Air Detection Environment System (ADeV) Android-Based Application Detect Air Quality Levels in Parking Area

Ade Silvia Handayani[1,*] Sopian Soim[1] Al Fatur Sayid Muhammad[1] Nyayu Latifah Husni[1] Rumiasih[1] Rosmalinda Permatasari[2] Andry Meylani[3]

[1] *Electrical Engineering, State of Polytechnic Sriwijaya; Jl. Srijaya Negara Bukit Besar 30139*
[2] *Civil Engineering, Tridinanti University; Jl. Kapten Marzuki No.2446*
[3] *Informatics Technology, Bina Darma University; Jl. Jenderal Ahmad Yani No.3, 9/10 Ulu*
[*] *Corresponding author. Email: ade_silvia@polsri.ac.id*

ABSTRACT

Motorized vehicles have exhaust gas which is one of the biggest factors causing high levels of air pollution because they contain harmful gases such as CO, CO2, HC and others. The indoor parking area is the area that is exposed to the largest exhaust gas apart from roads. Therefore, it is necessary to detect air quality to determine the level of exhaust gas in the area. In several previous research, some tools were created to detect the levels of these gases, but they tend to be expensive and not flexible. This research presents an Android-based Air Detection Environment System application using WSN technology to detect air quality levels in the parking area with multi-sensors as hardware for air quality detection systems in parking environments. The application designed to successfully detect air quality due to vehicle exhaust gas in the kind of CO, CO2, HC, PM10, temperature and humidity. This multi-sensor technology can send air quality data to the server and display android applications in the form of air quality status, time and location of the incident. The advantages of this application, the public can find out the environmental conditions in the parking lot in practical, real-time and mobile because they only need to open the application on the smartphone. The existence of this application is expected to increase safety and awareness of individuals against the quality of an unhealthy environment.

Keywords: *android, detection, air quality, wireless sensor network*

1. INTRODUCTION

Air pollution has been always discussed as a topic of environmental problems due to the more negative impact caused. Based on data from the World Health Organization (WHO), air pollution has a big impact and risk for health which 9 out 10 people breath air containing high level pollutants[1][2][3]. High levels of pollution have been shown to have adverse impact on humans, vegetation, agricultural crops, forest, also damage to building surface[4][5]. It estimated to cause 7 million people die every year from exposure to polluted air. Ambient air pollution alone caused some 4.2 million deaths in 2016, while household air pollution from cooking with polluting fuels and technologies caused an estimated 3.8 million deaths in the same period [3][6][7].

Air Pollution appears by several things such as industrial development and construction which always increase. One of the biggest things which caused an increase in air pollution is exhaust gas results from motorized vehicle transportation in an urban area[7][8]. Motorized vehicles produce exhaust gases which very dangerous for the environment. The process of burning fuel from motorbikes and cars produces exhaust gas which generally contains levels of CO, CO_2, NO2, HC, C, H_2, H_2O and N_2[1][6][9][10]. This condition causes the environment to be polluted by exhaust gas resulting from vehicle oxidation contain toxins that harmful to the environment[6].

Gas levels from vehicle fuel combustion can be measured using tools such as the Lutron AQ-9901SD Air Quality Meter, National Air Pollution Monitoring Network – NABEL[11], etc. The measuring equipment used provides very accurate data but tends to be very high in price for a significant scale-up. Therefore, a tool was created to measure air quality which produces data such as [11][12] but has a more affordable price[13][14][15][16].

In [17], A monitoring system based on Arduino microcontroller with a Bluetooth HC-05 module as a communication medium for sending data to smartphones. However, this tool cannot be widely applied due to the limitations of Bluetooth which has short-range and limited access in various places.

The technology that can be used to measure air quality is the Wireless Sensor Network (WSN) technology[18]. The WSN consists of several special sensor nodes which can detect and monitor physical parameters[19][20]. Some of the monitoring is carried out, such as temperature, humidity, pressure, shift and others automatically with a smart device so that it can to process data in real-time[21][22] and data results will be sent to the server and displayed on a smartphone via an Android-based application interface[23].

Android-based application as a software platform for Linux-based smartphone that can be used and developed openly or open source[24]. This makes it possible for developers to set, modify or even create their applications. It used to make users easier to access the internet flexibly using smartphones without limited space.

This implementation of Android-based mobile technology in the air quality detection system is carried out due to data shown by the Digital Marketing Research Institute Emarketer, it is stated that in 2018 more than 100 million Indonesians are active users of smartphones. Unfortunately, it makes Indonesia has ranked the fourth largest country with active smartphone users after China, India, and America[25]. This prompted the creation of an android application to measure air quality so that people could easily access air quality information which must be of sufficient quality.

Maintaining the quality of a mobile application device is very necessary to avoid errors when it is used by people. It is necessary to test using a method before being published to the public so that the application can operate properly, one of the methods that can be used is the ISO25010 method[26][27].

This research will design an Air Detection Environment System (ADeV) application, which is an Android-based application that will display air quality data in a parking area. This data obtained from a detection tool in the form of WSN technology with multi-sensors. This tool will detect air quality due to vehicle exhaust gases of CO, CO2, HC, PM10, temperature and humidity.

2. WSN IMPLEMENTATION AS AIR QUALITY MONITORING

In [28], a wireless sensor network system was applied to monitor air quality using DHT22 (temperature), MQ135 (air) and MQ7 (CO) sensors. The data obtained from the sensor in real-time is entered into a database server via a network using the Raspberry Pi which can be accessed in real-time via a web server. In [29] Paper, WSN has been implemented as a monitoring agricultural area such as temperature and humidity around the area. Continuous monitoring of these key environmental variables can assist farmers in improving the quality and productivity of food crops.

WSN applications are usually used for commercial and industrial applications to monitor data that would be difficult or expensive to monitor using wired sensors[30][31]. In its application, WSNs are spread over areas intended to collect data through their sensor nodes.

3. ANDROID AS AN IOT-BASED AIR QUALITY MONITORING APPLICATION

IoT has been implemented in many application such as [32-34] in The implementation of android applications in air quality monitoring systems based on the Internet of Things has been carried out in several previous studies, such as in [35] research, which implementing IoT-based android applications as a means of air quality monitoring for smart cities. In this research, MQ-2 sensor has been used to detect Carbon Monoxide (CO), Liquefied Petroleum Gas (LPG) and Smoke. DHT11 sensor have been used to measure Temperature and Humidity levels of the surrounding. Also, to measure the PPM levels of PM 2.5 and PM 10 particulate matters, SDS021 sensor has been interfaced with the Raspberry Pi.

Meanwhile, the research journal [23] designed an android application to monitor the quality of agricultural land online. The application is equipped with a database recap system that can call databases on internet servers. Sensor devices that are integrated with the Android application will be able to monitor rice fields so that farmers can see the quality of soil and water through the Android application. The application is equipped with a database recap system that can call databases on an internet server. Sensor devices that are integrated with the Android application will be able to monitor rice fields, so that farmers can see the quality of soil and water through the Android application.

4. RESEARCH METHODOLOGY

The steps of this research are hardware and software design and the final steps are system integration and system testing. Design of the air quality detection system hardware using WSN technology is a tool that is described in the following block diagram:

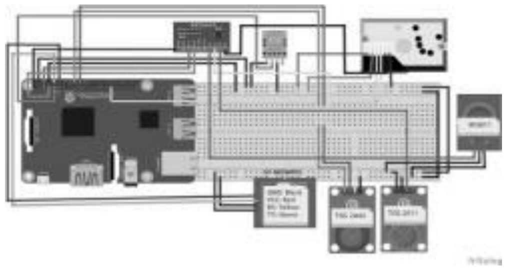


Figure. 1 Hardware Schematic Diagram

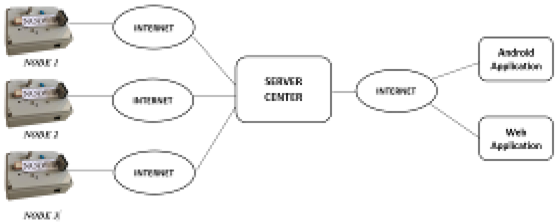


Figure. 2 Sensor Node Schematic Diagram

In Fig. 1, a design for an air quality detection system using three sensor nodes. Each sensor node consists of several supporting tools as shown in Fig. 2. The devices used are the Raspberry PI 3 model B microprocessor equipped with a TGS2442 sensor as a CO measuring sensor, MG811 sensor as a CO2 measuring sensor, TGS2611 sensor as a measuring sensor Hydro-carbon levels, the GP2Y1010AU0F sensor as a particulate dust gauge, the Neo6m GPS sensor as a sensor for mapping maps and the DHT11 sensor as a temperature and humidity sensor. Also, the ADC module or Analog to Digital Converter, namely ADS1115, is a converter of sensor reading values so that the Raspberry PI can process it as well as an I2C communication module.

After the sensor nodes send the environmental air quality detection results to the central server, the data will be processed and displayed in the android application. The goal is that it can be accessed quickly and easily to obtain information on certain environmental conditions without limited access. The diagram of the air quality detection system software using the Wireless Sensor Network is shown in Fig. 3.

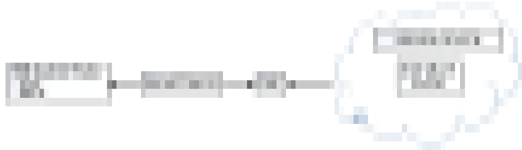


Figure. 3. Block Diagram Software Systems

Based on the block diagram in Fig. 3, an Android application designed using the Android Studio IDE software. IDE Android Studio is software that is used for developing android applications that are integrated with XML, Java, and Kotlin. In this research using XML and java languages.

The application design that will be made as a reference for application development can be seen as figure below,

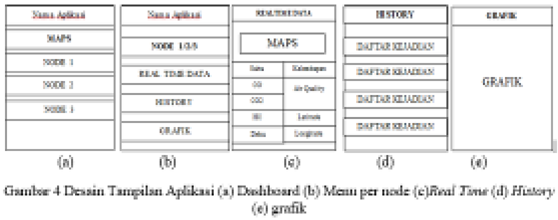


Figure. 4. Display Application Design (a) Dashboard (b) Menu per Node (c) Real Time (d) History (e) Graph

In the Android application display, the dashboard page is shown in Fig. 4 (a) which contains 3 button nodes and one map to map the sensor nodes. When the node button clicked, the node menu will appear as shown in Fig. 4 (b). On the node menu page, there are 3 options, namely real-time data, history and graphics. The real-time data page is shown in Fig. 4 (c) which shows the current condition of the air quality level detection results. The history page is shown in Fig. 5 (d), this page shows the air quality level data recorded by the sensor node then the graph page is shown in Fig. 4 (e) which will display the air quality level reading data in the form of a line graph.

5. RESULT AND DISCUSSION

5.1 Air Quality Detection System Hardware Design

Hardware design applied to a white box which has been successfully made based on the design sketch of the tool. In Fig. 5 (a) shows the display on the hardware of the air quality detection system with Raspberry pi 3 as a location to process data read by the sensor. The sensors placed on the lid of the box to read the air quality levels around it.

(a) (b)

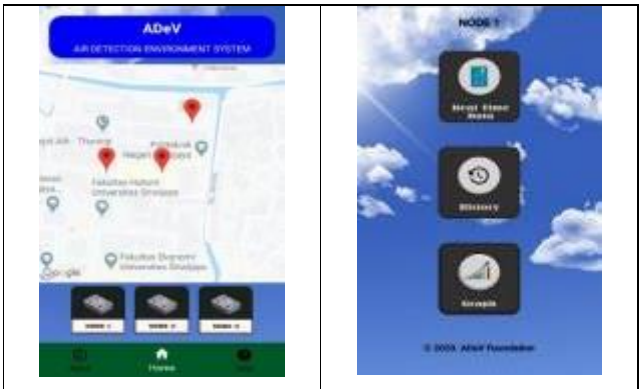
Figure. 5 Hardware (a) Component Inside (b) Display

In Fig. 5 (b) above, there are three sensor nodes used, namely node 1, node 2 and node 3. The sensors are placed on the cover of each sensor node to detect the level of air quality around it and provide map coordinates. The sensors used are the DHT11 sensor which functions as a sensor for measuring temperature and humidity, the MG811 sensor functions as a sensor for measuring CO2 gas levels, the GP2Y1010AU0F sensor is used as a sensor for measuring PM10 or dust levels, the TGS2442 sensor functions to measure CO levels, the TGS2611 sensor functions for measuring HC levels, and the GPS module NEO6M is used to get map coordinates.

5.2 Air Quality Detection System Software Design

The software has been successfully created based on block diagrams and display sketch designs that have been made. The software obtained is an application to detect air quality in the parking environment that has been integrated with a smartphone made using the Android Studio IDE software. The android application for measuring air quality levels is named "Air Detection Environment (ADeV) Application".

Fig. 6 (a) shows the main menu on the ADeV application display which consists of button node 1, node 2, and node 3. The main menu also displays maps for the current location of the 3 sensor nodes used. In Fig. 6 (b) below is a display if one of the button nodes is clicked (in this case a sample of node 1 is taken). In this display, 3 menus are generated, namely the real-time data, history, and graphics buttons.



(a) (b)

(c) (d)

Figure. 6 Display Layout (a) Main Menu (b) Node Menu (c) Real Time Data (d) History

Fig. 6 (c) will display the layout when the Real-Time Data button clicked. The layout called real-time data because this menu only displays the latest data on air quality levels read by the sensors in real-time. Data will continue to change as long as each sensor node device continues to read data for a certain time and remains connected to the internet network.

Fig. 6 (d) above show the display when History button clicked and it will present all air quality data and map coordinates read by the sensor nodes.

Figure. 7. Line Chart Layout Display

In Fig. 7 above, the graph will display air quality data according to the node, the selected time range, and the parameters you want to display. After that, when the generate button clicked, a line graph will appear according to the time range and selected parameters. This will make it easier for people to see the development of air quality levels at a certain time.

6. TESTING OF ADEV APPLICATIONS

6.1. ADeV Applications Test Using ISO 25010 Standards

The final process of application design is testing the application. In this research, testing used the ISO 25010 standard.

1) *Functional Suitability Test*

Functional suitability testing aims to check whether all the features of the designed application can be operated or not. Table 2 is the functional suitability test data.

Table 2. Functional Suitability Test Data

No	Feature	Evaluator					Score
		1	2	3	4	5	
1	Main Layout Displayed	1	1	1	1	1	5
2	Real Time Data Displayed	1	1	1	1	1	5
3	History Layout Displayed	1	1	1	1	1	5
4	Graph Layout Displayed	1	1	1	1	1	5
5	Function Button	1	1	1	1	1	5
6	Function Bar	1	1	1	1	1	5
Total		6	6	6	6	6	30

Eligibility Percentage (%) $= \frac{score\ obtained}{maximum\ score}\ x\ 100\%$

$= \frac{30}{30}\ x\ 100\%$

= 100%

Percentage of the calculated data obtained is 100%, so it can be concluded that all ADeV features are 100% can be used properly. Based on this percentage, the statement "Very Appropriate" obtained in the functional suitability parameter.

2) *Compatibility Test*

The aspect of compatibility consists of co-existence, testing co-run applications, testing on various operating systems, and testing different types of devices.

Tabel 3 Co-Existence Testing Data

No	Application		Success	Failed
1	ADeV	*Facebook*	1	0
2	ADeV	*Google Chrome*	1	0
3	ADeV	*Instagram*	1	0
4	ADeV	*Maps*	1	0
5	ADeV	*Youtube*	1	0
6	ADeV	*Gojek*	1	0
Total			6	0

Table 3 shows the ADeV application test data that is run simultaneously with 6 different applications on the same smartphone. Based on testing, the following percentage data were obtained as below.

Eligibility Percentage (%) $= \frac{score\ obtained}{maximum\ score}\ x\ 100\%$

$= \frac{30}{30}\ x\ 100\%$

= 100%

The percentage result obtained 100%, so it can be concluded that the ADeV application from the Co-Existence side is “Very Appropriate”.

Table 4 shows the ADeV test data that is run on different operating systems and smartphones. Testing includes suite setup or installation, build-in explorer or running processes, and teardown suite or uninstall applications.

Table 4 Summary Test of The Operating System and Device Types

No	Device	Setup Suite	Build-in-Explorer Suite	Teardown Suite	Score
1	Xiaomi Redmi 4A (7.1.2)	1	1	1	3
2	Samsung J2 Pro (7.1.1)	1	1	1	3
3	Asus (9.0)	1	1	1	3
4	Xiaomi redmi 3S (6.0.1)	1	1	1	3
5	Oppo A37(5.1.1)	0	0	0	0
6	Samsung galaxy J3 pro (7.0)	1	1	1	3
7	Samsung J5 Prime (8.0)	1	1	1	3
Total		6	6	6	21

Eligibility Percentage (%) $= \frac{score\ obtained}{maximum\ score}\ x\ 100\%$

$= \frac{30}{30}\ x\ 100\%$

= 100%

The calculation data obtained 85%, so ADeV matches the compatibility test with the predicate "Very Appropriate".

3) *Portability Test*

Portability test carried out by operating ADeV on Android mobile devices that have different operating systems and screen sizes. Based on Table 5, it can be seen that the results of the ADeV application portability test show that there are "Success" and "Failed". In this case, the dominant criterion is "Success" which means that every function contained in the application can run well when the application is run on multiple operating systems and varying screen resolutions.

Table 5 Portability Test Data

No	Rated Aspect	Result
A	Operating System	
1	Lollipop	Failed
2	Marshmallow	Success
3	Nougat	Success
4	Oreo	Success
5	Pie	Success
B	Screen Resolution	
1	MDPI : WXGA (1280 x 800 piksel)	Success
2	HDPI : WVGA (480 x 800 piksel)	Success
3	LDPI : WSVGA (1024x600 piksel)	Success

4) *Performance Efficiency Test*

Performance efficiency testing is carried out by calculating the average response time of the ADeV application for launching. In this test, it was carried out 3 times with three different types of smartphones and different specifications.

The first test, the Android smartphone used has 2 GB RAM specifications; 1.4GHz Cortex-A53 quad-core CPU; and Android OS 7.1.1 (Nougat). In the second test the Android smartphone used had 1.5 GB RAM specifications; Qualcomm Snapdragon 425 CPU; and Android OS 6.0.1 (Marshmallow). In the third test, the Android smartphone used had 6 GB RAM specifications; Snapdragon 636 CPU; Android OS 9.0 (Pie).

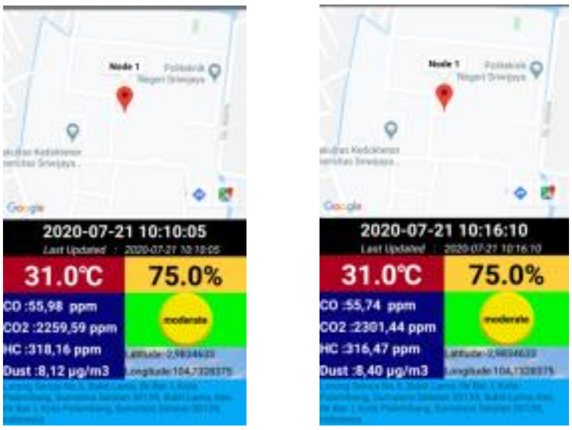
Table 6 Test Result Performance Efficiency

No	Category	*Response Time* (s)		
		Test 1	Test 2	Test 3
1	Splash Screen (*launch time*)	4.85	4.99	4.26
2	Main Menu Displayed	2.39	2.35	2.20
3	Node Menu Displayed	0.8	1.11	0.50
4	Real-Time Data Layout Displayed	1.69	1.82	1.56
5	History Layout Displayed	3.88	4.76	1.19
6	Graph Layout Displayed	4.73	5.22	3.53
Total		18,34	20,27	13,24
Average		3,06	3,39	2,21

From the test, it is concluded that the type of processor and RAM affects the performance value of an application. Tests on the third test obtained faster results when compared to the first and second tests. This is because test 3 uses the highest processor and RAM.

6.2. ADeV Applications Test in Detecting Air Quality Levels

This test carried out to test the results of the ADeV application design in detecting air quality. Testing is done by integrating the ADeV application with only one sensor node to get the air quality level detection results.



(a) (b)

Figure. 8 Real Time Display on 21st July, 2020 at (a) 10.10 AM and (b) 10.16 AM

(a) (b)

Figure. 9 History Display on 21st July, 2020 at (a) 10.23 AM - 10.20 AM (b) 10.18 AM - 10.14 AM

Figure. 10. CO Levels Graph

Figure. 11. CO2 Levels Graph

Figure. 12. HC Levels Graph

Figure. 13. Dust Levels Graph

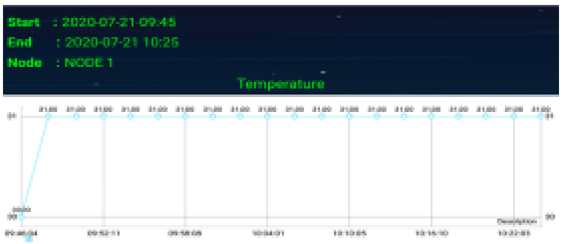


Figure. 14. Temperature Levels Graph

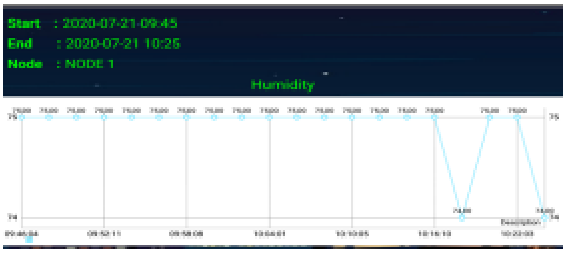


Figure. 15. Humidity Levels Graph

Based on the Real-Time data in Figure 8, the ADeV application has good performance because it can transmit information without time-lapse or real-time. In this figure, there is a circle of yellow and red. The yellow circle indicates that the air quality at that time is Moderate, while the red circle indicates that the air quality at that time is Hazardous. The location in the form of latitude and longitude appears according to the location of the device that is displayed on maps.

In Fig. 9, displays the sensor reading history, time recording occurred, and the location of the sensor node. This display has good performance because it can display historical data in the webserver database. Fig. 10-15 are the results of sensor readings in graphic form at a certain time unit. This graph also has good performance because it can display sensor readings in the database in the form of a line diagram.

7. ADeV APPLICATION PERFORMANCE ANALYSIS

Designing the hardware sensor node, this research uses a Raspberry PI 3 microprocessor to give commands, process all inputs and manage all outputs, air quality sensors as input and GPS and I2C sensors as supporting modules. The TGS2442 sensor used as a CO sensor, the 2611 sensor as an HC sensor, the MG811 sensor as a CO2 sensor, the GP2Y1010AU0F sensor as a dust particulate sensor, and the DHT11 sensor as a temperature and humidity sensor.

The ADeV application designed to provide several features, namely providing air quality information. Air quality information that can be accessed includes real-time conditions, history and graphs of gas levels at certain times. Real-Time used to see the latest air quality conditions, history features and graphics so that previous data can be view and access.

The quality of the ADeV application gets “very decent” functional suitability testing results because all ADeV features operate smoothly. The compatibility test was “very feasible” because ADeV can operate in conjunction with other applications. Meanwhile, the portability aspect of the application is declared "feasible" because the application can run on various screen sizes of a device and can be operated on several types of devices and versions of Android with a minimum Android Marshmallow and a maximum of Pie. For performance efficiency test results, the ADeV application has a fast response in displaying the layout and retrieving data stored on the database server.

The data displayed by Android is the result of the integration of sensor nodes that send data wirelessly to the server then the data is stored into the database. Then android reads data on the server using the HTTP protocol so that it can access data to be displayed on the user's Android device. Due to this system is based on IoT, an adequate internet connection is required to obtain accurate and real-time data.

8. CONCLUSION

The ADeV application as an Android-based air quality detection system application has a good ability to detect air quality in the form of real-time display of air quality data, time and location, history, and graphics. Based on the ISO 25010 standard, the air quality detection system using the ADeV application has high compatibility and good efficiency in almost all Android operating systems and its use is more flexible than using a computer or laptop.

REFERENCES

[1] J. B. Newbury *et al.*, “Association of air pollution exposure with psychotic experiences during adolescence,” *JAMA Psychiatry*, vol. 76, no. 6, pp. 614–623, 2019, doi: 10.1001/jamapsychiatry.2019.0056.

[2] Z. Sheng, S. Wang, X. Zhang, X. Li, B. Li, and Z. Zhang, “Long-Term Exposure to Low-Dose Lead Induced Deterioration in Bone Microstructure of Male Mice,” *Biol. Trace Elem. Res.*, vol. 195, no. 2, pp. 491–498, 2020, doi: 10.1007/s12011-019-01864-7.

[3] World Health Organization, “Air pollution.” https://www.who.int/news-room/air-pollution (accessed Dec. 20, 2020).

[4] M. Peng, H. Zhang, R. D. Evans, X. Zhong, and K. Yang, “Actual Air Pollution, Environmental Transparency, and the Perception of Air Pollution in China,” *J. Environ. Dev.*, vol. 28, no. 1, pp. 78–105, 2019, doi: 10.1177/1070496518821713.

[5] Y. Qin and H. Zhu, “Run away? Air pollution and emigration interests in China,” *J. Popul. Econ.*, vol. 31, no. 1, pp. 235–266, 2018, doi: 10.1007/s00148-017-0653-0.

[6] F. J. Kelly and J. C. Fussell, “Air pollution and public health: emerging hazards and improved understanding of risk,” *Environ. Geochem. Health*, vol. 37, no. 4, pp. 631–649, 2015, doi: 10.1007/s10653-015-9720-1.

[7] D. Parenden and Cipto, “Estimation of emissions for petrol vehicles in some roads in Merauke city,” *Int. J. Mech. Eng. Technol.*, vol. 10, no. 1, pp. 326–334, 2019.

[8] H. Kotta, S. Mangkoedihardjo, Y. Ludang, and S. Trisutomo, “The design of riparian zone in waterfront area of Tanjung Bunga, Makassar,” *Int. J. Civ. Eng. Technol.*, vol. 9, no. 8, pp. 580–584, 2018.
[9] N. Amarachi, O. Emeka, A. Christopher, A. Lovell, and E. Conrad, “Automotive Exhausts,” vol. 6, no. 4, 2016.
[10] M. Watany, “Variability in Vehicle’ Exhaust Emissions and Fuel Consumption in Urban Driving Pattern,” *Am. J. Veh. Des.*, vol. 3, no. 1, pp. 31–38, 2015, doi: 10.12691/ajvd-3-1-5.
[11] A. Arfire, A. Marjovi, and A. Martinoli, “Mitigating Slow Dynamics of Low-Cost Chemical Sensors for Mobile Air Quality Monitoring Sensor Networks,” *Proceeding EWSN ’16 Proc. 2016 Int. Conf. Embed. Wirel. Syst. Networks*, pp. 159–167, 2016.
[12] H. Agra, T. Klein, A. Vasl, G. Kadas, and L. Blaustein, “Measuring the effect of plant-community composition on carbon fixation on green roofs,” *Urban For. Urban Green.*, vol. 24, pp. 1–4, 2017, doi: 10.1016/j.ufug.2017.03.003.
[13] P. Jiang, C. Li, R. Li, and H. Yang, “An innovative hybrid air pollution early-warning system based on pollutants forecasting and Extenics evaluation,” *Knowledge-Based Syst.*, vol. 164, pp. 174–192, 2019, doi: 10.1016/j.knosys.2018.10.036.
[14] S. J. Johnston *et al.*, “City scale particulate matter monitoring using LoRaWAN based air quality IoT devices,” *Sensors (Switzerland)*, vol. 19, no. 1, pp. 1–20, 2019, doi: 10.3390/s19010209.
[15] T. Sayahi *et al.*, “Development of a calibration chamber to evaluate the performance of low-cost particulate matter sensors,” *Environ. Pollut.*, vol. 255, 2019, doi: 10.1016/j.envpol.2019.113131.
[16] D. Hasenfratz *et al.*, “Deriving high-resolution urban air pollution maps using mobile sensor nodes,” *Pervasive Mob. Comput.*, vol. 16, no. PB, pp. 268–285, 2015, doi: 10.1016/j.pmcj.2014.11.008.
[17] N. Kaur, R. Mahajan, D. Bagai, and P. G. Student, “Air Quality Monitoring System based on Arduino Microcontroller,” *Int. J. Innov. Res. Sci. Eng. Technol. (An ISO Certif. Organ.*, vol. 5, no. 6, pp. 9635–9646, 2016, doi: 10.15680/IJIRSET.2015.0506018.
[18] A. Sahfutri, N. L. Husni, M. Nawawi, I. Lutfi, A. Silvia, and E. Prihatini, “Smart Parking Using Wireless Sensor Network System,” *2018 Int. Conf. Electr. Eng. Comput. Sci.*, vol. 17, pp. 117–122.
[19] P. Kumar and K. K. Naik, “Android based wireless sensor network application for airborne platforms (AWAAPS),” *Proc. 2nd IEEE Int. Conf. Eng. Technol. ICETECH 2016*, no. March, pp. 776–782, 2016, doi: 10.1109/ICETECH.2016.7569353.
[20] A. S. Handayani, N. L. Husni, S. Nurmaini, and R. Permatasari, “Environmental Application with Multi Sensor Network,” vol. 9, no. 1, 2020.
[21] M. Ta, “applied sciences Real-Time Monitoring of Indoor Air Quality with Internet of Things-Based E-Nose,” 2019.
[22] A. Pangestu, M. Yusro, W. Djatmiko, and A. Jaenul, “THE MONITORING SYSTEM OF INDOOR AIR QUALITY BASED ON INTERNET OF THINGS,” vol. 5, no. 2, pp. 141–152, 2020.
[23] A. Kurubaran, V. K. S, K. Vignesh, and A. Bhaskaran, “Air Pollution Monitoring System using Android Application,” vol. 10, no. 21, pp. 112–115, 2018.
[24] B. F. Tavares, I. M. Pires, G. Marques, and N. M. Garcia, “Mobile Applications for Training Plan Using Android Devices : A Systematic Review and a Taxonomy Proposal,” doi: 10.3390/info11070343.
[25] Ministry of Communication and Information, “Kementerian Komunikasi dan Informatika.” https://kominfo.go.id/content/detail/6095/indonesia-raksasa-teknologi-digital-asia/0/sorotan_media (accessed Dec. 20, 2020).
[26] International Organization for Standardization, “ISO/IEC 25010:2011(en), Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models.” https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en:en (accessed Dec. 20, 2020).
[27] B. Wulandari, G. N. I. P. Pratama, N. Hasanah, and N. Yuniarti, “Augmented Reality As Android Based Learning Media for Wood Field Laboratory Work,” *J. Phys. Conf. Ser.*, vol. 1413, no. 1, 2019, doi: 10.1088/1742-6596/1413/1/012035.
[28] N. A. Zakaria, Z. Z. Abidin, N. Harum, L. C. Hau, N. S. Ali, and F. A. Jafar, “Wireless internet of things-based air quality device for smart pollution monitoring,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 11, pp. 65–69, 2018, doi: 10.14569/IJACSA.2018.091110.
[29] M. R. M. Kassim and A. N. Harun, “Applications of WSN in agricultural environment monitoring systems,” *2016 Int. Conf. Inf. Commun. Technol. Converg. ICTC 2016*, pp. 344–349, 2016, doi: 10.1109/ICTC.2016.7763493.
[30] O. H. Yahya, H. T. S. Alrikabi, and I. A. Aljazaery, “Reducing the data rate in internet of things applications by using wireless sensor network,” *Int. J. online Biomed. Eng.*, vol. 16, no. 3, pp. 107–116, 2020, doi: 10.3991/ijoe.v16i03.13021.
[31] M. Abdulkarem, K. Samsudin, F. Z. Rokhani,

and M. F. A Rasid, “Wireless sensor network for structural health monitoring: A contemporary review of technologies, challenges, and future direction,” *Struct. Heal. Monit.*, vol. 19, no. 3, pp. 693–735, 2020, doi: 10.1177/1475921719854528.

[32] Latifah, N.H., Sitangsu, S., Rasyad, S. and Handayani, A.S., 2020. Real Time Garbage Bin Capacity Monitoring. *Computer Engineering and Applications Journal*, *9*(2), pp.127-134.

[33] N. L. Husni, Robi, E. Prihatini, Nurhaida, A. Silvia and Firdaus, "Garbage Monitoring and Warning System," *2019 International Conference on Electrical Engineering and Computer Science (ICECOS)*, Batam Island, Indonesia, 2019, pp. 171-175.

[34] A. S. Handayani, N. L. Husni, R. Permatasari and C. R. Sitompul, "Implementation of Multi Sensor Network as Air Monitoring Using IoT Applications," *2019 34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, JeJu, Korea (South), 2019, pp. 1-4.

[35] B. Kalinga Institute of Industrial Technology, IEEE Singapore Section, Institute of Electrical and Electronics Engineers, IEEE Power Electronics Society, and IEEE Industry Applications Society, “1st IEEE International Conference on Sustainable Energy Technologies and Systems, IEEE-ICSETS 2019 : 26 February – 1 March 2019, Kalinga Institute of Industrial Technology, Bhubaneswar, Odisha, India,” *2019 IEEE Int. Conf. Sustain. Energy Technol.*, pp. 173–177, 2019.