Automatic Sign of Commencement of Work from Enterprise Resource Planning

1st Aryo Nugroho
Faculty of Computer Science
Narotama University
Surabaya, Indonesia
aryo.nugroho@narotama.ac.id

2nd Darian Rizaludin
Faculty of Computer Science
Narotama University
Surabaya, Indonesia
darian@rad.net.id

3rd Santirianingrum Soebandhi
Faculty of Computer Science
Narotama University
Surabaya, Indonesia
santirianingrum@narotama.ac.id

4th Lukman Junaedi
Faculty of Computer Science
Narotama University
Surabaya, Indonesia
lukman.junaedi@narotama.ac.id

5th Slamet Winardi
Faculty of Computer Science
Narotama University
Surabaya, Indonesia
slamet.winardi@narotama.ac.id

6th Moh Noor Al-Azam
Faculty of Computer Science
Narotama University
Surabaya, Indonesia
noor.azam@narotama.ac.id

***Abstract*— Enterprise Resource Planning (ERP) is a package of systems and software used by enterprises to manage their daily business activities, such as financial management, procurement, production, projects, human resources, and others. This system can facilitate business with real-time and accurate information so that the leaders can make business decisions well. The use of ERP in a manufacturing company usually stops only in the administrative department and does not reach the production machinery. In this paper, the researchers propose a way of interconnecting the ERP system with parts of the production machine to get more energy savings and speed up the time commencement of work. By using signal lamps that are connected to the ERP system and communicating through Message Queue Telemetry Transport (MQTT), the staff in the production machine area can immediately find out the status of the readiness of materials and production equipment in the inventory section.**

***Keywords*— ERP; IoT; MQTT; Industry 4.0**

I. Introduction

The industry is one part of the economic process that produces goods or provides services. In the industry - especially in manufacturing, production usually through highly mechanized and automated means. Ever since we knew industrialization, the change of technology has always triggered a paradigm shift that is now known as the "industrial revolution.". Our industry already went through three industrial revolutions, namely the first industrial revolution, when we use mechanization for the first time. The second industrial revolution is when we use electricity massively. And finally, the 3rd industrial revolution is when we use digital technology widely[1].

Around the 1960s, the idea emerged of using computers to record company activities. At that time, the concept of recording company activities was applied to inventory management and control in the manufacturing sector. Software engineers create programs to monitor inventory, reconcile balances, and report status. In the 1970s, this concept had developed into a Material Requirement Planning (MRP) system for scheduling production processes.

In the 1980s, the MRP expanded to include more manufacturing processes, prompting many to refer to it as MRP-II or Manufacturing Resource Planning. By the early 1990s, this system had evolved beyond the control of inventory and operational processes to other administrative functions such as accounting and human resources. That's when the term Enterprise Resource Planning (ERP) began as we know it today[2].

At present, the use of ERP in the company is nearing its zenith, because technology has far developed. And as we realize, every technological leap will trigger an industrial revolution. Now we are at the beginning of the 4th industrial revolution or better known as Industry 4.0.

II. Review of Literature

A. *Enterprise Resource Planning (ERP)*

The ERP terminology was first introduced by Gartner in 1990[2]. On its development, ERP presents a comprehensive database of corporation information and can integrate more than 50 or 60 different applications[3]. Such as manufacturing, project management, finance management, human resources management, and many more.

For the first time, ERP was an improvement or continuation of MRP-II. From the technological perspective, ERP aimed to organize the planning and implementation using electronics on a global basis in an enterprise. It's integrating a piece of individual information operating at each site or between participants in each chain supply. It pulls the different business fields collectively, presenting in a screen on every aspect of supply chain activities at a superficial and detailed level. Therefore, MRP-II cannot offer this capability.

It is essential to distinguish ERP from MRP-II in two main aspects. First, MRP-II operates mostly in the factory locally. MRP-II cannot offer the scope that ERP has, which goes beyond the local boundaries of factories or divisions to include groups and suppliers. Second, MRP-II principally is a reactive technology, while ERP is a proactive and real-time technology[3].

At this time, the use of ERP is no longer a particular thing in companies. Many companies have used the ERP system to support their business as a whole. Even the use of ERP systems is no longer the monopoly of large companies, middle and small-scale companies have also used it. That's because the ERP systems are no longer just proprietary software, ERP systems with open licenses have even begun to be widely available.

B. *Industry 4.0*

The 3rd industrial revolution or digitalization in the industry, has now continued its journey accompanied by Internet technology and future-oriented technology in the field of "smart" objects both in machinery and the products itself. This journey seems to have produced a new fundamental paradigm shift in industrial production.

The vision of production in the future will contain a modular and cost-effective production system. This makes it a scenario where the products will control their production processes. This process will realize an individual product manufacturing in size one, but at the same time maintain the economic conditions in mass production that have been done so far. Motivated by these ultimate expectations, the terminology "Industry 4.0" was established as a substitute for the terminology "planned 4th industrial revolution"[1].

In the transformation of management systems related to physical assets and computing capabilities, so currently, we are familiar with the Cyber-Physical System (CPS) terminology. The current technology development has resulted in the availability of sensors with higher accuracy that is price affordable. It also makes the data acquisition systems easier with communication networks that are available everywhere. This development has undoubtedly changed the competitive nature of the industry and forced more factories to move towards implementing high technology methodologies[4][5]. As a result, the growing use of sensors and machines connected to the network has resulted in the generation of data with very high volumes continuously, or now to be known as Big Data[6][7]. In such environmental conditions, CPS can be further developed to manage the Big Data and improve machine interconnectivity to achieve smart, resilient, and self-adapting machine goals[4].

Furthermore, integrating this CPS with production, logistics, and services in industrial practice will transform today's factories into Industrial 4.0 factories with significant economic potential. One example of CPS in this research is the integration of the existing ERP system with a piece of equipment that will inform the production machine room that the materials and equipment are ready, so the production staff must start preparing their production machines.

C. *RESTful Web Services*

REpresentational State Transfer (REST) was initially introduced as an architectural style for building large-scale distributed hypermedia systems. This architectural style is a rather abstract entity, whose principles have been used to explain the scalability of the excellent HTTP 1.0 protocol and also limit the design of the next version, HTTP 1.1[8].

The REST architecture is based on four principles[9]:

- Identification of resources through URI. This identification provides a global addressing space for resource discovery and flexible services.
- Resources are manipulated using HTTP methods: PUT, GET, POST, and DELETE.
- Resources are separated from their representations so that their content can be accessed in various formats (e.g., HTML, XML, plain text, PDF, JPEG, etc.).
- A stateful interaction through the hyperlinks, whereas interactions with a stateless source (e.g., request messages) are independent.

This study uses RESTfull Web Services to connect the existing ERP systems in the company with the automatic Commencement of Work (CoW) sign system. This connection bridge will use a Message Queue Telemetry Transport (MQTT) broker to forward messages from the ERP system to the signal hardware installed at the production site.

D. *Message Queue Telemetry Transport (MQTT)*

Machine-to-Machine (M2M) communication will always develop according to the development of the Internet of Things (IoT). Especially when entering Industry 4.0. At that time, more and more devices will need to communicate with other devices.

MQTT is a message protocol that is very simple and lightweight. Designed for use in devices that have limited resources, low bandwidth, high latency, and unreliable networks[10]. Publish is a function used by a device to send data or commands to another device. While the subscribe function is used to read data or commands, that sent by other devices that must be known.

To distinguish messages received from publishers to be forward to which subscribers, the MQTT system uses the topic for a message. MQTT topics are arranged in a multilevel arrangement similar to directories and files in the file system that we are familiar with, using slashes (/) as the boundary. By using this topic naming system, a user-friendly and descriptive naming structure can be made according to user choice.

In this study, MQTT is used as a protocol by the ERP system to send messages to the Commencement of Work Sign hardware located in the production department.

E. *Pulse Width Modulation (PWM)*

Pulse Width Modulation (PWM) is a method for reducing the average power in an electrical signal, by effectively cutting it into separate signal parts. The average value of the voltage applied to the load is controlled by turning the switch off and on quickly. The longer the switch in on compared to the off period, the higher the total power supplied to the load[11].

In this study, the PWM signal generated from a micro-controller will be used to control the light intensity of the Light Emitting Diode (LED). As we know, Red-Green-Blue (RGB) light is the basis of the color of the rays we see. By adjusting the duty cycle of each RGB light from the LED, we will get a color that will be used as a signal to prepare a job in the production room.

III. ANALYSIS AND DISCUSSION

In this study, there is a global system -Automatic Commencement of Work (CoW), which has two sub-system parts that must be tested. In general, the research stages are carried out, as shown in Figure 1.

Existing ERP will be accessing the Automatic CoW system through a RESTful Web Services sub-system. This sub-system then translates the ERP command into a specific message for the particular production machine. Then the message sent through the MQTT broker to a specific CoW Sign sub-system -that is located in the production room. This sign will determine when the staff must start the job on a production machine.

A. *Literature Studies*

In this phase, a reference search is done for problems that will be expected to be encountered. The main problem is the integration of the existing ERP system into an IoT device, which is located quite far away. The goal at this stage is to obtain hypothesis support in designing the whole system.

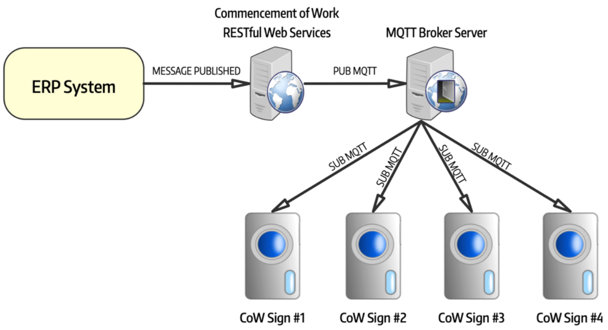

Fig. 1. Block Diagram of Commencement of Work Sign

B. Design of Commencement of Work LED Sign

After getting some references from the existing pieces of literature, in the second phase, proceed with making the design of the Globally Automatic WoC System. The main challenge of this design is how ERP systems located on one of the servers in the server room can provide information to WoC Sign, which is in the production room easily (figure 2).

The design proposed in this research is to make RESTful Web Services accessed by the existing ERP system at certain stages. RESTful Web Services will then instruct the CoW Sign to display certain colors through an MQTT broker.

C. RESTful API

To make it easier to make the interconnection of an existing system without having to shut down the system is to create a bridge between the old system and the addition of the new one. With this technique, the old system will continue to run as usual, but if there is information that needs to be sent to the new system, the old system can pass it over the bridge[9][8].

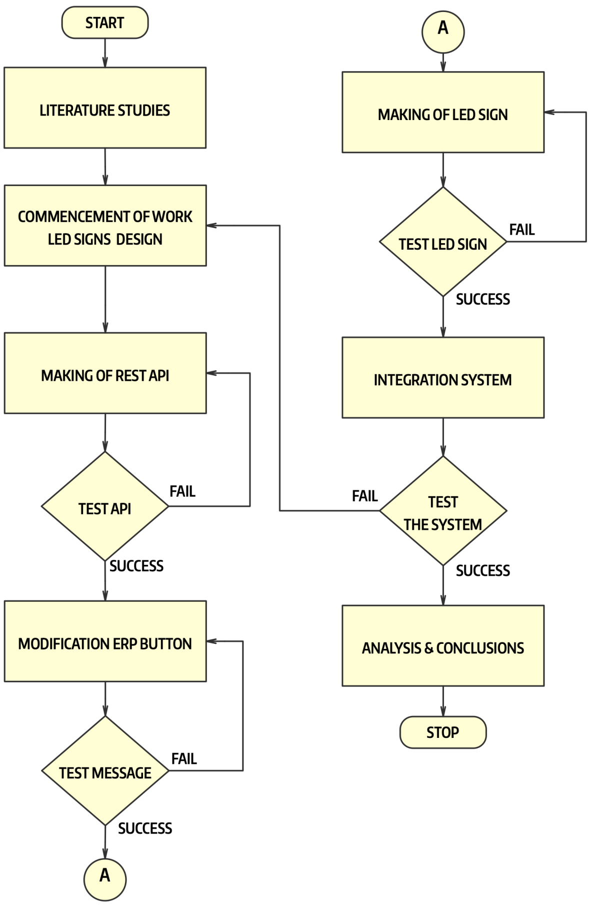

Fig. 2. The Research Flowchart

In the design of WoC Sign in general, this study will not change the existing ERP system used. This research creates a new WoC Sign system outside the current ERP system that is

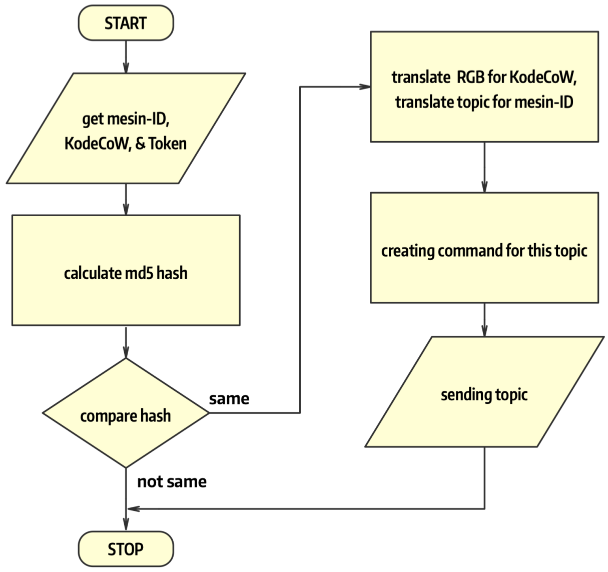

Fig. 3. RESTful API Flowchart

currently running, the information of which is obtained from the existing ERP system. Therefore, to bridge the two systems, this research creates a bridge in the form of RESTful API Web Services.

This RESTful Web Services API is created with the HTTP GET method to facilitate modification in the existing ERP system. At certain points in the ERP system, which need to change the condition of the WoC Sign lights in the production room, a syntax is added to send the message. The URL access syntax is: http://ip.address.api/index.php? Engine=ID&cow=CODE&token=HASH.

The engine variable is the identification of the CoW Sign that is on a specific number machine. The CoW code variable is the color code that will be sent to the CoW Sign, and finally, the token is the md5 result of these commands for security verification.

The RESTful API Web Services flowchart can be seen in Figure 3.

D. MQTT

MQTT -stands for Message Queuing Telemetry Transport, a lightweight publishes/subscribe system where we can publish and receive messages as a client. MQTT is a simple messaging protocol designed for Internet of Things (IoT) devices, which have limited resources and with low bandwidth[10].

In this study, MQTT is used as a transport protocol for sending commands from the RESTful API to the WoC Sign. MQTT broker server used in this study is VerneMQ, which is compatible with the OASIS MQTT 5.0 standard[12].

The MQTT standard settings used in this study are as follows:

- Topic: a WoC Sign will subscribe to MQTT brokers for in specific topic, and the topic format used is <company name>/woc/<engine number>. Engine_number is the number of the machine to be signaled.
- QoS: Quality of Service used in MQTT is QoS 0 - since the communication network used is a reliable internal network.

E. CoW Sign

This Commencement of Work Sign uses ESP-32 as its control center. ESP-32 is a low-cost and low-power microcontroller chip on a microcontroller chip. This chip has integrated WiFi and Bluetooth modules in dual-mode [13].

The ESP-32 series uses a microprocessor from Tensilica Xtensa LX6 in dual-core and single-core variations. ESP-32 was made and developed by Shanghai-based Espressif Systems, and this is the successor to the ESP-8266 microcontroller, better known as Wemos.

In this study, we utilize some of the features contained in ESP-32, including:

- 3 digital pins-out for the LED indication of CoW running systems, established WiFi connections, and MQTT broker subscribed.
- 1 digital pin-in for identifying CoW systems in setup mode or running mode.
- Communication with MQTT brokers will use the WiFi using its WiFi module.
- 3 PWM pins to make multiple colors of RGB LED as a specific machine's work status code.

The program that runs in CoW Sign is following the flowchart in figure 4.

When the CoW Sign is first turned on, 3 GPIO (General Purpose Input / Output) ports are set up as output, used as an LED indicator to operate, WiFi Connect, and Setup Mode. It also requires a GPIO pin as input to indicate Setup Mode (if the pin is in logic 0) or in Operation Mode (if the pin is in logic 1).

When the pin setup mode is active (logic 0), the WoC Sign is in Setup Mode, and the program will go to the setup procedure. This procedure task is to communicate with the user via UART (Universal Asynchronous Receiver-Transmitter). The function of this procedure is so that the user can fill in the WiFi SSID used, determine the Broker's MQTT IP Address and port, and this Topic WoC Sign will subscribe.

The setup procedure will end with saving all settings made by the user, and the user must reboot WoC Sign again.

If the pin setup mode is inactive (logic 1), the WoC Sign is in running mode. In this mode, the program will first read the settings, connect to the WiFi following the SSID and WPA key that was entered previously. If the connection is successful, then WoC Sign will subscribe to the MQTT broker according to the predetermined topic. In this condition, WoC Sign is ready to take orders from ERP to give a signal by making the LED light up with specific colors.

Four message variables formatted in JSON sent by ERP via API WebServices and MQTT. Three variables are numeric

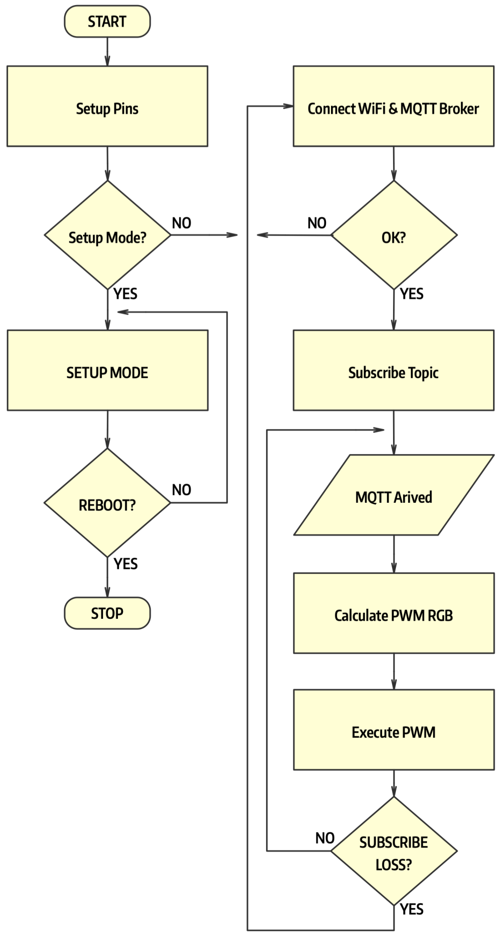

Fig. 4. CoW Sign Flowchart

codes of the intensity of the colors red, green, blue, and one variable check is an alphanumeric containing the hashing of the codes (figure 5).

The variable red, green, and blue light intensity codes are numeric from 0 to 255 (1 byte), which will be used later be adjusted in the PWM ESP-32 duty cycle. The numeric number of 0 means a duty cycle of 0% or the LED is off, while the numeric number of 255 means a duty cycle of 100% or an LED intensity of 100%.

As is known, we can play the intensity of red, green, and

```
{
 “red” : 255,
 “green” : 255,
 “blue” : 255,
 “check” : “37937c72cda976880d965f9c9b1ed405c2d94200”
}
```

Fig. 5. JSON Format WoC Sign

blue colors to get a lot of colors. However, to facilitate the production staff in translating, only 5 states are taken, as shown in table 1.

The encoding of the production state on this RGB LED is done on the ERP system, so that message sent by ERP through API WebServices and MQTT broker is already in ready-to-use condition. This is to anticipate the addition of statuses that might occur in the future. Adding these statuses can be directly done in ERP without having to modify the CoW Sign fundamentally.

The check variable is a variable for control to ensure that the sender of the message is the correct ERP system and not from another system. This check takes the form of the SHA1 hash of all variables sent, plus additional information that only the ERP system and WoC Sign know about.

TABLE I. LED & STATUS PRODUCTION

PRODUCTION STATE	LED		
	RED	***GREEN***	***BLUE***
No Order – Machine Off	***0***	***0***	***0***
Has Order – Material Not Ready	***255***	***0***	***0***
Has Order – Material 50%	***0***	***255***	***255***
Has Order – Material 100%	***0***	***255***	***0***
Machine on Operating	***255***	***255***	***255***

Therefore, only the results of the same SHA1 hash will be processed by the WoC Sign, while the results of the same check will be reported in a warning log to the administrator.

F. Integration the System

Experiments are carried out in stages by starting with the creation of a Web Services API. This sub-system must be able to translate according to the URI, which will be accessed by the existing ERP system following the information on the orders and materials available in the inventory.

The second attempt was to test the interconnection between Web Services APIs and MQTT. In this experiment, the community MQTT broker was used, which was provided for the trial. And because WoC Sign as a subscriber is still under development, checks are made for this interconnection using the MQTT dashboard, which is widely available on Google Play. The target of this stage is that the information sent by accessing the Web Services URI API must match the data received by the MQTT dashboard through the MQTT broker.

To conduct a trial at this stage, we do it by creating a small program using the python programming language, accessing the Web Services API with random data. As a comparison, using the python language also subscribe to MQTT with an agreed topic.

The tester program will access the Web Services API 1 access per-second for 2 different topics. The comparison program will receive the messages from the MQTT broker sent by the tester program, and then we will compare the data transmitted and collected.

If the Web Services API sub-system works well, research will continue to modify the existing ERP system.

Modifications start from making a simple push button to send commands via the Web Services API. The initial target of this integration proves that the existing ERP system can send commands to the WoC Sign in the production room.

After interconnection from the existing ERP system until the MQTT broker can work adequately, the Commencement of Work (CoW) Sign hardware begins.

As planned, this CoW Sign will use NodeMCU ESP-32. This research will also utilize WiFi modules that are already available in the System on Chip (SoC). To be able to provide voltage to the LED, this CoW Sign uses an NPN transistor driver (figure 6).

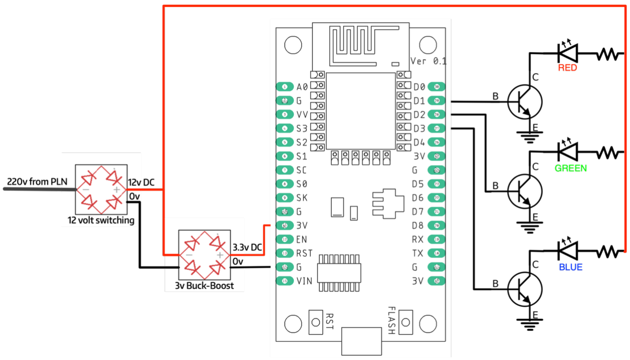

Fig. 6. Scheme Diagram CoW Sign

The bases of these transistors will be connected to PWM-1 (D1) for the red LED data, PWM-2 (D2) for the green LED data, and PWM-3 (D3) for the blue LED data. By using a transistor as this switch, PWM can be performed on LEDs with a 12-volt voltage via a microcontroller with an operational voltage of 3.3 volts.

IV. TESTING SYSTEM

In the second phase of the experiment, we used an ERP system simulation program that accesses the Web Services API to send 1 random message per second for 1 hour. Also, on the subscriber side, a program subscribes to MQTT on a particular topic. The results of the subscriber will be stored in a text-file.

By comparing the messages sent and received, the reliability of the system will be known. The results of this test can be seen in table 2.

TABLE III. BUTTON ERP SYSTEM TEST

Machine #	Button Test				
	#1	***#2***	***#3***	***#4***	***#5***
Machine #1	OK	OK	OK	OK	OK
Machine #2	OK	OK	OK	OK	OK

In the overall sub-system integration experiment, the ERP system that has been modified by giving a button to indicate the readiness of the production material also works well (table 3).

TABLE II. TESTING API & MQTT

Topic	Messages		
	Sent	***Received***	***Success***
narotama/cow/1	3.600	3.600	100%
narotama/cow/2	3.600	3.600	100%

V. CONCLUSION AND FUTURE WORKS

Although it has not been done thoroughly and automatically, this research proves that the industrial era 4.0 has begun. The integration of ERP with cloud, IoT, and production machines is not a strange thing anymore. It is only a matter of time before the implementation of Industry 4.0 penetrates the entire production department.

There is still a lot that needs to be further researched and tested on this system, including:

- System security needs to be improved. The message sent by the ERP system must be confirmed that it was sent by the ERP system and not by others
- Automation of the system also needs to be improved. This experiment only proves that the concept of industry 4.0 can be implemented, but the automatic integration with decisions on ERP data has not been carried out in total. Because of that, there is a need for other research that can help the decision when the production machine should run and at what machine number at the specific production request.
- Furthermore, the 4.0 industrial revolution will touch machines directly. Therefore there needs to be other research so that the ERP system can instantly run a machine and report the condition of the engine to the ERP system.

REFERENCES

[1] H. Lasi, P. Fettke, H.-G. Kemper, T. Feld, and M. Hoffmann, “Industry 4.0,” *Bus. Inf. Syst. Eng.*, vol. 6, no. 4, pp. 239–242, Aug. 2014.

[2] L. Hossain, J. D. Patrick, and M. A. Rashid, Eds., *Enterprise Resource Planning: Global Opportunities and Challenges*. IGI Global, 2002.

[3] S. Philpott, “ERP: integrating enterprise-wide planning functions for the aerospace and defence industry,” *Aircr. Eng. Aerosp. Technol.*, vol. 67, no. 4, pp. 16–18, Apr. 1995.

[4] J. Lee, B. Bagheri, and H.-A. Kao, “A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems,” *Manuf. Lett.*, vol. 3, pp. 18–23, Jan. 2015.

[5] N. Jazdi, “Cyber physical systems in the context of Industry 4.0,” in *2014 IEEE International Conference on Automation, Quality and Testing, Robotics*, Cluj-Napoca, Romania, 2014, pp. 1–4.

[6] J. Lee, E. Lapira, B. Bagheri, and H. Kao, “Recent advances and trends in predictive manufacturing systems in big data environment,” *Manuf. Lett.*, vol. 1, no. 1, pp. 38–41, Oct. 2013.

[7] J. Shi, J. Wan, H. Yan, and H. Suo, “A survey of Cyber-Physical Systems,” in *2011 International Conference on Wireless Communications and Signal Processing (WCSP)*, Nanjing, China, 2011, pp. 1–6.

[8] L. Richardson and S. Ruby, *RESTful web services*. Farnham: O’Reilly, 2007.

[9] C. Pautasso, O. Zimmermann, and F. Leymann, “Restful web services vs. ‘big’’ web services: making the right architectural decision,” in *Proceeding of the 17th international conference on World Wide Web - WWW ’08*, Beijing, China, 2008, p. 805.

[10] M. N. Al-Azam, D. Rizaludin, Y. S. Raharjo, and A. Nugroho, “Message Queuing Telemetry Transport dalam Internet of Things menggunakan ESP-32,” *J. MEDIA Inform. BUDIDARMA*, vol. 3, no. 3, p. 159, Jul. 2019.

[11] B. Anindito, A. G. Sooai, M. M. Achlaq, M. N. Al-Azam, A. Winaya, and M. Maftuchah, “Indoor Agriculture: Measurement of The Intensity of LED for Optimum Photosynthetic Recovery,” in *2018 5th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, Malang, Indonesia, 2018, pp. 356–361.

[12] Andrew Banks, Ed Briggs, Ken Borgendale, and Rahul Gupta, Eds., “MQTT Version 5.0.” OASIS Standard, 07-Mar-2019.

[13] M. N. Al Azam, I. Arif, D. Rizaludin, U. Ayuba, A. Haris, and A. Nugroho, “Wireless Sensor Network using Wireless Fidelity for Environmental Monitoring System,” in *2019 International Seminar on Application for Technology of Information and Communication (iSemantic)*, Semarang, Indonesia, 2019, pp. 1–7.