

Implementation SEMAR-IoT-Platform for Vehicle as a Mobile Sensor Network

Yohanes Yohanie Fridelin Panduman[#], Sritrusta Sukaridhoto^{*}, Anang Tjahjono[#],
Rizqi Putri Nourma Budiarti^{**}

[#] *Electrical Engineering Department, Politeknik Elektronika Negeri Surabaya, Surabaya, Indonesia*
E-mail: yohanyfp@pasca.student.ac.id, anang.tj@pens.ac.id

^{*} *Informatics and Computer Department, Politeknik Elektronika Negeri Surabaya, Surabaya, Indonesia*
E-mail: dphoto@pens.ac.id

^{**} *Engineering Department, Universitas Nahdlatul Ulama Surabaya, Surabaya, Indonesia*
E-mail: rizqi.putri.nb@unusa.ac.id

Abstract— With the rapid development of IoT technology in various fields such as smart cities and industry 4.0, the need for wireless sensor network-based systems has increased, one of which is the concept of using a vehicle as a mobile sensor network or known as VaaMSN. Many developers use the IoT platform as a cloud computing service in developing the VaaMSN system. However, not all IoT platform service providers provide monitoring features on every device and provide information such as device location, purpose, condition. Therefore, this research aims to develop an IoT Platform that can receive data and provide information on each device, making it easier to process data and control devices. Therefore, this research aims to develop an IoT platform called the SEMAR-IoT-Platform that able to received data and provide information on each device for easier data processing and control devices. The SEMAR-IoT-Platform integrates Big Data, Data Analytics, Machine Learning, using the principles of Extract, Transfer, and Load (ETL) for data processing and provides communication services using HTTP-POST, MQTT, and NATS. The test results show that the system has been successfully implemented to complement a simple IoT system with an average delay time of HTTP, NATS, and MQTT communications of less than 150ms for the data storage process, and for the data visualization process has an average delay time of less than 300ms.

Keywords— Internet of Things; Internet of Things Platform; VaaMSN; Cloud Computing; Big Data; Mobile Sensing.

I. INTRODUCTION

Connecting everyday devices to the Internet is the current Internet of Things (IoT) paradigm, with the ability to remotely monitor or control devices around us [1] and be able to implement in various fields such as smart cities, the smart industry even in industry 4.0, smart environment, and smart aquaculture. Because of this ability, it causes IoT technology to grow rapidly, according to Cisco [2] that 50 billion object devices will be connected to the internet by 2020, besides other research [3] estimates that by 2025 there will be 64 billion devices connected to the internet.

Smart cities require a system to collect public data in a large area. Thus, it uses the concept of a wireless sensor network [4] as a data collection solution, but in general, the system used is based on fixed sensors, which requires higher costs. Besides, in the industrial era 4.0, which changed the entire business system, it has influenced the methods used by

industry [5], such as in the logistics sector [6] requiring a system to monitor stock and vehicle conditions. Several previous research on the integration of mobile sensors on the vehicle with cloud computing shows that the system has been implemented in various fields. In 2017, there was research on a mobile sensor-based traffic monitoring system installed on the vehicle and connected to cloud computing services [7]. Another research in 2019 presents the use of WSN devices mounted on vehicles to monitor gas leaks in certain areas [8]. Therefore, placing a sensor device on a vehicle to monitor the condition of the car or the environment around the car is a concept of Vehicle as a Mobile Sensor (VaaMS) [9] or it can also be called the Vehicle as a Mobile Sensor Network (VaaMSN) [10] can be a solution to this problem. However, in developing a VaaMSN system, it is necessary to consider cloud computing services to receive, process, and display data.

Some constraints in developing cloud computing services are often experienced by developers such as the need for large resources or lack of ability from developers, causing relatively long development time. Several companies are developing IoT platform services as a solution to this problem. The IoT platform includes communication between devices or sensors, data storage, and the application service environment in general, the IoT platform is generally placed on a cloud server [11]. However, not all IoT platform service providers provide monitoring features on every device and provide information such as location, destination, conditions. Also, the need to aggregate data to facilitate processing but still store message data in its entirety.

One ongoing research on the development of VaaMSN that integrates with cloud computing services for the monitoring process in the road environment. Implementation of VaaMSN as edge computing with SEMAR cloud (Smart and Analytical Environmental Monitoring in Real-Time) to monitor air quality [12], the developed VaaMSN system consists of single-board computing (SBC) connected with 4G Wi-Fi modem devices, GPS and air sensor. Besides that, the VaaMSN system was also developed to detect holes using an IMU sensor attached to the SBC, location data, and the IMU is sent to cloud computing services to be classified whether or not there are holes [13]. The researchers also built a road condition monitoring system by improving from the previous system, namely by applying vision-based sensors. A portable wireless camera is mounted on the front of the car to be able to detect holes in the road, captured images are processed on VaaMSN using a data model that has been developed, hole images are sent to the cloud server along with the current time and location from the GPS sensor [14]. But the SEMAR cloud computing system that was developed is still focused on certain systems. Therefore, SEMAR can be upgraded as an IoT platform that can be used on VaaMSN systems with various types of monitoring. IoT platform system that can receive data and provide information on each device, making it easier to process data and control devices. This journal consists of five parts

consisting of the introduction of this research is shown in section 1. Section 2 presents the detailed method and system design used in this research. Section 3 explained the discussion about the results of implementation and testing that have been produced in our research. In section 4, we explained the conclusions of this research and future work plans as a continuation of this research

II. METHODS AND SYSTEMS DESIGN

In this research, the development of the SEMAR-IoT-platform system consists of several methods and steps. First, we focused on needs and characteristics analyzing in developing an IoT platform with a comparison method to get the parameters and features of an IoT platform system. The results of the comparison would be used as a reference for designing the SEMAR-IoT-Platform system architecture. In the next step, we would develop an IoT platform based on this design. The final step is platform testing on the VaaMSN system protocol to test the performance of the SEMAR-IoT-Platform.

A. IoT Platform Analysis and Comparison

Several previous research on developing IoT platforms as cloud computing services for smart factories, the results show that the system design consists of several parts such as communication protocols using Kafka / Redis, databases, data visualization, and data analysis services using machine learning algorithms [15]. Other research shows the creation of an IoT platform architecture implemented for edge computing using a transparent computing-based IoT architecture, this architecture aims to provide services dynamically for lightweight IoT devices [16]. The cloud computing service proposed in this research consists of an edge server layer that functions to distribute computation, control, and storage on an edge computing network. This technology concept was able to be used as a reference for developing an IoT platform for the VaaMSN system.

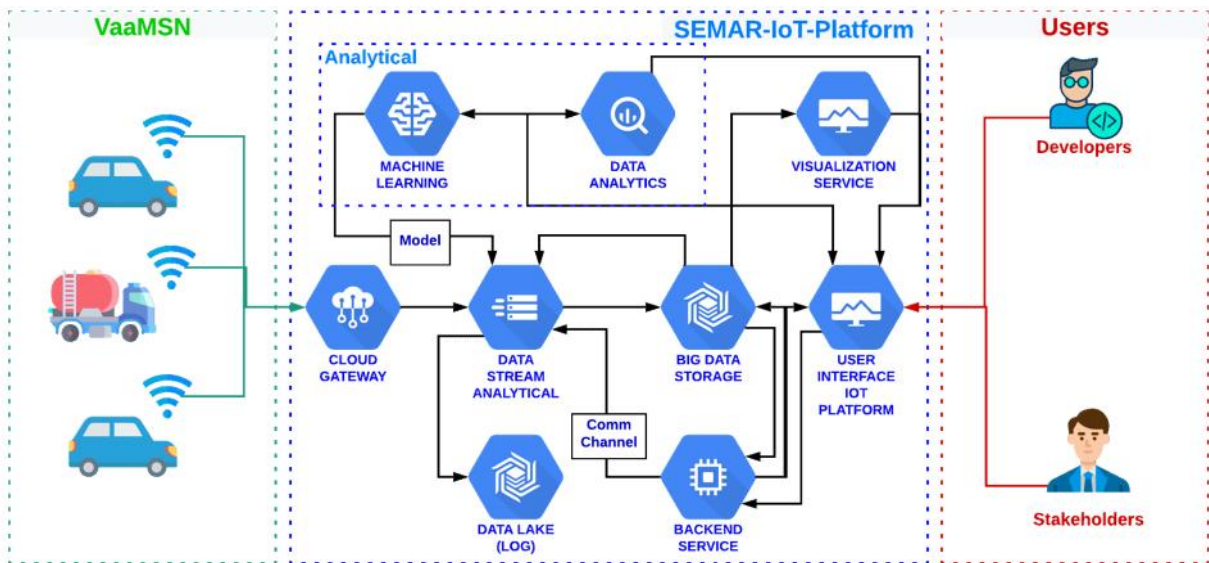


Fig. 1. The architecture of the SEMAR-IoT-Platform

B. SEMAR-IoT-Platform Architecture

In this section, we will present the SEMAR-IoT-Platform architecture which was developed based on several parameters and criteria as a result of the analysis and comparison process of several platforms. Figure 1 shows the IoT platform system architecture that we propose in this study consists of several parts such as Backend Services as the main service to control the system, User Interface as the interface of SEMAR-IoT-Platform for users, Cloud Gateway which consists of several communication protocol services, Data Stream Analytical, Big Data Storage, Data Lake for storing sensor data logs, Visualization Services consisting of services for data visualization and Analytics services consisting of Machine Learning services and Data Analysis services.

Backend service is the main service for managing every process on the IoT Platform. We develop backend services using Python programming with the Cloud9 framework based 'modal controller' running on port 4000. The backend service consists of several key features such as authentication, personal user management, user group management, device group management, device management, and communication channel management. Each of these services has a page on the user interface. In this research, we developed a user interface based on PHP programming. The authentication feature functions as authentication and identification of users who can access SEMAR-IoT-Platform. The personal user management feature functions for the management of personal data such as names, usernames and e-mails of users. SEMAR-IoT-Platform provides a user group feature so that users able to share IoT projects.

The device group management feature is a feature for registering device groups owned by the user, the device group in question is an IoT project undertaken by the user, in this feature the user can create a group that is private or public, besides the communication and data storage process based on device group that has been registered. Therefore, there is a communication channel management feature on the device group management features page. The communication channel management feature functions to manage the communication channels that are active or used by users. As shown in Figure 1, when a user changes communication data such as adding, activating or deactivating communication, the backend service will send a message to the Data Stream Analytical service to determine the active communication channel. Besides, every data transmission made from the VaaMSN device must match the existing code in the device group. The device management feature functions to register each device used along with the data fields to be sent from the device, this feature will provide a code for each registered device, the code must be included when sending data from the VaaMSN device, this is because the code functions as the identity of the device.

Cloud Gateway is a communication service as the main gateway that connects the IoT Platform with VaaMSN devices. We developed 3 communication protocol services consisting of HTTP-POST [20] running on port 3001, MQTT [21] using services from Mosquitto on port 1883

and NATS [22] using NATS Server service running on port 4222. This service will receive data from the VaaMSN data device in JSON format. Data received from the Cloud Gateway will be forwarded to the Data Stream Analytical service.

Data Stream Analytical is a service that functions to distribute data sent to cloud gateways to other sub-systems on the IoT Platform. Each communication protocol on the Cloud Gateway has its own Data Stream Analytical service, but with the same work process. As previously explained, this service can only receive data from a communication channel that is activated by the user through the communication channel management feature, so that if a message is sent from an unregistered or inactive communication channel then the message is not processed by the system. In this service we develop Extract, Transform and Load methods for data processing when a message is forwarded by the Cloud Gateway service, the message must be in JSON form and contain information such as device code.

The Extract process will convert the message into a JSON object, the results of the extract process will be stored in the Data Lake, where the data to be stored on the Lake will be added to the original message (string), the time of receipt on the server, the type of communication channel and other information such as message topics for MQTT, message subject for NATS, access token and IP sender for HTTP-POST. Data Lake is a service that serves as a place to store log data from VaaMSN before the data undergoes a transformation process on the Data Stream Analytical service.

The Transform process aims to change the form of the data to match the form that was registered on the previous IoT platform. The system will retrieve the registered column data along with the device code in the message sent, then the JSON object data will be transformed according to the column data. This process aims to facilitate the use of sensor data for further processes such as visualization, prediction, and data analysis. Furthermore, the system will perform a data prediction process using a data model that has been created on the Analytical service using a previously created dataset. The results of the prediction process will be added to the results of the Transformation process. The last process is Load, the data that has been processed to be loaded and stored on the Data Storage service and Visualization service.

Visualization service as a service to visualize sensor data owned by users. We use the Elasticsearch service running on port 9200 to store sensor data which is used specifically for data visualization, on the user interface page, we developed a system to retrieve Elasticsearch data in tabular form. In addition to this, there is a Kibana service that runs on port 5601 to create visualizations, such as line diagrams, pie charts and maps. The visualization results created in Kibana will be exported to the user interface.

This service aims to provide analysis services to users. Analytical services consist of Machine Learning services and Data Analysis services. Data Analysis Services aims to provide business analysis based on sensor data that users have such as determining a summary of sensor data for a certain time so that it can make it easier for users to

represent the results of sensor data sent to the VaaMSN device. We use the Metabase service to provide business analysis. Besides, the VaaMSN tool is very close to the geospatial analysis associated with this position. With the Data Analysis service, users can determine the distribution of sensor data based on a certain area. The results of the data analysis process will be sent to the user interface page for easy viewing by users.

Machine Learning services aim to produce a model based on the sensor data obtained. We develop machine learning services using supervision-based algorithms, such as Support Vector Machine [23] and Decision Tree [24]. This service was developed on a python basis using Scikit-learn for the data training process to produce models based on data that has been stored in Data Storage. The results of the data model will be used during the Transformation process in the Data Stream Analytical service.

C. Experiment Method

In this section, we will describe several methods that are used to test the SEMAR-IoT-Platform system that has been developed. The first test is to implement SEMAR-IoT-Platform on a simple VaaMSN system. System testing consists of creating new device groups, adding communication channels, adding new data devices with a certain format, sending data from the IoT Platform using one of the communication channels to displaying data continuously.

The second method is testing the website page performance of the SEMAR-IoT-Platform using a tool from <https://www.webpagetest.org/> to test the access speed of a website using several types of internet connections. The parameter in this test consists of several parameters [25] such as Time to First Byte, which is the time duration from the HTTP request to the first byte of the browser page. Start Render Time is the time when something is first displayed from a blank page. Parameter Document Complete Time the time it takes to load all content on the page without executing javascript. Fully Loaded Time measures the time the browser has loaded all content until there is no network activity for 2 seconds.

The third method is communication testing by measuring the time it takes from the data sent by the device to the IoT Platform until the data is displayed. Data communication testing was carried out as many as 300 (n) repetitions with a delivery time carried out every 1 second, in this test we used 2 different internet service providers. When the data will be sent from the VaaMSN device to the Cloud Gateway, the system will add the delivery time (T_1) to the JSON message before the message is sent to the server. Furthermore, before the data sent is stored in data storage, the system will add the value of the time the data will be stored in the Data Storage (T_2). The result of T_2 reduced by T_1 shows the delay in sending data from the VaaMSN device to the data storage. The average results (\bar{x}_1) of each experiment will be stored in the test results table with the following equation model:

$$\bar{x}_1 = \frac{\sum_{i=1}^n (T_{2i} - T_{1i})}{n} \quad (1)$$

The data visualization page will be displayed continuously with a data refresh every 1 second. The data visualization delay time that able to be calculated by adding the time needed to refresh the data on the visualization display (T_3) with the delay time on data storage ($T_2 - T_1$). The average results of each experiment (\bar{x}_2) will be stored in the test results table with the following equation model:

$$\bar{x}_2 = \frac{\sum_{i=1}^n ((T_{2i} - T_{1i}) + T_3)}{n} \quad (2)$$

A. VaaMSN for Air Quality Monitoring Systems

This section describes the design of the VaaMSN system used for SEMAR-IoT-Platform testing, we develop the VaaMSN system for air quality monitoring systems in certain areas. The system consists of a single board computing (SBC) connected to the air sensor, GPS sensor and 4G WiFi as shown in Figure 2.

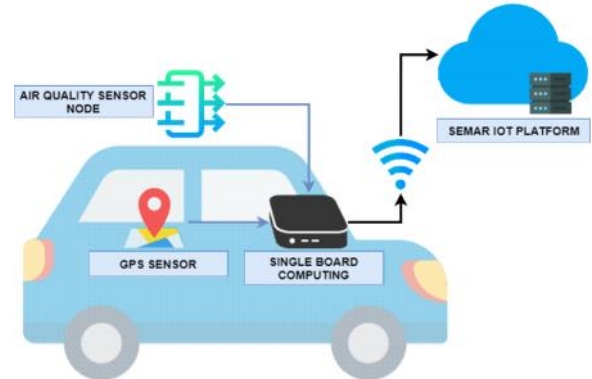


Fig. 2. VaaMSN for Air Quality Monitoring Systems Design

The air sensor device is placed on top of the vehicle which consists of a microcontroller device connected to 5 air sensors such as MQ131 (ozone), MQ7 (carbon monoxide), MiCS 2714 (nitrogen dioxide), MQ135 (sulfur dioxide), Shinyei PPD42 (particular / dust) as shown in Figure 3. Air sensor device data will be sent to SBC using the MQTT communication protocol connected via a 4G WiFi wireless network. SBC acts as edge computing to process sensor data before it is sent to the IoT platform.

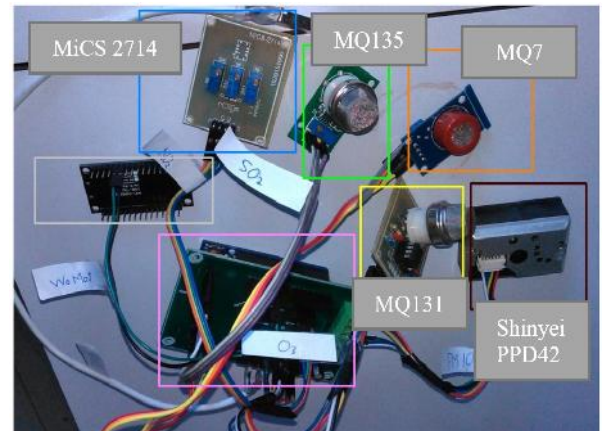


Fig. 3. Air Quality Sensor Node

When the air sensor data is received, the SBC will retrieve location data (latitude and longitude) from the GPS sensor to be added to the air sensor data. Furthermore, the SBC system adds a delivery time in the form of a Unix Timestamp and device code that registered on the IoT platform on that data. Therefore, the data sent in JSON format consists of "device_code", "date_add" for the current time, "pm10", "no2", "co", "so2", "o3" for air quality data and "gps.latitude", "gps.longitude" for location data. The VaaMSN system sends data continuously every 5 seconds using MQTT communication with topics according to the communication channels registered in SEMAR-IoT-Platform

III. RESULTS AND DISCUSSION

In this section we explain the results of system testing that has been carried out based on the methods described in sub-chapters 3.C and 3.D, system testing consists of SEMAR-IoT-Platform testing by implementing the IoT Platform on the VaaMSN system for air quality monitoring, testing web page performance and communication testing.

A. Implementation of SEMAR-IoT-Platform on VaaMSN Systems

The results of the test show that the features of the SEMAR-IoT-Platform system have been successfully implemented as cloud computing services for the VaaMSN system. In the device group management feature, the process of adding a new device group has been successfully carried out. The communication channel management feature has been functioning properly, where the system can automatically run the selected communication protocol service.

Fig. 4. User Interface for Add New Device

In the device management feature, the process of adding a new device has been successfully carried out, the column content on the device consists of "pm10" for Shinyei PPD42 sensor data, "o3" for MQ131 sensor data, "so2" for MQ135 sensor data, "no2" for sensor data MiCS 2714, "co" for MQ7 sensor data, "gps" which consists of

"latitude" and "longitude" for GPS sensor data. Figure 4 shows the user interface page for adding a new device.

The VaaMSN device has been successfully developed in accordance with the system design described in section 3.D. Figure 5 presents the placement of the air quality monitoring sensor node on the vehicle, the sensor is placed on top of the vehicle which aims to facilitate air quality measurement. Air sensor data has been successfully sent to the SBC device for further processing and forwarding to SEMAR-IoT-Platform.



Fig. 5. Air Quality Sensor Node on Vehicle

The data that has been sent by VaaMSN has been received by the MQTT Cloud Gateway communication channel and forwarded to the Data Stream Analytical service for processing using the Extract, Transform and Load methods. The experimental results show that the data sent has been successfully received, processed and stored in Data Storage and Elasticsearch. Data on Elasticsearch will be displayed using the Kibana service embedded on the user interface page, Figure 6 shows the visualization of user interface data in this test.

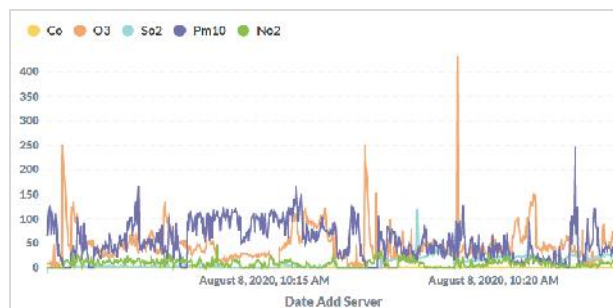


Fig. 6. User Interface for Data Visualization

B. Web Page Performance Testing

This experiment is used to test how the performance of a website page when accessed using several types of internet connection. Table 1 shows the results of tests carried out using several types of internet networks provided by the equipment, such as 3G Slow networks with speeds of 400Kbps and RTT 400ms (Round-trip time), 3G networks with speeds of 1.6Mbps and RTT 300ms, 4G networks with speeds of 9Mbps and 170ms RTT, as well as LTE Network with speeds of 12Mbps and 70ms RTT. Based on the experimental method, the parameters in this test consist

of Time to First Byte, Start Render Time, Document Complete Time and Fully Loaded Time.

TABLE I
WEB PAGE PERFORMANCE RESULT

Conne- ction	Time to First Byte	Start Render Time	Document Complete Time	Fully Loaded Time
3G Slow (0.4 Mbps)	0.92 s	1.40 s	1.32 s	1.32 s
3G (1.6 Mbps)	0.65 s	1.15 s	1.03 s	1.03 s
4G (9 Mbps)	0.37 s	0.75 s	0.58 s	0.58 s
LTE (12 Mbps)	0.14 s	0.50 s	0.39 s	0.38 s

C. Communication Testing

This experiment is used to test how the system's performance in sending data from VaMSN devices to the IoT Platform service.

TABLE II
AVERAGE DELAY TIME OF COMMUNICATION TESTING

Comm. Channel	Provider 1		Provider 2	
	Data storage	Data Visualiza- tion	Data storage	Data Visualiza- tion
HTTP- POST	93.31 ms	207.94 ms	136.80 ms	283.66 ms
MQTT	42.31 ms	155.78 ms	92.19 ms	238.01 ms
NATS	50.67 ms	160.69 ms	73.66 ms	218.11 ms

Table 2 shows the results of the communication test. We used 2 different internet service providers. The first provider has an internet speed of 4ms with 2.05 Mbps for upload and 9.86Mbps for download. The second provider has an internet speed of 58ms with 0.7 Mbps for upload and 22.09 Mbps for download. The server location is in Singapore while our test location is in Surabaya, Indonesia. The test result shows that HTTP-POST communication has more delay time compared to other communication channels, meanwhile, the MQTT communication channel has the least delay time.

IV. CONCLUSIONS

In this research, the SEMAR system improvement has been successfully implemented into an IoT platform for the VaaMSN system. Experiments show that the developed IoT platform system can be implemented in the VaaMSN system for air quality monitoring. SEMAR-IoT-Platform can process data on sensor data sent by VaaMSN devices using Extract, Transform and Load methods based on data structures that have been registered by the user in the device management feature, making it easier for further data processing. The results of data processing have been successfully stored in Data Storage and Elasticsearch for continuous data visualization on the user interface page. In the communication channel feature, the system has

successfully implemented 3 communication channel services consisting of MQTT, HTTP-POST and NATS, the test results show that the average communication delay is less than 150ms for the data storage process, the data visualization process has an average delay time. - an average of less than 300ms. The communication channel service that has the smallest delay time is MQTT communication.

In the future, we will implement vision-based VaaMSN sensor data processing, so that the SEMAR-IoT-Platform to be built is expected to implement image processing and deep learning systems for processing data in the form of images.

ACKNOWLEDGMENT

This research was funded from "Applied Leading Research in Higher Education (in Indonesia Penelitian Terapan Unggulan Perguruan Tinggi)" by KEMENRISTEKDIKTI in 2020 with the title "Implementation of Vehicle as a Mobile Sensor Network is integrated with the Smart Environment Monitoring and Analytics in Real-time (SEMAR) system as a Road Surface and Environmental Monitor to support Smart City (in Indonesia - Implementasi Vehicle as a Mobile Sensor Network terintegrasi dengan Smart Environment Monitoring and Analytics in Real-time (SEMAR) system sebagai Pemantau Permukaan Jalan dan Lingkungan untuk mendukung Smart City)". With the funding scheme number "130/SP2H/AMD/LT/DRPM/2020"

REFERENCES

- [1] J. Moonej, W. K. G. Seah, A. Pekar, and I. Zolotova, "Decentralised IoT Architecture for Efficient Resources Utilisation," IFAC-PapersOnLine, vol. 51, no. 6, pp. 168-173, 2018, doi: 10.1016/j.ifacol.2018.07.148.
- [2] D. Evans, "The internet of things: How the next evolution of the internet is changing everything," CISCO white Pap., vol. 1, pp. 1-11, 2011.
- [3] P. Newman, "IoT Report: How Internet of Things technology growth is reaching mainstream companies and consumers," Bus. Insid., vol. 28, 2019.
- [4] U. Umar, M. U. H. Al Rasyid, and S. Sukaridhoto, "Distributed database semantic integration of wireless sensor network to access the environmental monitoring system," Int. J. Eng. Technol. Innov., vol. 8, no. 3, pp. 157-172, 2018.
- [5] S. Luthra and S. K. Mangla, "Evaluating challenges to Industry 4.0 initiatives for supply chain sustainability in emerging economies," Process Saf. Environ. Prot., vol. 117, pp. 168-179, 2018, doi: 10.1016/j.psep.2018.04.018.
- [6] S. ZHU, Dai-xian and Xin, MAO and LIU, "Logistics Vehicle Monitoring Intelligence System," no. Amma, pp. 127-131, 2017.
- [7] A. Celesti, A. Galletta, L. Carnevale, M. Fazio, A. Lay-Ekuakille, and M. Villari, "An IoT cloud system for traffic monitoring and vehicular accidents prevention based on mobile sensor data processing," IEEE Sens. J., vol. 18, no. 12, pp. 4795-4802, 2018, doi: 10.1109/JSEN.2017.2777786.
- [8] K. Guo, P. Yang, D. H. Guo, and Y. Liu, "Gas Leakage Monitoring with Mobile Wireless Sensor Networks," Procedia Comput. Sci., vol. 154, pp. 430-438, 2018, doi: 10.1016/j.procs.2019.06.061.
- [9] S. Abdelhamid, H. S. Hassanein, and G. Takahara, "Vehicle as a mobile sensor," Procedia Comput. Sci., vol. 34, pp. 286-295, 2014, doi: 10.1016/j.procs.2014.07.025.
- [10] Y. Y. F. Panduman, A. R. A. Besari, S. Sukaridhoto, R. P. N. Budiarti, R. W. Sudibyo, and F. Nobuo, "Implementation of integration VaaMSN and SEMAR for wide coverage air quality monitoring," Telkomnika (Telecommunication Comput. Electron. Control., vol. 16, no. 6, pp. 2630-2642, 2018, doi: 10.12928/TELKOMNIKA.v16i6.10152.

- [11] J. Guth et al., "A Detailed Analysis of IoT Platform Architectures: Concepts, Similarities, and Differences," pp. 81–101, 2018, doi: 10.1007/978-981-10-5861-5_4.
- [12] Y. Y. F. Panduman, M. R. Ulil Albaab, A. R. Anom Besari, S. Sukaridhoto, A. Tjahjono, and R. P. Nourma Budiarti, "Implementation of data abstraction layer using kafka on SEMAR platform for air quality monitoring," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 9, no. 5, pp. 1520–1527, 2019, doi: 10.18517/ijaseit.9.5.8547.
- [13] A. Mochamad Rifki Ulil, Fiannurdin, S. Sukaridhoto, A. Tjahjono, and D. K. Basuki, "The Vehicle as a Mobile Sensor Network base IoT and Big Data for Pothole Detection Caused by Flood Disaster," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 239, no. 1, 2019, doi: 10.1088/1755-1315/239/1/012034.
- [14] A. Rasyid et al., "Pothole Visual Detection using Machine Learning Method integrated with Internet of Thing Video Streaming Platform," *IES 2019 - Int. Electron. Symp. Role Techno-Intelligence Creat. an Open Energy Syst. Towar. Energy Democr. Proc.*, pp. 672–675, 2019, doi: 10.1109/ELECSYM.2019.8901626.
- [15] M. Kim, J. Lee, and J. Jeong, *Open Source Based Industrial IoT Platforms for Smart Factory: Concept, Comparison and Challenges*, vol. 11624 LNCS. Springer International Publishing, 2019.
- [16] J. Ren, H. Guo, C. Xu, and Y. Zhang, "Serving at the Edge: A Scalable IoT Architecture Based on Transparent Computing," *IEEE Netw.*, vol. 31, no. 5, pp. 96–105, 2017, doi: 10.1109/MNET.2017.1700030.
- [17] A. Salami, Amirfardad and Yari, "A framework for comparing quantitative and qualitative criteria of IoT platforms," *2018 4th Int. Conf. Web Res.*, pp. 34–39, 2018.
- [18] R. L. DUMITRU, "IoT Platforms: Analysis for Building Projects," *Inform. Econ.*, vol. 21, no. 2/2017, pp. 44–53, 2017, doi: 10.12948/issn14531305/21.2.2017.04.
- [19] Y. Y. F. Panduman, S. Sukaridhoto, and A. Tjahjono, "A Survey of IoT Platform Comparison for Building Cyber-Physical System Architecture," *2019 2nd Int. Semin. Res. Inf. Technol. Intell. Syst. ISRITI 2019*, pp. 238–243, 2019, doi: 10.1109/ISRITI48646.2019.9034650.
- [20] M. A. A. da Cruz, J. J. P. C. Rodrigues, P. Lorenz, P. Solic, J. Al-Muhtadi, and V. H. C. Albuquerque, "A proposal for bridging application layer protocols to HTTP on IoT solutions," *Futur. Gener. Comput. Syst.*, vol. 97, no. 2019, pp. 145–152, 2019, doi: 10.1016/j.future.2019.02.009.
- [21] R. A Light, "Mosquito: server and client implementation of the MQTT protocol," *J. Open Source Softw.*, vol. 2, no. 13, p. 265, 2017, doi: 10.21105/joss.00265.
- [22] T. team, "NATS - Open Source Messaging System | Secure, Native Cloud Application Development," *Nats.io*, 2020. <https://nats.io/>.
- [23] C. C. Chang and C. J. Lin, "LIBSVM: A Library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, 2011, doi: 10.1145/1961189.1961199.
- [24] S. Bashir, U. Qamar, F. H. Khan, and M. Y. Javed, "An Efficient Rule-Based Classification of Diabetes Using ID3, C4.5, & CART Ensembles," *Proc. - 12th Int. Conf. Front. Inf. Technol. FIT 2014*, pp. 226–231, 2015, doi: 10.1109/FIT.2014.50.
- [25] M. Schofield, "Meaningfully Judging Performance in Terms of User Experience)," *Weav. J. Libr. User Exp.*, vol. 1, p. 4, 2016.