

Internet of Things-Based Bus and Student Monitoring System on Free School Transportation Madiun City

Arfian Dwiki Rosyadi¹, Lintang Diah Puspaningrum², Musawer Hakimi³

¹²Politeknik Negeri Madiun, Indonesia

³Samangan University, Afghanistan

Corresponding : fian.scp@gmail.com ¹

Received : March 26, 2024

Accepted : April 02, 2024

Published : April 23, 2024

Citation: Rosyadi, A. D., Puspaningrum, L. D., & Hakimi, M. (2024). Internet of Things-Based Bus and Student Monitoring System on Free School Transportation Madiun City. Digitus : Journal of Computer Science Applications, 2(2), 69-84.

<https://doi.org/10.61978/digitus.v2i2.464>

ABSTRACT: Traffic incidents and violations, particularly among students, pose safety concerns for parents and authorities. To address this issue, the Madiun City Government introduced the Free School Transport program through the Madiun City Transportation Office, providing school buses for students in Madiun City. However, despite this initiative, students and parents face challenges tracking bus arrivals due to weather conditions, traffic congestion, and schedule delays. This study proposes developing MASBUS, an Internet of Things (IoT)-based monitoring system for real-time school bus tracking. The system integrates a GPS module for location tracking and an RFID module for student identification. Data is transmitted via NodeMCU ESP8266 to a web server, stored in a MySQL database, and accessed through an Android application. The MASBUS app provides real-time location updates and sends automatic notifications when students board or leave the bus. The implementation of this system enhances student safety, improves transportation efficiency, and enables real-time monitoring for parents and school authorities. MASBUS contributes to a smarter and safer school transportation system by offering a more transparent and reliable tracking solution.

Keywords: Bus, Free School Transportation, Internet of Things, Monitoring System, Student.



This is an open access article under the CC-BY 4.0 license

INTRODUCTION

Madiun City is one of the cities in East Java Province. According to the Madiun City Statistics Center (2021), the population of Madiun City in September 2020 according to the SP2020 results was 195,175 people, with an area of 33.23 km², the population density of Madiun City based on the results of the 2020 population census was 5,873 people per km² (Madiun, 2021). Given this population density, community activities have also become busier, particularly regarding personal transportation needs. As stated by Ulfatiana & Parwiyanto (2020), the increasing number of private vehicles makes traffic conditions worse, particularly among students who do not yet have a driver's license (Ulfatiana & Parwiyanto, 2020).

Based on data from 2015-2018, the second highest number of accidents by profession is students, namely 486 accidents and from 2017-2018 there have been 12,500 cases of violations by students

with the highest violation cases of not carrying and not having a motorcycle driving license as many as 23,873 of the total violations that occurred in Madiun City (Wahyuni, 2020). To address this issue, the Madiun City Government, through the Transportation Department, introduced the Free School Transport (ASG) program on February 1, 2019. This program aims to provide safe transportation for students and reduce private vehicle use (Alm et al., 2014). Currently, the ASG fleet consists of five buses and 20 rented public vehicles operated by private parties under the department's management.

School transportation plays a crucial role in ensuring student safety and accessibility. However, challenges such as unpredictable bus schedules, traffic delays, and the inability of parents to track their children's journey persist. In response, the Madiun City Government, through the Madiun City Transportation Office, introduced the Free School Transport program to facilitate student commuting (Chen et al., 2022; Daniels & von der Ruhr, 2014). Despite this initiative, a major challenge remains: an integrated system that enables real-time tracking of school buses and student presence.

Existing studies have explored IoT-based monitoring systems for transportation safety. Asriyadi et al. (2022) developed a portable security system integrating GPS, RFID, and NodeMCU to track the whereabouts of travel bags. Similarly, Haryanto et al. (2022) designed a student location monitoring system using Ublox Neo-6M GPS and NodeMCU V3, allowing parents to track their children's journey home in real-time (Haryanto & Wiranata, 2022). While these studies demonstrate the effectiveness of GPS and IoT-based tracking, they primarily focus on personal security and student monitoring outside school bus systems (PARULIAN et al., 2023).

This study builds upon previous research by developing MASBUS, an IoT-based school bus tracking system that integrates GPS for real-time location tracking and RFID for student identification. The system transmits data via NodeMCU ESP8266 to a web server, stores it in a MySQL database, and provides real-time updates through an Android application. By addressing existing gaps, MASBUS enhances student safety, improves transportation efficiency, and offers a comprehensive monitoring solution for parents and authorities (Nurani et al., 2022).

However, MASBUS has certain operational limitations. The system relies on an active internet connection via WiFi for real-time data transmission and tracking. This dependency may affect system performance in areas with unstable or no internet coverage, potentially impacting the accuracy of student monitoring and bus tracking.

Table 1. Research Literature Review

No.	Title	Author	System Overview	Differences
1	Design of Portable Security System Using GPS and RFID Based on NodeMCU.	Asriyadi, et al., 2022	Developed an IoT-based security system using GPS and RFID technology integrated with presence of travel bags. In NodeMCU to detect a contrast, our research travel bag's presence and focuses on monitoring ensure its contents' security. The system stores data and utilizing an Android connects to Wi-Fi to track real-time location using the Blynk application.	This study focuses on securing personal belongings by detecting the presence of travel bags. In contrast, our research focuses on monitoring school buses and students, developed with Android Studio and Kotlin instead of Blynk.
2	Student Location Monitoring System Using Ublox Neo-6M GPS at SD Muhammadiyah 1 Surakarta	Haryanto, et al., 2022	Developed an IoT-based student location monitoring system using NodeMCU V3, Ublox Neo-6M GPS, and Firebase Realtime Database. This system lets parents track their children's location in real-time via an Android application while commuting home.	Unlike this study, our research employs the Neo-6M GPS module, which offers higher accuracy in determining the school bus's location. Additionally, our system integrates RFID technology to monitor students' presence in the bus, providing parents with precise departure and arrival information via an Android application.

METHOD

The experimental research method will be used. This method was chosen because it refers to the research objectives of the author's final project, namely to produce an Internet of Things (IoT) device product with a system for monitoring buses and students on Madiun City Free School Transport. The stages of the experimental method can be seen in Figure 1.

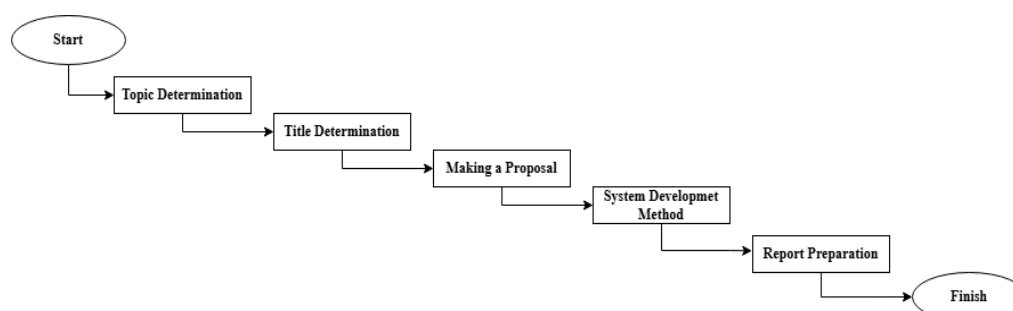


Figure 1. Experimental Methods

1. Topic Selection

The research begins with defining the topic, which is the foundation for achieving the

research objectives. In this case, the focus is on developing a school bus and student monitoring system that uses IoT technology to track bus locations and student presence.

2. Title Definition

A precise and unambiguous research title is formulated based on the chosen topic to ensure clarity and relevance. The study is titled “Internet of Things-Based Bus and Student Monitoring System on Free School Transportation Madiun City”(Lee et al., 2023; Yin et al., 2020).

3. Proposal Development

A research proposal is prepared to outline the study’s objectives, methodology, and expected outcomes. The proposal is presented to the Madiun City Transportation Office, the managing authority of the school transport system, to seek approval and support.

4. System Development Methodology

The waterfall model is adopted for system development, as it provides a structured and sequential approach suited to projects with well-defined requirements. This method ensures a systematic flow from requirement analysis to implementation and testing, promoting consistency and reliability (Kurniawati & Badrul, 2021).

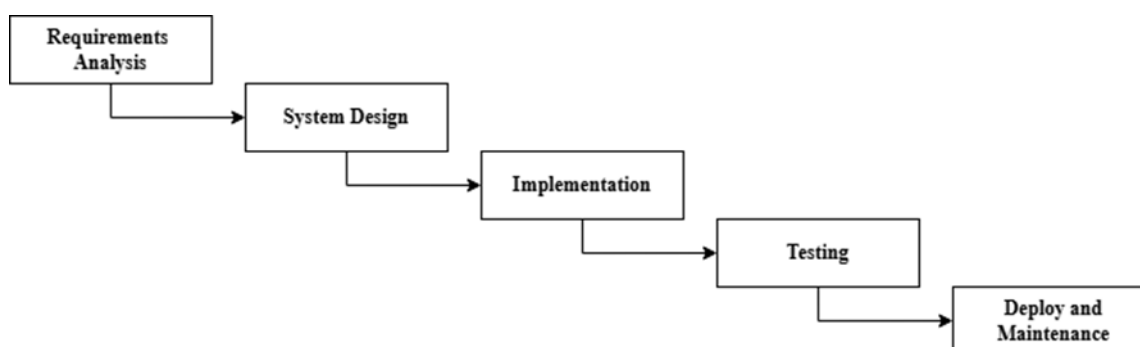


Figure 2. Stages of Waterfall Software System Development Method

Looking at Figure 2 above, the following is an explanation of each stage of the waterfall software development method:

- Requirements analysis, conducting observations, discussions, or interviews with relevant stakeholders to identify the system's needs. Defining key functional requirements, including real-time GPS tracking, RFID-based student identification, and parent notifications.
- System design, including designing system components, block diagrams, schematic designs, flowcharts, and use case diagrams. Developing an architecture where data from GPS and RFID is transmitted to a web server via NodeMCU ESP8266 and processed using a MySQL database (Anjali & Lutfi, 2023; Kuswanto et al., 2023).
- Implementation: The system was developed using Arduino IDE for programming the NodeMCU ESP8266, Visual Studio Code with PHP Laravel framework for the web-based

system, and Android Studio with Kotlin for the mobile application.

- d. System testing (Blackbox testing) and functional testing, verifying that each system feature operates as expected without errors. GPS accuracy, ensuring the system correctly retrieves latitude and longitude and displays it in real time. RFID verification checks if the correct RFID tag ID is scanned and recorded. Data transmission, ensuring data from NodeMCU is stored and displayed correctly in the MySQL database. Notification system, verifying that parent notifications are triggered when students board or leave the bus.
- e. Deployment and maintenance, uploading the web system to hosting services and the Android application to the Google Play Store, and performing maintenance and updates if system errors or feature enhancements are needed.

5. Report Compilation

The final stage involves documenting the research process, findings, and conclusions in a comprehensive report. The report includes the background, system development process, and evaluation results, serving as a reference for future studies and practical implementations.

RESULT AND DISCUSSION

The Internet of Things, also known by its acronym IoT, is a concept that aims to extend the benefits of continuous internet connectivity that allows us to connect machines, appliances, and other physical objects with networked sensors and actuators to acquire data and manage their performance. The Internet of Things utilizes internet connectivity to microcontrollers and sensors to obtain data, which can then be managed within the system. (Yanez et al., 2020). After the data is managed, it is stored in a database and then displayed in an application, either a web application or an Android application (Efendi, 2018).

The tool requirements include hardware to make an IoT device where these tools include NodeMCU ESP8266 V3, Ublox Neo-M8N GPS module, 3V-5V active Buzzer module, LCD 1602

+ I2C, micro-USB data cable, 1x40 pin header, project box X4, 13.56MHz RFID tags, 5V USB output charger module, RFID reader 13.56MHz, small on/off switch, 1s battery indicator, 12V adapter cable, DC jack socket, screwdriver plus, m3 bolt, hot glue gun, cutter, scissors, 40W solder, two 18650 batteries with 4800 mAh capacity, windows-based laptop, Android-based smartphone, 6V input-24V output 5V USB step down module, and 10 cm jumper cable female to female connector. The screen that will be used in the IoT device is an LCD screen that can accommodate as many as two lines of 16 characters and has I2C. I2C is a two-way serial communication standard using two channels designed to send and receive data. The 1602 LCD module with I2C is used because utilizing I2C communication can save the number of pins used and is sufficient to perform simple writing output. (Divito et al., 2022).

This IoT device will also be equipped with a buzzer module. A buzzer works when electricity or electric voltage flows into a circuit that uses piezoelectric. Piezo buzzers can work well in producing frequencies in the range of 1-6 kHz to 100 kHz (Zanofa & Fahrizal, 2021).

The author uses NodeMCU ESP8266 because the module is a microcontroller designed with ESP8266. ESP8266 functions to connect the microcontroller to the Wi-Fi network. NodeMCU is based on the Lua programming language and can use Arduino IDE software. (Asriyadi & Ciksadan, 2022).

The material requirements for working on and completing this research include software and consumables for assembling tools, designing, and programming web and Android systems. These materials include the Windows 10 operating system, Android 12 operating system, Arduino IDE,

Android Studio, Visual Studio Code, browser, Postman, 10 meters of small roller soldering tin, black duct tape, two pieces of glue gun, Fritzting, and MySQL Workbench.

MySQL is one of the most well-known database servers. It uses SQL to access its database. MySQL's license is FOSS License Exception, which has a commercial version. (Ramadhani & Mukhaiyar, 2020).

MySQL Workbench is an application for database management. It allows users to manage, design, and organize MySQL databases with an intuitive interface. MySQL Workbench can also be used for database design, commonly called ERD. (Nasution & Maulana, 2024).

The bus and student monitoring system was developed using a web server, employing PHP with the Laravel framework and a MySQL database. Kotlin was used as the programming language for the Android application.

PHP (Hypertext Preprocessor) is a programming language often used by developers. It is used to create web applications. PHP can run on the server side, often called a server-side language. So, without a web server that keeps running, it will not be able to run. (Fitriana, 2022).

Laravel is a PHP framework released under the MIT license with source code from GitHub. Like other frameworks, Laravel is built with the concept of MVC (Model-Controller-View). Laravel is also equipped with a command line tool called “Artisan” which can be used for bundle packaging and bundle installation via the command prompt (Y. & Rahayu, 2022).

Kotlin is a Java Virtual Machine (JVM)- based programming language developed by JetBrains. It is a pragmatic programming language for Android that combines object-oriented (OO) and functional programming. Because Kotlin is JVM-based, functions in Android Java can be called and used in Kotlin programming.

The system development process began with a needs analysis to identify the requirements for creating the monitoring system and ensure all necessary features were included. During this stage, consideration was also given to the map model that would be integrated into the monitoring system.

The next stage involved system design, which included creating database designs, system block diagrams, interface designs, and other essential system elements. The block diagram of the system implemented in this research can be seen in Figure 3.



Figure 3. System Block Diagram

Database

Furthermore, related to the database in Figure 4, there are six tables that the author describes in Table 2 below:

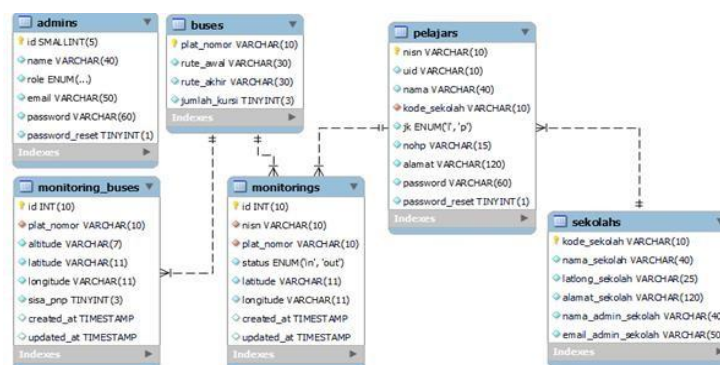


Figure 4. Tables in the Database

Table 2. Database Table Explanation

No.	Table Name	Column Name, Data Type, Length	Explanation
1	admins	id ☐ data type small integer, maximum five digits	Primary key, unique identity of each different admin account
		name ☐ string data type, maximum 40 characters	Admin account name
		role ☐ enum data type, content: 'super', 'admin', 'admin_sekolah'	The actor admin account, the account from the school's table, will be assigned the role admin_sekolah
		email ☐ string data type, maximum 50 characters	Email the admin account to log in to the system
		password ☐ string data type, maximum 60 characters	Admin account password to login to the system

Internet of Things-Based Bus and Student Monitoring System on Free School Transportation Madiun City

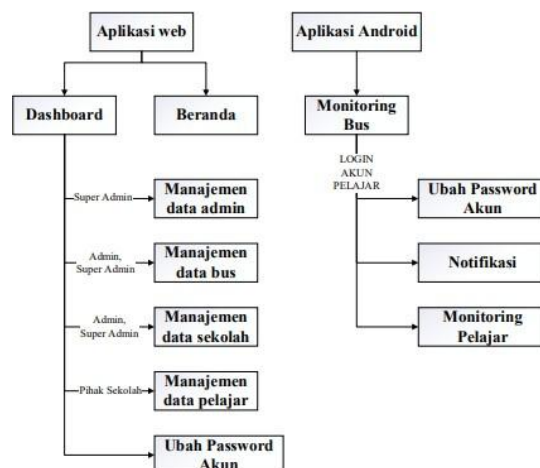
Rosyadi, Puspaningrum, and Hakimi

		password_reset ☐ data type tiny integer, maximum one digit	If the content is 1, the default password is still used; if 0, the password has been changed.
2	buses	plat_nomor ☐ string data type, maximum 10 characters	Primary key, unique number plate to initialize each bus in the system
		rute_awal ☐ string data type, maximum 30 characters	Bus departure location name
		rute_akhir ☐ string data type, maximum 30 characters	Bus end location name
		jumlah_kursi ☐ data type tiny integer, maximum three characters	Number of seats for bus passengers
3	sekolahs	kode_sekolah ☐ string data type, maximum 10 characters	Primary key, unique code to initialize the school in the system
		nama_sekolah ☐ string data type, maximum 40 characters	School name
		latlong_sekolah ☐ string data type, maximum 25 characters	School location point, contains latitude, longitude
		alamat_sekolah ☐ string data type, maximum 120 characters	School location address
		nama_admin_sekolah ☐ string data type, maximum 40 characters	The name of the school admin is the manager of student data in the system, and this data is automatically registered in the admin's table.
		email_admin_sekolah ☐ string data type, maximum 50 characters	School admin email to log in to the system, this data is automatically registered in the admins table with the role admin_sekolah
4	pelajars	nisn ☐ string data type, maximum 10 characters	Primary key, the national student identification number of each student
		uid ☐ string data type, maximum 10 characters	Unique identification of different student RFID tags
		nama ☐ string data type, maximum 40 characters	Student name
		kode_sekolah ☐ string data type, maximum 10 characters	<i>Foreign key</i> of kode_sekolah of the school table, initialize the school origin of the student
		jk ☐ enum data type, content: 'l', 'p'	Gender, male or female
		nohp ☐ string data type, maximum 15 characters	Student's parent's mobile number

5	monitorings	alamat □ string data type, maximum 120 characters	Student's home address
		password □ string data type, maximum 60 characters	Student account password to login to the Android app
		password_reset □ data type tiny integer, maximum one digit	If the content is 1, it means that the default password is still used; if 0, the password has been changed.
		id □ integer data type, maximum 10 digits	Primary key, the identity of each data record
		nisn □ string data type, maximum 10 characters	Foreign key of the student table nisn, monitoring the initialization of the student
		plat_nomor □ string data type, maximum 10 characters	Foreign key of the buses table plat_nomor, initialization of the monitoring of the buses
		status □ enum data type, content: 'in', 'out'	Determines whether the learner is in or out when monitored
		latitude □ string data type, maximum 11 characters	Student latitude point
		longitude □ string data type, maximum 11 characters	Longitude of the student
		created_at □ timestamp data type	Date, month year, hour data created
		updated_at □ timestamp data type	Date month year hour data modified
6	monitoring_buses	id □ integer data type, maximum 10 digits	Primary key, the identity of each data record
		plat_nomor □ string data type, maximum 10 characters	Foreign key of the buses table plat_nomor, initialization of the monitoring of the buses
		altitude □ string data type, maximum seven characters	Bus height position in meters above sea level
		latitude □ string data type, maximum 11 characters	Bus latitude point
		longitude □ string data type, maximum 11 characters	Bus longitude point
		sis_pnp □ data type tiny integer, maximum three digits	Remaining seats after being filled by bus passengers
		created_at □ timestamp data type	Date, month, year, hour data created

Application Navigation Structure

Figure 5. Application Navigation Structure



The design of the navigation structure in Figure 5 above describes two application structures, namely web and Android. The web application navigation structure has the dashboard and the home pages. On the dashboard page, several menus can be accessed according to the actor who logged in. All actors can access the account password change page. The admin data management page can only be accessed by super admin actors. The bus data and school data management pages can be accessed by super admin and admin actors. Meanwhile, school actors can only access the student data management page.

Then, in the Android application's navigation structure, there is a bus monitoring page. When the Android application user has logged in to the student account, they will get access to the student account password change feature, notification feature, and student monitoring feature.

After completing the design phase, the next stage was system implementation. This involved assembling and programming the Internet of Things (IoT) devices, which required careful attention to avoid damaging components. The initial step involved soldering several modules that lacked pin headers. The writer then connected each module to the NodeMCU using jumper cables to ensure they functioned correctly. The process began with the GPS module, where coding was performed to operate the NodeMCU and display GPS output on the Arduino IDE Serial Monitor. After successful testing, the same steps were repeated for the buzzer, RFID, and LCD 1602 modules.

Once each module was successfully implemented with the NodeMCU, the writer connected multiple modules simultaneously. The first test combined the GPS and buzzer modules, triggering the buzzer for 0.5 seconds whenever latitude and longitude data were detected. The setup was expanded to include the LCD 1602 module to display GPS coordinates. All three modules successfully operated simultaneously.

Challenges arose when integrating the GPS and RFID modules, as running them simultaneously proved difficult. After researching solutions online, the issue was resolved by implementing the

millis() function. This function allows the internal clock to run independently, enabling the GPS module to continue processing data while the RFID module operates seamlessly.

The web and Android systems were also programmed during the IoT programming stage. The web application uses Visual Studio Code, while the Android application is programmed using Android Studio.

The following phase was system testing, which employed a black-box testing method to evaluate the functionality of the developed system. Finally, the deployment phase was conducted, in which the web application was installed on a server, and the Android application was prepared for use by being deployed to Android devices.

The development of the bus and student monitoring system is divided into three parts: (a) the IoT device, (b) the web application, and (c) the Android application.

The IoT device is placed inside the school bus near the middle entrance. The HTTP protocol sends data collected from the GPS and RFID modules to the web server. The physical form of the IoT device is seen in Figure 6 below.



Figure 6. The Physical of IoT Devices

Currently, data from the IoT device is transmitted to the server using the HTTP protocol, which does not provide end-to-end encryption. Although the hosting server already supports HTTPS via Let's Encrypt SSL, the IoT device relies on HTTP due to firmware limitations and hardware compatibility. This condition poses a potential security risk, such as data interception by unauthorized parties. Therefore, enhancing security measures, such as full support for HTTPS or additional encryption for transmitted data, should be considered for future development.

The web application consists of a front-end interface that does not require account validation and a back-end interface that requires account validation. The back end is used for monitoring and managing admin, bus, school, and student data, which the admin can only access. The web application also processes data received from the IoT devices, which is then stored in a MySQL database. MySQL is one of the most widely used software for database servers, MySQL is open source and uses SQL. (Putra et al., 2020). The web application view can be seen in Figure 7 and 8.

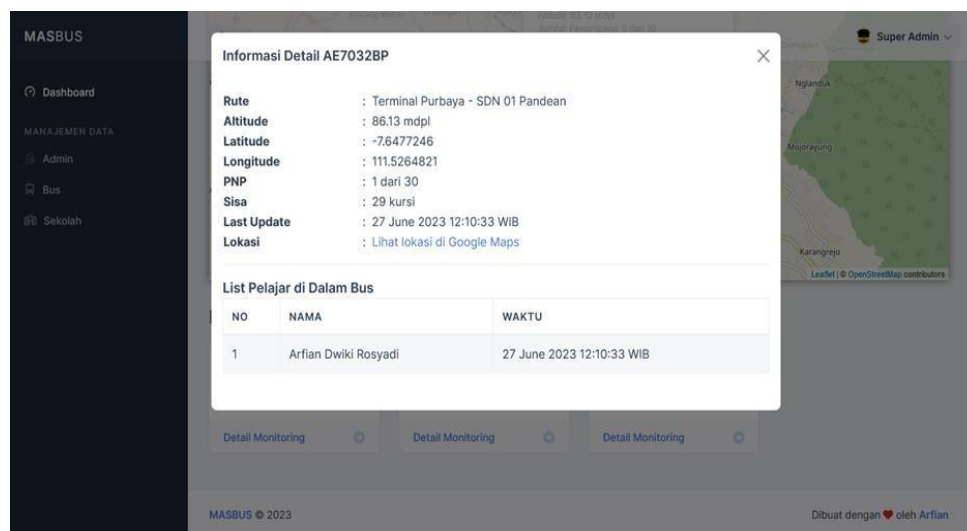


Figure 7. Web View for Monitoring Bus Detail Information

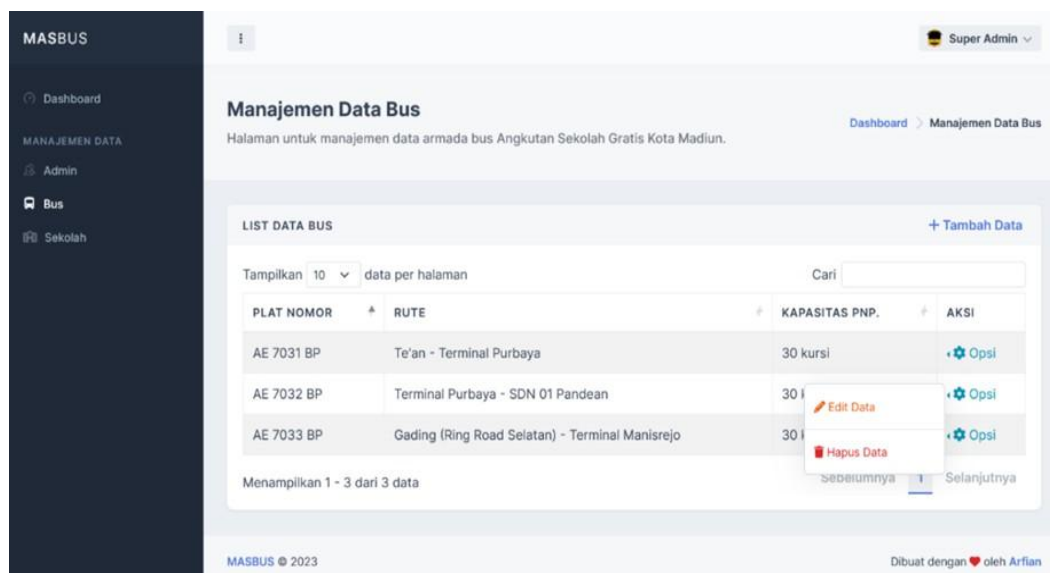


Figure 8. Web View for Bus Data Management

The Android application acts as an extension of the front-end functionality of the web application, displaying the map of bus and student locations as shown in Figure 9. To allow users to monitor students, account validation is required for students who are already registered in the monitoring system. Location data of buses and students is retrieved from the web server via an API. When a student taps their RFID tag on the IoT device, the system sends a notification to users with validated student accounts, informing them whether the student is boarding or has alighted from the school bus.

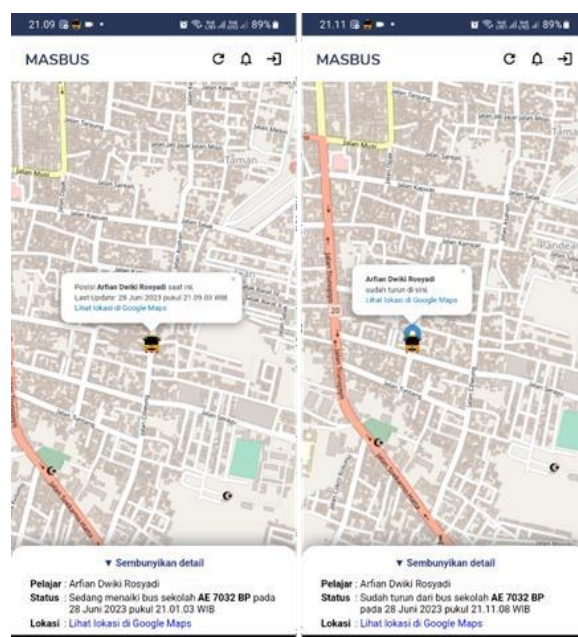


Figure 9. Android Application View of Bus and Student Monitoring System

This research was tested using the black-box testing method. Black-box testing can be interpreted as a test that determines whether all software functions have run properly according to the functional requirements that have been defined. (Fahrezi et al., 2022).

Black-box bus and student monitoring system testing was carried out directly on free school transportation in Madiun City. The test involves three scenarios: IoT device testing, web system testing, and Android system testing. Based on the test results, it can be concluded that the IoT device and the bus and student monitoring system on the web and Android platforms are functioning properly and ready for use.

Recommendations for Future Research

For future improvements, enhancements to the physical design of the IoT device could improve its aesthetic appeal, such as neatly perforating the project box for better cable management. Additionally, if feasible, the device should be able to temporarily store GPS data when Wi-Fi access is lost, ensuring continuous location tracking.

From a security perspective, data transmission should be fully migrated to HTTPS to enhance protection against interception. If the IoT hardware does not support HTTPS, an alternative solution could involve encrypting data payloads before transmission. Furthermore, implementing

token-based authentication (such as JWT or API keys) could prevent unauthorized access to the system's API.

For system functionality, additional features such as bus and passenger analysis could be developed to support the Madiun City Transportation Department's reporting needs. Furthermore, the system could incorporate bus status tracking (indicating whether a bus is in operation or parked) and utilize third-party notifications, such as Firebase Cloud Messaging (FCM), to improve real-time communication with users.

CONCLUSION

An Internet of Things (IoT) device product with a system for monitoring buses and students in Madiun City Free School Transport has been successfully made. This IoT device product can monitor buses and students in real-time using GPS modules as location information and RFID as a data information reader for each student with an RFID tag. Data from the IoT device is sent to the web server via API as an HTTP protocol, then stored in the MySQL database. The data stored in the database is then forwarded to the Android application via API so that it can be seen and monitored by users, namely parents of students in real-time. In addition, student parents will also get notifications automatically when students tap RFID tags on IoT devices on the school bus.

REFERENCE

- Alm, J., Buschman, R. D., & Sjoquist, D. L. (2014). Foreclosures and local government revenues from the property tax: The case of Georgia school districts. *Regional Science and Urban Economics*, 46(1), 1–11. <https://doi.org/10.1016/j.regsciurbeco.2014.01.007>
- Anjali, C. N., & Lutfi, M. (2023). Rancang Bangun Sistem Otomatisasi Pergantian Air Kolam Budidaya Ikan Nila Berdasarkan Parameter Kondisi Air Menggunakan Nodemcu ESP8266. *INFORMAL: Informatics Journal*, 8(2), 128. <https://doi.org/10.19184/isj.v8i2.41478>
- Asriyadi, F., & Ciksadan. (2022). *Rancang Bangun Sistem Keamanan Portable Menggunakan GPS dan RFID Berbasis NodeMCU*. Elektronika Kendali Telekomunikasi Tenaga Listrik Komputer.
- Chen, X., Ding, J., & Lu, Z. (2022). A Decentralized Trust Management System for Intelligent Transportation Environments. *IEEE Transactions on Intelligent Transportation Systems*, 23(1), 558–571. <https://doi.org/10.1109/TITS.2020.3013279>
- Daniels, J. P., & von der Ruhr, M. (2014). Transportation Costs and US Manufacturing FDI. *Review of International Economics*, 22(2), 299–309. <https://doi.org/10.1111/roie.12110>
- Divito, D. M., Budi, A. S., & Setiawan, E. (2022). *Implementasi Finite State Machine pada Sistem Notifikasi Pesanan Food Court*. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer.
- Efendi, Y. (2018). Internet of Things (IoT) Sistem Pengendalian Lampu Menggunakan Raspberry Pi Berbasis Mobile. In *Jurnal Ilmiah Ilmu Komputer* (pp. 19–26).
- Fahrezi, A., Salam, F. N., Ibrahim, G. M., Syaiful, R. R., & Saifudin, A. (2022). Pengujian Black Box

- Testing pada Aplikasi Inventori Barang Berbasis Web di PT. *AINO Indonesia. Jurnal Ilmu Komputer dan Pendidikan*, 2.
- Fitriana, S. (2022). *Rancang Bangun Sistem Monitoring Persediaan Barang pada SP Part Lampung Berbasis Web*. *Jurnal Ilmu Data*.
- Haryanto, K., & Wiranata, P. D. (2022). Sistem Monitoring Lokasi Siswa Menggunakan GPS Ublox Neo-6M di SD Muhammadiyah 1 Surakarta. In *JURNAL ILMLAH STMIK AUB* (pp. 138–150).
- Kurniawati, & Badrul, M. (2021). Penerapan Metode Waterfall Untuk Perancangan Sistem Informasi Inventory pada Toko Keramik Bintang Terang. *Jurnal PROSISKO*, 48.
- Kuswanto, J., Miftahul, W. A., & Asharudin, F. (2023). Smart Fish Farm Budidaya Ikan Nila Menggunakan NodeMCU Terintegrasi Berbasis Internet Of Things. *Smart Comp: Jurnalnya Orang Pintar Komputer*, 12(1). <https://doi.org/10.30591/smartcomp.v12i1.5061>
- Lee, L. H., Yoon, Y. J., Kim, D., Noh, H., Jones, S., & Lee, H. Y. (2023). Perceived transportation barriers moderate the association between depressive symptoms and household transportation use: A pilot study. *Journal of Transport and Health*, 33. <https://doi.org/10.1016/j.jth.2023.101713>
- Madiun, B. P. S. K. (2021). *Hasil Sensus Penduduk 2020*. <https://madiunkota.bps.go.id/pressrelease/2021/01/21/120/hasil-sensus-penduduk2020.html>
- Nasution, M. N., & Maulana, R. (2024). Pengembangan Aplikasi Sistem Informasi Akademik Berbasis Web Menggunakan Framework Laravel: Studi Kasus di SMK Assalam Depok. *Jurnal Informatika Terpadu*, 158.
- Nurani, A. A., Musnansyah, A., & Syamsi, D. (2022). Sistem Monitoring Parameter Hidroponik Rumahan Menggunakan Nodemcu Dengan Metode Prototype Home Hydroponic Parameter Monitoring System Using Nodemcu With Prototype Method. *E-Proceeding of Engineering*, 733.
- PARULIAN, D., Widodo, B., Stepanus, & Tobing, T. (2023). SISTEM PENGENDALIAN DAN MONITORING KUALITAS AIR TAWAR SERTA PEMBERIAN PAKAN OTOMATIS PADA BUDI DAYA IKAN NILA MENGGUNAKAN NODEMCU ESP-12F. *Lektrokom : Jurnal Ilmiah Teknik Elektro*, 4(1). <https://doi.org/10.33541/lektrokom.v4i1.4201>
- Putra, F. D., Riyanto, J., & Zulfikar, A. F. (2020). Rancang Bangun Sistem Informasi Manajemen Aset pada Universitas Pamulang. *Journal of Engineering, Technology & Applied Science*, 39.
- Ramadhani, R. F., & Mukhaiyar, R. (2020). Penggunaan Database Mysql dengan Interface PhpMyAdmin sebagai Pengontrolan Smarthome Berbasis Raspberry Pi. *Jurnal Teknik Elektro Indonesia*, 130.
- Ulfatiyana, M., & Parwiyanto, H. (2020). Difusi Inovasi Pelayanan Program Angkutan Sekolah

- Gratis (ASG) di Kota Madiun. *Jurnal Administrasi Publik*, 130.
- Wahyuni, A. S. (2020). *Evaluasi Kinerja Bus Sekolah Gratis di Kota Madiun*. Terminal Bus Purboyo, Sub Terminal Manisrejo, Jl.Niti Kusumo). Diploma thesis.
- Y., A. H., & Rahayu, S. (2022). *Rancang Bangun Aplikasi Penentuan Status Gizi Balita Berbasis Android*. Seminar Nasional Teknologi dan Riset Terapan.
- Yanez, W., Mahmud, R., Bahsoon, R., Zhang, Y., & Buyya, R. (2020). Data Allocation Mechanism for Internet-of-Things Systems With Blockchain. *IEEE Internet of Things Journal*, 7(4), 3509– 3522. <https://doi.org/10.1109/JIOT.2020.2972776>
- Yin, C., Lu, Y., Xu, X., & Tao, X. (2020). Railway freight subsidy mechanism based on multimodal transportation. *Transportation Letters*, 1–12. <https://doi.org/10.1080/19427867.2020.1791507>
- Zanofa, A. P., & Fahrizal, M. (2021). Penerapan Bluetooth Untuk Gerbang Otomatis. *Jurnal Portal Data*, 4.