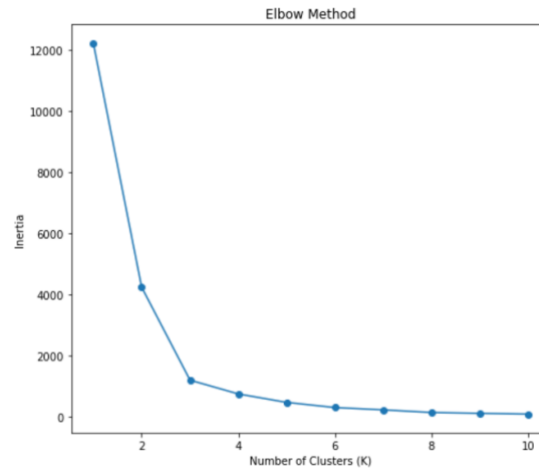
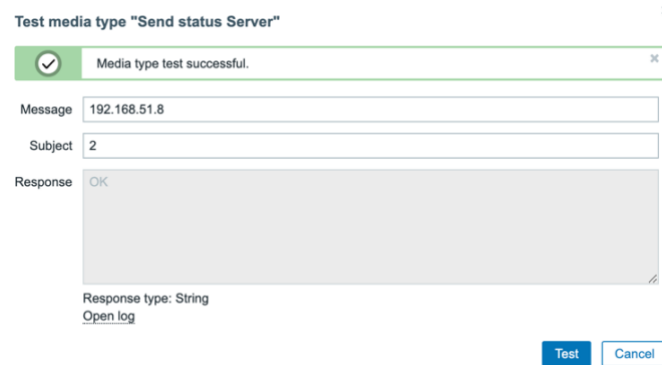


Figure 9 Optimal K Selection Using Elbow Method

Based on the Elbow Method graph, the optimal number of clusters was determined to be 3. The data is then divided into these three clusters, associating the clustering results with the original data and grouping the data based on the specified clusters. The data is encoded into JSON format and prepared for visualization in a scatter plot using JavaScript. Next, the Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE) are calculated. The error calculations are performed using the mean absolute error (MAE) and root mean squared error (RMSE) methods because both provide values on a measurement scale, allowing for a clearer view of how close the measurements are to the predictions [16].

2.4 Testing and Integration

The testing step was carried out once the features were created to make sure they all worked as intended. Any faults or problems discovered during testing were fixed by the researchers. The enhancements were incorporated into the system after testing was finished. The Telegram API bot, webhooks, web apps, and Zabbix were among the system components that the researchers made sure were operating correctly.



Test media type "Send status Server"

Media type test successful.

Message: 192.168.51.8

Subject: 2

Response: OK

Response type: String
Open log

Test Cancel

Figure 10 Testing media types webhook sending status *down* to website application.

2.5 Monitoring and Evaluation

At this stage, a procedure for monitoring and assessing the outcomes of the work is conducted regularly. Both ongoing and completed projects are evaluated, and adjustments are made in response to new information and evolving needs. Data from the monitoring system is stored in a database, and servers are grouped according to the frequency of disruptions using the k-Means clustering technique. The researchers use this analysis to identify trends and patterns in disruptions, which helps plan server improvements that address underlying issues and enhance the performance and reliability of the servers.

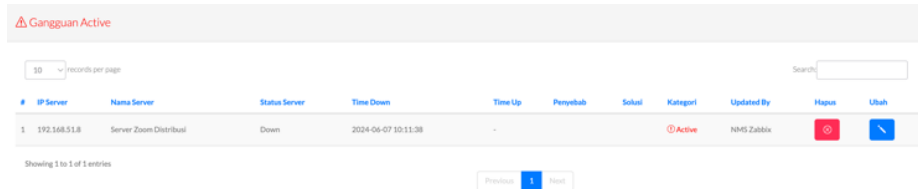
2.6 Completion and Documentation

The completion stage includes the finalization of the project, where researchers ensure that all planned and developed features are running well. All remaining tasks are moved to the "Done" column on the Kanban board. After that, documentation is made covering the process of creating and configuring the application. The results of the server network disruption data analysis are then documented and used as recommendations for further improvements. The monitoring system then enters the production stage and routine maintenance will be carried out to ensure the system is running well. User feedback continues to be collected as a reference for future system updates and improvements.

3 Results

If server network downtime occurs, the downtime status will be sent by Zabbix to the web application that has been built. In the web application, the server

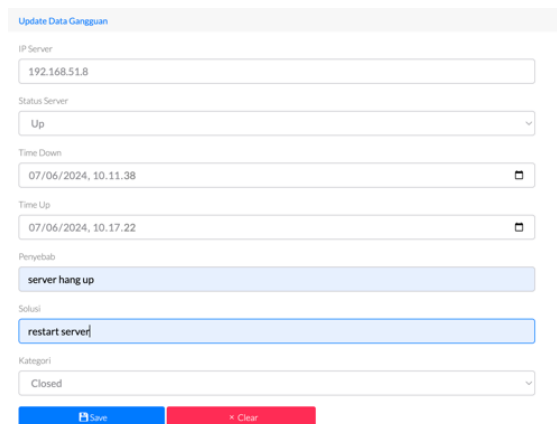
network disruption (downtime) sent by Zabbix includes the server IP and server status as parameters. This data is sent to a MySQL database and displayed on the website.



#	IP Server	Nama Server	Status Server	Time Down	Time Up	Penyebab	Solusi	Kategori	Updated By	Hapus	Ubah
1	192.168.51.8	Server Zoom Distribusi	Down	2024-06-07 10:11:38	-			Active	NM0 Zabbix		

Figure 11 Active disruptions display on the website application.

When the server is back up, Zabbix will send the "up" status to the MySQL database, and the disruption will move to the "Closed Disruptions" column. At this stage, the administrator will validate the disruption by providing the cause and solution for the issue.



Update Data Gangguan

IP Server: 192.168.51.8

Status Server: Up

Time Down: 07/06/2024, 10:11:38

Time Up: 07/06/2024, 10:17:22

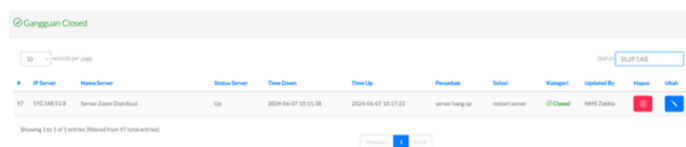
Penyebab: server hang up

Solusi: restart server

Kategori: Closed

Figure 12 Filling in the cause and solution of the server disruption by the administrator.

In the Closed Disruptions section, there is data on the Server IP, Server Name, Server Status, Time Down, Time Up, Cause, Solution, Category, Updated By, Delete, and Edit. Thus, the disruptions are well-documented in the web application that has been built.



#	IP Server	Nama Server	Status Server	Time Down	Time Up	Penyebab	Solusi	Kategori	Updated By	Hapus	Ubah
97	192.168.51.8	Server Zoom Distribusi	Up	2024-06-07 10:11:38	2024-06-07 10:17:22	server hang up	restart server	Closed	NM0 Zabbix		

Figure 13 Display of closed disruptions on the web application.

This website also features a menu for visualizing network disruptions that have occurred, including a Graph of the Number of Disruptions Per Day and a Graph Based on the Number of Server Disruptions.

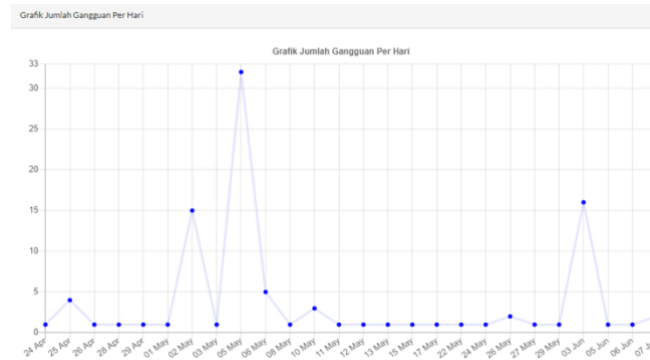


Figure 14 Graph of the number of disruptions per day on the web application.

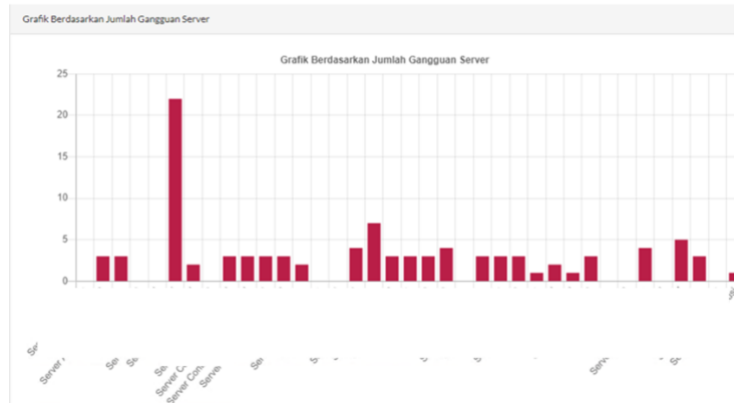


Figure 15 Graph based on the number of server disruptions on the web application.

The results of clustering using K-Means were obtained by grouping servers into 3 clusters, with the optimal number of clusters determined using the Elbow Method. Each cluster represents a distinct pattern of server disturbances: Cluster 2 corresponds to servers with frequent disturbances and requires immediate attention, Cluster 3 represents servers with moderate disturbances that may need regular monitoring, while Cluster 1 includes stable servers with rare disturbances. This clustering approach helps prioritize server maintenance and effectively identify patterns of server disturbances based on their frequencies. The K-Means clustering results are visualized in the following scatter plot.



Figure 16 Scatter plot using K-Means clustering.

Notification of server network downtime, with the downtime status also sent by Zabbix to Telegram. In the Telegram application, the server network disruption (downtime) sent by Zabbix includes parameters such as Device IP, Event, Time, and Original Problem ID. When the disruption is resolved and the server network is back up, Zabbix will send the parameters Device IP, Event, Down Time, and Time. The following image is an example of a notification from Zabbix to Telegram.

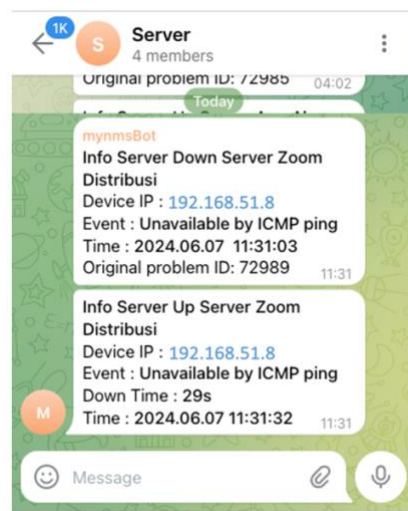


Figure 17 Downtime and uptime notifications in the telegram application.

4 Conclusions

This research successfully developed and implemented a web-based Network Monitoring System using Zabbix to monitor the status of server downtime

(inactive time) and uptime (active time) in real time. The system can also send notifications directly to the web application that has been created and to the Telegram application, making it easier for network administrators to quickly address disruptions. By storing historical data on network disruptions in a website built using PHP and MySQL, the system allows for further analysis to detect trends and patterns of disruptions. Additionally, this system can group servers using the K-Means method based on the number of disruptions that occur. This information is very useful for strategic planning to enhance server network reliability, identify servers that need to be prioritized for maintenance, and implement more effective preventive measures against network disruptions.

Acknowledgements

We extend our gratitude to all parties at PT Perusahaan Listrik Negara (Persero) and Bandung Institute of Technology for their support in helping us complete this research.

References

- [1] J. Renita, B. Noble, and N. Edna Elizabeth, "Network's Server Monitoring and Analysis Using Nagios," 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), doi: 10.1109/WiSPNET.2017.8300092
- [2] A. Mardiyono, W. Sholihah and F. Hakim, "Mobile-based Network Monitoring System Using Zabbix and Telegram," 2020 3rd International Conference on Computer and Informatics Engineering (IC2IE), doi: 10.1109/IC2IE50715.2020.9274582
- [3] S. S. Kamenov, "Experimental monitoring on network based tactile sensing system," 2019 28th Int. Sci. Conf. Electron. 2019 - Proc., pp. 1–4, 2019, doi: 10.1109/ET.2019.8878661.
- [4] A. Hidra, Defni., D. Prayama, and F. Agustin, "Design and Implementation of Network Monitoring System Using Nagios with Email and SMS Alert," J. Ilm. Poli Rekayasa, vol. 10, no. 1, p. 42, 2014, doi: 10.30630/jipr.10.1.56.
- [5] Y. G. Shan, L. Chao, G. Guangjian and F. Gao, "Research on Monitoring of Information Equipment Based on Zabbix for Power Supply Company," 2021 3rd International Conference on Applied Machine Learning (ICAML), Changsha, China, 2021, pp. 487-491, doi: 10.1109/ICAML54311.2021.00108.
- [6] J. Kiruthika, S. Khaddaj, D. Greenhill and J. Francik, "User Experience Design in Web Applications," 2016 IEEE Intl Conference on Computational

- Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES), Paris, France, 2016, pp. 642-646, doi: 10.1109/CSE-EUC-DCABES.2016.253.
- [7] Telegram, "Telegram APIs," Telegram, [Online]. Available: <https://core.telegram.org/>. [Accessed 12 08 2020].
 - [8] T. Sutikno, L. Handayani, D. Stiawan, M. A. Riyadi and I. M. I. Subroto, "Whatsapp, Viber and Telegram: which is the Best for Instant Messanging?," International Journal of Electrical and Computer Engineering (IJECE), vol. 6, no. 3, pp. 909-914, 2016.
 - [9] A. K. Jain, "Data clustering: 50 years beyond K-means," Pattern Recognit. Lett., vol. 31, no. 8, pp. 651–666, 2010.
 - [10] R. Xu and D. Wunsch, "Survey of clustering algorithms," IEEE Trans. Neural Netw., vol. 16, no. 3, pp. 645–678, May 2005.
 - [11] J. Macqueen, "Some methods for classification and analysis of multivariate observations," in
 - [12] S. P. Lloyd, "Least squares quantization in PCM," IEEE Trans. Inf. Theory, vol. 28, pp. 129–136, Mar. 1982.
 - [13] S. Soundararajan, J. D. Arthur, and O. Balci, "A methodology for assessing agile software developemtn methods," Agile Conference, Aug 2012, doi: 10.1109/Agile.2012.24.
 - [14] P. Mary and T. Poppendieck, "Lean software development: An agile toolkit," Nachdr., Addison-Wesley, 2010.
 - [15] R. Polk, "Agile and kanban in coordination," in Proc. Agile Conf., 2011, pp. 263–268.
 - [16] Paula Zenni Lodetti, Edison A. C. Aranha Neto et al., "MAE and RMSE Analysis of K-means Predictive Algorithm for Photovoltaic Generation", International Conference on Electrical, Computer and Energy Technologies (ICECET), Prague, Czech, 20-22 July 2022.