ISSN No:-2456-2165

Monitoring and Controlling System of Orchid Greenhouse Parameter Using Android and Web for Von Florist in Semarang Regency

[1]Sindung Hadwi Widi Sasono*
Polytechnic State of Semarang, Indonesia

[2]Helmy Iskandar, Herdiansyah Yusuf Irham, Sri Kusumastuti
Polytechnic State of Semarang, Indonesia

Abstract:- Indonesia is a tropical country that has an abundance of decorative plants; the orchid is one of them. Orchids are commonly grown in Indonesia due to their charming color. A special treatment is required to grow an orchid properly. Among them are high air humidity, low air temperature, and a particular amount of sunlight. If these parameters aren't fulfilled, the orchid's growth may be interrupted and even wilt. To solve this issue, a system that monitors the air temperature, air humidity, light intensity, and soil moisture and controls the air temperature and humidity of the greenhouse is needed. And to make the monitoring and controlling process easier, Android and web-based applications have been made. Both applications are tested using black-box testing to ensure that they are easy to use. A performant and quick application can also impact the usability of an application. To know whether the produced applications fit those definitions, a performance test and control delay measurement were conducted. For the latter, the measurement process was conducted in the Polytechnic State of Semarang, representing a place with fast and stable network speed, and the orchid greenhouse in Sumowono, Semarang regency, as the opposite and onsite location. From the black-box testing results, both applications run as designed. The average performance of the Android application is fairly light, while the web application has a decently long loading time of 2680 ms on one of the pages. For the controlling delay measurement, the average delay in the Polytechnic State of Semarang and the orchid greenhouse in Sumowono, Semarang regency, was 343 ms and 6641 ms, respectively According to the ETSI standard, the former delay is considered fair while the latter is bad.

Keywords:- *IoT, Greenhouse, Orchid, Android App, Web App*

I. INTRODUCTION

Indonesia is a tropical country rich in ornamental plants. One of the commonly produced ornamental plants is the orchid. Orchids belong to the *Orchidaceae* family (orchid family). Orchid flowers are one of the most attractive parts because of their colors. The color is caused by the dye contained in the flower's plastids. Orchids blooms at different time depending on the species. Generally, mature orchids bloom after 1-2 months since planted. Usually, 2 flower stalks are produced with around 20-25 buds per stalk. Orchids that are mature and 2 months old will produce 2 stalks with 20-25 buds per stalks.

Indonesia is able to produce orchids in large quantities. In 2020, the total production of orchids in Indonesia can reach up to 11.6 million stalks. However, this production amount was lower than the in the previous year, which was around 18.6 million stalks [1]. One of the orchid cultivators in Indonesia is Von Florist. Von Florist is one of the UMKM (*Usaha Mikro, Kecil dan Menengah*) engaged in the field of orchid floriculture, especially *Phalaenopsis amabilis* (Moon orchid) and dendrobium orchids. These orchids come from PT. Ekakarya Graha Flora, the largest orchid producer in Indonesia. Von Florist has a greenhouse for orchid flowering located on Kemawi Sumowo street, Sumuwono sub-district, Semarang district, Central Java province. The greenhouse is 8x20 meters in size and has a production capacity of 2,000 orchids per month.

Special handling is required in growing orchids. In order for orchids to grow, high air humidity (±85%), low air temperature (27-29°C) and appropriate sun intensity are required [2]. If these parameters are not met, the growth of orchid plants can be hampered and even cause the orchid to wilt and die.

To deal with this problem, a greenhouse parameter monitoring and control system was made for orchids. In order to make the monitoring and control process easier to do, an Android and web-based system was created. In addition to being easier, the Android and web-based system allow users to monitor and control the greenhouse parameters of orchid plants from anywhere and anytime.

II. RESEARCH METHODS

A. Overall Research Methods

The research methods contain the steps taken to develop a monitoring and controlling system for an orchid greenhouse based on Android and web. Fig 1 is the architecture of the entire orchid greenhouse monitoring and control system, from the node devices in the orchid greenhouse, the Internet of Things (IoT) panel, to the server that stores and calculates all of the greenhouse parameters.

ISSN No:-2456-2165

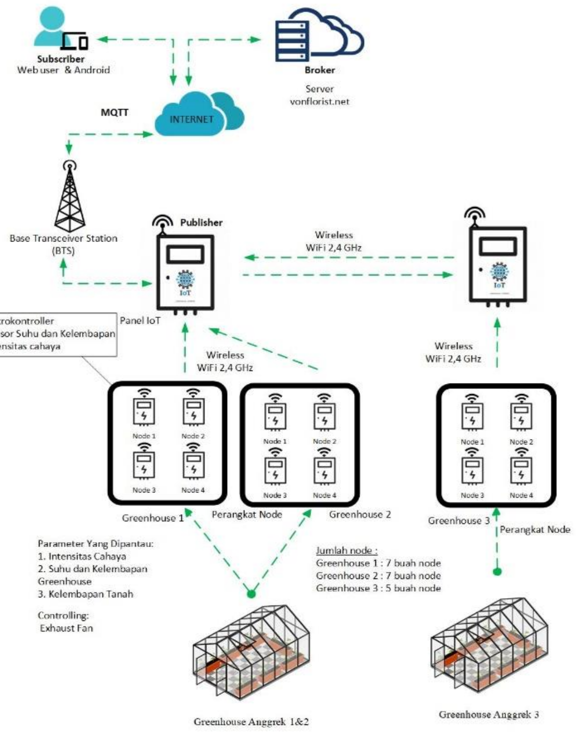

Fig 1 Architecture of the Orchid Greenhouse Parameter Monitoring and Control System

The Android and web-based orchid greenhouse parameter monitoring and control system will communicate with the server using the HTTP protocol with the help of REST API and MQTT. The REST API will be used to monitor greenhouse parameters, namely air temperature, air humidity, soil moisture, and light intensity. The REST API is also used for the user authentication process. The MQTT protocol is used for the greenhouse parameter control process, namely air humidity by controlling the exhaust fan based on the current air temperature in the greenhouse. The control process is carried out by sending the upper and lower thresholds of the desired greenhouse. The data that will be retrieved via the REST API is obtained from the database on the server. The data on the server is received from the IoT panel via MQTT. The data obtained is then processed and stored in the MySQL database.

B. Android and Web App Design

Android and Web application design contains the steps taken so that both apps run as intended. Fig 2 is a use-case diagram that shows the Admin and Staff actors on the left and right as users and their roles in the middle.

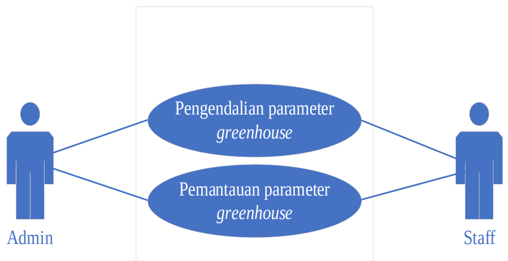

Fig 2 Use-Case Diagram of Orchid Greenhouse Parameter Control and Monitoring System

In addition to determining the role of the user, the application scenario is also determined which is divided into several subsystems, as shown in Table 1.

Table 1 Use-Case Subsystem of Android and Web Application

Subsystem	***User/actor***	**Description**
Login	Admin and Staff	Admin and Staff can enter their username and password to make sure that the user is indeed the Admin or Staff.
Summary monitoring	Admin and Staff	Admin and staff can monitor the air temperature, air humidity, soil moisture, and light intensity of the greenhouse. They can also monitor the previously mentioned parameters from a chart on that day or different time range.
Node monitoring	Admin and Staff	Admin and staff can monitor the greenhouse parameters based on the location of the node in the orchid greenhouse.
Parameter controlling	Admin and Staff	Admin and staff can control the orchid greenhouse parameters.

To store the orchid greenhouse parameter data and users' login credentials, a database is used. is the Entity Relationship Diagram (ERD) that shows the relationship between each table. In the database, 4 tables are used, "parameter_greenhouse", "summary_greenhouse", "user_notification", and "user".

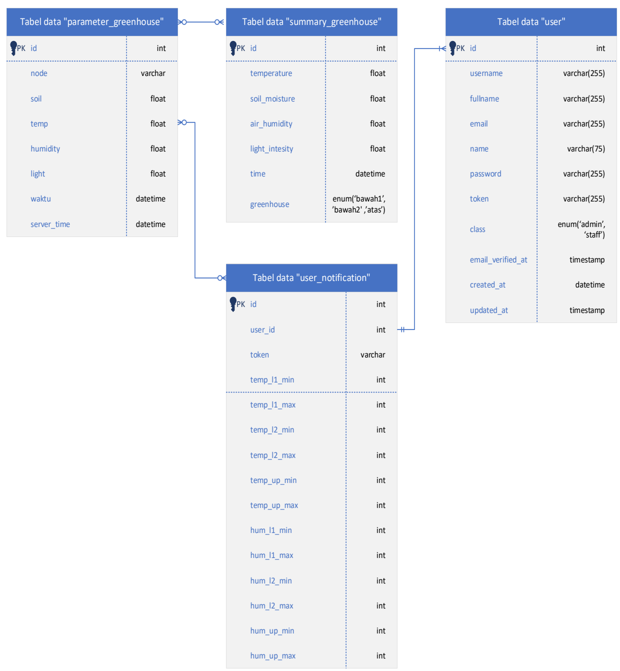

Fig 3 ERD Diagram of The Orchid Greenhouse Parameter Monitoring and Control Syste

ISSN No:-2456-2165

Fig 4 is the login page interface design in Android. The login process requires a username and password or QR code from the IoT panel.

Fig 4 Login Page Interface Design on Android Applications

Fig 5 is the summary page interface design in Android. This page shows the latest greenhouse parameters which value is calculated by averaging the greenhouse parameters from all nodes.

Fig 5 Summary Page Interface Design on Android Applications

Fig 6 shows the node page interface design in Android. This page shows the latest greenhouse parameter from each individual node device in the orchid greenhouse.

Fig 6 Node Page Interface Design on Android Applications

Fig 7 shows the threshold control page in Android. This page is used to modify the upper threshold and lower threshold of the air temperature to control the exhaust fan in the orchid greenhouse and the notification threshold for the air temperature and air humidity of the orchid greenhouse.

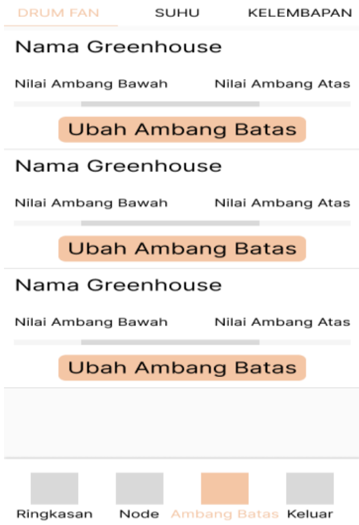

Fig 7 Threshold Page Interface Design on Android Applications

Fig 8 is the login page interface design in web. The login process requires a username and password.

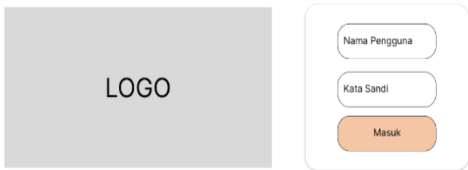

Fig 8 Login Page Interface Design on Web Applications

Fig 9 is the summary page interface design in Android. This page shows the latest greenhouse parameters which value is calculated by averaging the greenhouse parameters from all nodes.

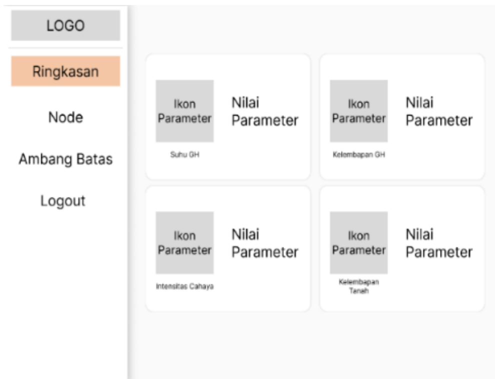

Fig 9 Summary Page Interface Design on Web Applications

Fig 10 shows the node page interface design in Android. This page shows the latest greenhouse parameter from each individual node device in the greenhouse.

ISSN No:-2456-2165

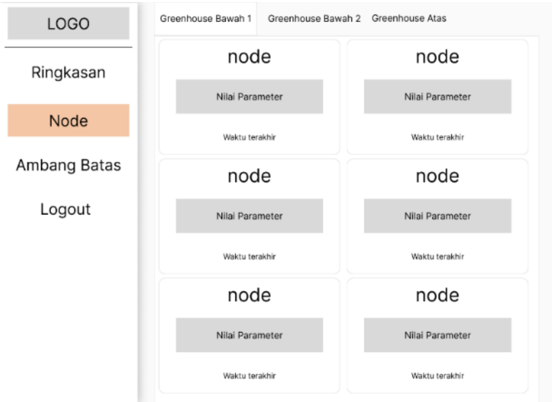

Fig 10 Node Page Interface Design on Web Applications

Fig 11 shows the threshold control page in Android. This page is used to modify the upper threshold and lower threshold of the air temperature in the greenhouse and the notification threshold for air temperature air humidity.

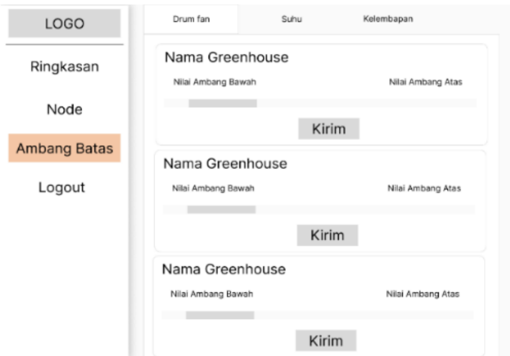

Fig 11 Threshold Page Interface Design on Web Applications

C. *Test Design*

The testing phase is carried out to ensure that the resulting system can work properly. Three tests were carried out, namely application functionality testing, application performance testing, and application control delay testing.

- Application functionality test (black-box testing) is conducted to test whether the features of the system work as expected.
- Application performance test is used to measure the memory, CPU, and energy usage for the Android app. For the web app, the page load time is measured.
- Control delay test is conducted to measure the time it takes to modify the thresholds in the IoT panel since the thresholds were modified from the app. This delay time is obtained by calculating the difference between the time when the threshold is updated via the application and when thresholds are displayed on the IoT panel. Equation 1 shows the formula used to measure the delay.

$$delay = (Tr\text{-}Tx) \text{ second} \quad (1)$$

The control delay test was conducted using Telkomsel as the Internet provider and done in 2 places, namely, the orchid greenhouse in Kemawi Street, Sumowono district, Semarang regency, Central Java, and Polytechnic State Semarang in Prof. Sudarto Street, Tembalang regency, Semarang City, Central Java. The control delay test is conducted in two places to compare the delay on-site, represented by the orchid greenhouse in Sumowono district, and in ideal condition, represented by Polytechnic State Semarang.

III. RESULT AND DISCUSSION

❖ *Android and Web App User Interface Design*

This step shows the result of the interface design for the Android app. **Error! Reference source not found.** shows the design result of the login page in Android, used to authenticate the user.

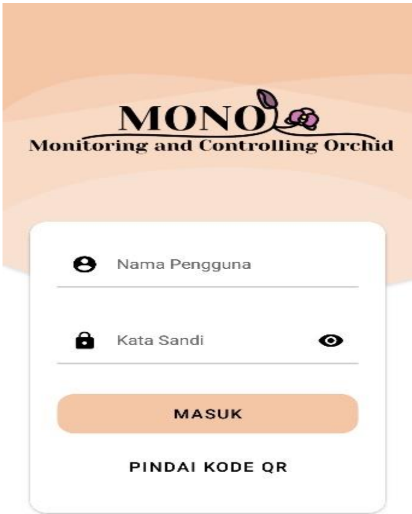

Fig 12 Design Result of Login Page in Android Application

Fig 13 is the design result of the summary page on the Android application. This page shows the latest greenhouse parameters which value is calculated by averaging the greenhouse parameters from all nodes.

Fig 13 Design Result of Summary Page In Android Application

Fig 14 is the design result of the node page on the Android application. This page shows the latest greenhouse parameter from each individual node device in the greenhouse.

Fig 14 Design Result of Node Page In Android Application

Fig 15 is the design result of the threshold control page on the Android application. This page is used to modify the upper threshold and lower threshold of the air temperature in the greenhouse and the notification threshold for air temperature air humidity.

Fig 15 Design Result of Node Page In Android Application

Fig 16 is the design result of the login page in web. The login process requires a username and password.

Fig 16 Design Result of Login Page In Web Application

Fig 17 is the design result of the summary page in web. This page shows the latest greenhouse parameters which value is calculated by averaging the greenhouse parameters from all nodes.

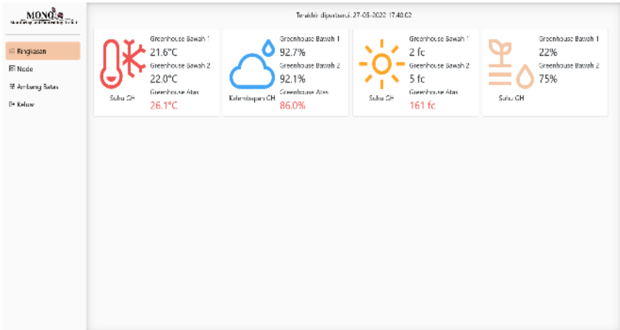
Fig 17 Design Result of Summary Page in Web Application

Fig 18 is the design result of the node page in web. This page shows the latest greenhouse parameter from each individual node device in the greenhouse.

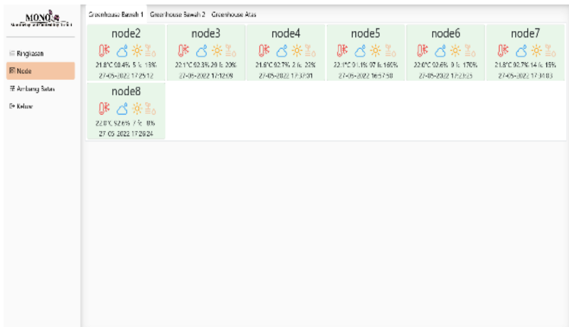

Fig 18 Design Result of Node Page in Web Application

Fig 19 is the design result of the threshold page in web. This page is used to modify the upper threshold and lower threshold of the air temperature in the greenhouse and the notification threshold for air temperature air humidity.

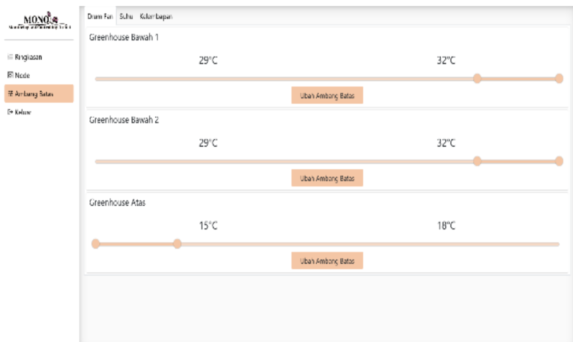
Fig 19 Design Result of Threshold Page in Web Application

A. Test Result

➢ *Functionality Test Result (Black-Box Testing)*

The functionality test (black-box testing) is conducted to test whether the features of the system work as expected. ality test for the Android app

Table 2 shows the result of the functionality test for the Android app

ISSN No:-2456-2165

Table 2 Functionality Test Results on Android Application

No.	Test	Expected Outcome	Status
1.	*Intro: Splash*	Shows the app icon for 1.5 seconds.	Success
2.	User Authentication: Login	User can login using the already registered username and password or QR code on the IoT panel.	Success
3.	Summary menu	Shows the summary for each greenhouse parameters, which value is calculated by averaging the greenhouse parameters from all nodes, on all greenhouses along with the last update time.	Success
4.	Chart menu	Shows the chart that compares the greenhouse parameter against a specified time, e.g., days or hours.	Success
5.	Node menu	Shows the greenhouse parameters from each individual node in the orchid greenhouse.	Success
6.	Threshold menu	Shows the thresholds control to modify the greenhouse thresholds, and notification thresholds for the air temperature and humidity of the greenhouse.	Success
7.	Notification	If the user has set the notification thresholds for either the air temperature or humidity of the greenhouse, the user will receive a notification containing the out of thresholds parameter values.	Success

The functionality test is also conducted on the web application. Table 3 shows the result of the functionality test for the web application.

Table 3 Functionality Test Results on Web Application

No.	Test	Expected Outcome	Status
1.	User Authentication: Login	User can login using the registered username and password or QR code on the IoT panel.	Success
2.	Summary menu	Shows the summary for each greenhouse parameters, which value is calculated by averaging the greenhouse parameters from all nodes, on all greenhouses along with the last update time.	Success
3.	Node menu	Shows the greenhouse parameters from each individual node in the orchid greenhouse.	Success
4.	Threshold menu	Shows the thresholds control to modify the greenhouse thresholds, and notification thresholds for the air temperature and humidity of the greenhouse.	Success
5.	Notification	If the user has set the notification thresholds for either the air temperature or humidity of the greenhouse, the user will receive a notification containing the out of thresholds parameter values.	Success

- *App Performance Test Result*

The app performance test for the Android app is conducted using the "Android Profiler" and "Network Profiler" in Android Studio.

Fig 20 shows the CPU, memory, and energy usage of the Android app.

Fig 20 Profiling Result on Android Application

The peak CPU usage of the Android app is around 52% when loading the login page. This high CPU usage is caused by the login process that requires the device to create a request to the server. The highest memory usage for the Android app is at 207.5 MB when loading the chart page. This is because when loading the chart page, specifically when picking the date range using the date picker, the date picker is displayed on top of the chart instead of temporarily hiding the chart and displaying date picker. The highest energy usage for the Android app happened while logging in, loading the main page for the first time, and when the chart page is first loaded. Fig 21 shows the network activity of the Android app.

ISSN No:-2456-2165

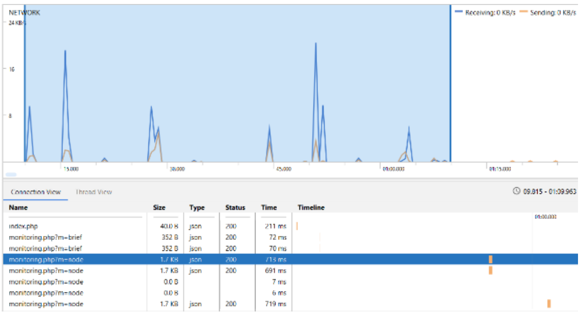

Fig 21 Network Profiling Result On Android Application

The slowest request to complete is when retrieving the node data in the node page, which took 719 ms. This is because the JSON response's size is relatively big, at around 1.7 KB.

The performance test for the web app is conducted using the "Network Monitor" in Google Chrome.

Table 4 shows the result of the "Network Profiler".

Table 4 Network Monitoring Result on Web Application

No.	Page	Load Time (ms)		
		DOM Content Loaded	Load	Entire Page
1.	*Login*	151	211	305
2.	Summary	1100	1320	2150
3.	Node	700	700	1250
4.	Threshold	399	503	2680

The slowest page to load in the web app is the threshold page. This is because when loading that page, the greenhouse thresholds have to be retrieved which took around 2180 ms.

➢ *Control Delay Measurement Test Result*

Control delay test is conducted to measure the time it takes to modify the thresholds in the IoT panel in the orchid greenhouse since the thresholds were modified from the app. Table 5 shows the result of the control delay measurement test conducted in Semarang State Polytechnic which represents the ideal location.

Table 5 Control Delay Test Results at Semarang State Polytechnic

No.	Time	Delay (ms)
1	14:12	3875
2	14:16	469
3	14:18	593
4	14:20	261
5	14:22	64
6	14:29	67
7	14:30	48
8	14:31	80
9	14:32	66
10	14:33	68
11	14:34	92
12	14:35	72
13	14:37	127
14	14:38	175
15	14:39	225
16	14:40	73
17	14:41	157
18	14:42	167
19	14:43	83
20	14:45	104
Average		343

From the test results, the average controlling delay is around 343 ms. Based on the ETSI standard this is categorized as average. Table 6 shows the controlling delay results in the orchid greenhouse in Kemawi Street, Sumowono district, Semarang regency, Central Java that represents the on-site condition.

Table 6 Control Delay Test Results at The Orchid Greenhouse

No.	Time	Delay (ms)
1	13:19	1300
2	13:20	1370
3	13:23	10950
4	13:28	2240
5	13:29	2010
6	13:38	12170
7	13:43	2270
8	13:48	4900
9	13:50	15870
10	13:53	1770
11	14:03	13370
12	14:23	17740
13	14:27	4380
14	14:29	3050
15	14:31	1380
16	14:32	1730
17	14:33	2270
18	14:34	7760
19	14:36	18140
20	14:48	8150
Average		6641

From the conducted test results, the average controlling delay is around 6641 ms, which is considered to be bad in based on the ETSI standard.

IV. CONCLUSION

From the results of the conducted tests, the following conclusions are drawn:

- The monitoring and controlling system of orchid greenhouse parameter using Android and web for Von Florist in Semarang regency can monitor the parameters of an orchid greenhouse, namely air temperature, air humidity, light intensity, and soil moisture, and control the air temperature and humidity via exhaust fans.
- From the functionality test result, the Android and web app run as designed thus capable of logging in the user, display the summary menu, node menu, threshold

ISSN No:-2456-2165

control menu, and notification. For the Android app, the splash screen and chart menu are displayed properly.

- From the Android app's performance chart, the CPU usage of the Android app reaches 52%, while the memory usage can reach 207.5 MB, and the slowest load time happened while loading the node data at around 719 ms. For the web app, the slowest page to load is the threshold control menu which took 2680 ms to load the entire page.
- The average controlling delay of the app while inside the orchid greenhouse, that represents the on-site condition, is 6641 ms which is considered poor according to the ETSI standard. While inside the Polytechnic State Semarang, that represents the ideal location, the average controlling delay is 343 ms which is considered decent to the ETSI standard.

SUGGESTION

From the conducted system optimization, there are some suggestions that are expected to be fulfilled in further development, such as:

- The summary and node data for both Android and web apps use REST API to take the data from the database to the app. With the use of cache, the data retrieval process time can be much shorter.
- Using a web framework, e.g., Laravel, to help hasten the development and page load time.
- Rent a better internet service to reduce the delay control time while on-site.
- Develop the web app using a software patter, such as MVC (Model–View–Controller), to help with future code maintenance.
- For both Android and web app, add a page that contains information about the harvest.

REFERENCES

[1.] B. P. S. BPS, "Produksi Tanaman Florikultura (Hias) 2020," 2020. https://www.bps.go.id/indicator/55/64/1/produksi-tanaman-florikultura-hias-.html (accessed Dec. 26, 2021).

[2.] D. W. Darmono, *Menghasilkan Anggrek Silangan*. Depok: Penebar Swadaya, 2005.

[3.] V. D. Bachuwar, A. D. Shligram, and L. P. Deshmukh, "Monitoring the soil parameters using IoT and Android based application for smart agriculture," 2018, p. 020003. doi: 10.1063/1.5047679.

[4.] F. Rahmasari, "Integrasi Sistem Pemantauan dan Pengendalian Smart Agriculture dan Smart Farm Berbasis Android," 2020.

[5.] V. Vinesta, "Sistem Pemantauan dan Pengendalian Greenhouse Anggrek di SMK Negeri 1 Bawen," 2021.