# Design and Build a Monitoring Application for Android Based Poltekad Weapons Warehouse Using Generative AI (Artificial Intelligence)

Kharis Setiaji[1)], Imam Ashar[2)], and Choirul Rio Prabowo[3).]
[1) 2) 3)] Military Telecommunication Engineering Study Program, Politeknik Angkatan Darat Jl.Raya Anggrek No.1 Junrejo, Batu, Indonesia
E - mail : setiajikharis@gmail.com [1)], imamasharstmt@gmail.com [2)], choirul.rio.p@gmail.com [3).]

***Abstract:*** *The management of the weapons warehouse within the Army Polytechnic (Poltekad) plays a crucial role in maintaining personnel security and training readiness. However, a persistent obstacle is the manual or semi-digital recording process. This slows down data collection, makes it prone to errors, and makes it difficult for officers to quickly submit reports to superiors. Based on these issues, this research developed a solution in the form of an Android-based monitoring application equipped with generative AI technology. The goal is simple: to facilitate real-time and intelligent weapons monitoring. Using a prototype development method, this system is designed not only to record incoming and outgoing weapons data but also utilizes AI to automatically compile daily narrative reports. The end result is that this application is proven to be effective, weapons stock data is more accurate, and administrative work for officers is significantly reduced thanks to the AI's automatic reporting feature.*
*Keywords: Monitoring, Armory, Android application, Generative AI*

## INTRODUCTION

The Army Polytechnic (Poltekad) plays a crucial role in producing soldiers who are not only physically resilient but also proficient in defense technology. In military education like Poltekad, logistics management, particularly the management of weapons depots, plays a vital core element in ensuring smooth and effective operations. Weapons depots are not simply storage areas for assets, but also vital points that demand the highest security standards and precise data accuracy. Weapons readiness directly impacts the smooth running of education, training, and the operational readiness of units. However, the reality on the ground often presents significant challenges. To date, mechanisms for monitoring the entry and exit of weapons still rely heavily on conventional methods. Officers on duty or in the logistics department often have to manually record weapon serial numbers, borrower names, and return dates in unintegrated logbooks or spreadsheets. This manual work method has significant weaknesses, ranging from slow data recapitulation to a high risk of human error due to officer fatigue. As a result, presenting

accurate data in real time to superiors is difficult.

The need to modernize this system aligns with developments in information technology research over the past three years. Various literature reviews highlight the urgency of digital transformation in asset management. As revealed in a recent study, the use of mobile (Android)-based applications integrated with digital systems has been shown to significantly reduce inventory inspection time compared to manual methods and increase the validity of asset data in the field (Rasyid Ridha & Supriatna, 2023). Furthermore, developments in Artificial Intelligence (AI) technology also offer new breakthroughs. Another study confirmed that the integration of Generative AI into management systems serves not only as a question-and-answer tool but also automates the creation of complex narrative and routine reports, drastically reducing the human administrative burden (Paulose & Neelanath, 2024). Although both technologies are mature, their integrated application in a specific military security environment like the Army Training Center (Poltekad) remains largely unexplored.

Departing from the gap between manual problems in the field and potential technological solutions, this research focuses on a key problem formulation, namely how to design an effective weapons warehouse monitoring application to replace manual recording, and how to integrate Generative AI intelligence so that the system can process transaction data into accurate, automated reports. Answering these problems, this research aims to produce a prototype of an Android-based application that can present weapons availability data in real-time and transparently. Furthermore, the goal of this research is to implement intelligent features based on Generative AI to assist officers in compiling daily reports and warehouse activity analysis without the need for lengthy typing, thereby minimizing human error.

Through the development of this system, the author has high hopes for the benefits that can be felt, both practically and academically. For the Poltekad institution, this application is expected to be a real tactical solution, where the administrative burden of warehouse officers can be reduced and leaders can monitor logistics readiness at any time with valid data. Meanwhile, academically, this research is expected to enrich the treasure trove of Informatics Engineering knowledge by proving that Generative AI can be implemented safely and effectively in sensitive real-world cases such as military asset management, going beyond its general use which is limited to creative content creation.

## RESEARCH METHODS

The method used in this research is an experimental method with a Research and Development (R&D) prototype approach. This method was chosen because it allows for direct testing through the implementation of a specially designed program hypothesis, namely an Android application for monitoring armory that is integrated in real time with a website API. This allows the system to be tested iteratively to achieve the desired result: accurate data synchronization between the RFID/Hikvision hardware and the display on the mobile device (Gunawan et al., 2024).

This study uses two main variables: independent and dependent. The independent variables are the implementation of the REST API architecture in the Android application and the Generative AI algorithm run in the program for text analysis. The dependent variables are the response speed of weapons stock data (latency) and the accuracy of the generated automated report narrative. This study is expected to provide insight into the extent to which the integration of artificial intelligence in military logistics systems can improve reporting efficiency without compromising strict data security standards (Putrie et al., 2025).

### Research Components

The Android (Java) application is the main user interface media that functions to display visualization of weapon stock data and is a gateway for interaction with intelligent assistants (Alpiansyah et al., 2023).

REST API (Application Programming Interface) functions as a bridge for exchanging JSON data between mobile applications and warehouse servers that store data from RFID and facial recognition cameras (Wahyudi et al., 2025).

Generative AI Service is a natural language processing (NLP) component that is tasked with processing raw transaction log data into standard military report sentences (Cusumano et al., 2024)

Database Server (MySQL) is a centralized data storage that receives input from hardware (RFID Reader & Hikvision) before being distributed to the application (Sami et al., 2024).

FCM (Firebase Cloud Messaging) serves as a real-time notification delivery service from the server to users' Android devices, allowing important information such as borrowing status, anomalies, and daily reports to be received automatically without requiring the application to be opened (Widodo et al., 2025).

The testing device is an Android smartphone with certain minimum specifications and a stable internet connection to test data retrieval from the server (Bintang Sadewa et al., 2024).

### System Planning

The system design in this study aims to establish a transparent arsenal monitoring mechanism by utilizing data from a third party (warehouse server) and reprocessing it using artificial intelligence. This system is composed of several key interconnected components, starting with physical sensor readings, data synchronization via API, and finally visualization on the user's mobile screen (Ryzki et al., 2024)

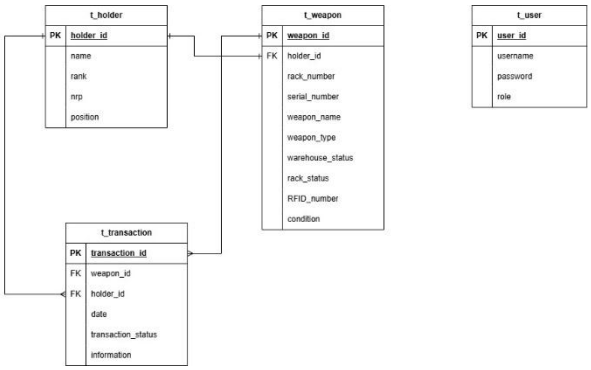


Figure 1. Entity Relationship Diagram (ERD)

The diagram illustrates the ERD of the armory monitoring system, which consists of four main tables: t_holder, t_weapon, t_transaction, and t_user. The t_holder table stores personnel identity data such as name, rank, nrp, and position, while the t_weapon table contains weapon inventory information including serial_number, weapon_type, rack_number, warehouse_status, RFID_number, and condition, which are linked through holder_id. All borrowing and return activities are recorded in the t_transaction table using the relationships weapon_id and holder_id, allowing weapon movement history to be systematically tracked. Meanwhile, the t_user table is used for application authentication by storing username, password, and role. This relational structure ensures that personnel, weapon, and transaction data are well integrated, supporting accurate, controlled, and secure armory inventory management (Wahyudi et al., 2025).

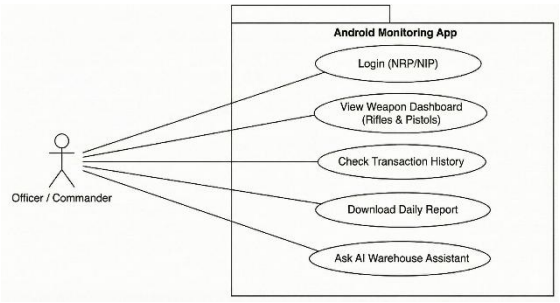


Figure 2. Usecase Diagram

The data validation process is performed by matching physical weapons with the data displayed in the application. This system is designed to minimize human error by automating record-keeping. Only authenticated users with a username and password can access critical features such as downloading reports and transaction history, ensuring data integrity and security in accordance with military standards.

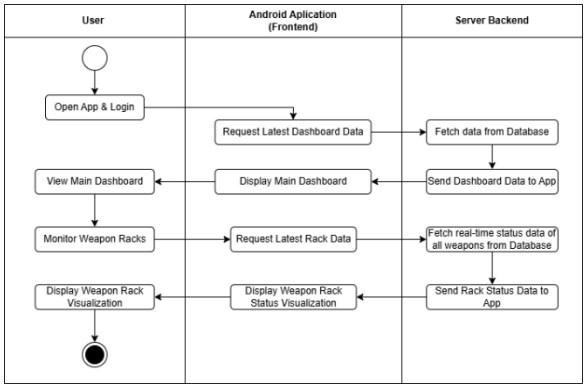


Figure 3. Activity Diagram Weapon Racks

The figure 3 illustrates the Activity Diagram of the armory monitoring system, showing the interaction flow between the user (Commander, Pasipam, and warehouse personnel), the Android application as the frontend, and the server backend. The process begins with user login as an authentication step, followed by the application requesting the latest dashboard data from the server. The server retrieves inventory data from the database and sends it back to be displayed on the main dashboard. Next, the user accesses the weapon rack monitoring feature, the application requests the latest rack data, and the server returns the real-time status of all weapons. This information is then visualized within the application, allowing users to monitor armory conditions directly without manual refresh, after which the system remains ready to process subsequent requests (Trisna Putra, 2025).

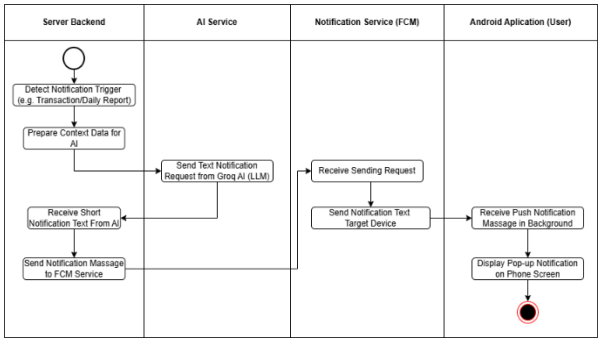


Figure 4. Activity Diagram Push Notification

The figure 4 illustrates the Activity Diagram of a smart notification system based on Generative AI involving the backend server, AI service, Firebase Cloud Messaging (FCM), and the Android user application. The flow begins when the server detects important events such as transactions or daily reports, then prepares contextual data and sends it to Groq AI to generate narrative notification text. The AI-generated output is returned to the server and forwarded to the FCM service, which subsequently distributes the message to users' Android devices. The application receives the push notification in the background and displays a pop-up on the smartphone screen, allowing information to be delivered in real time without manual intervention, thereby supporting responsive and efficient weapon warehouse monitoring (Nasution & Kamil Siregar, 2023).

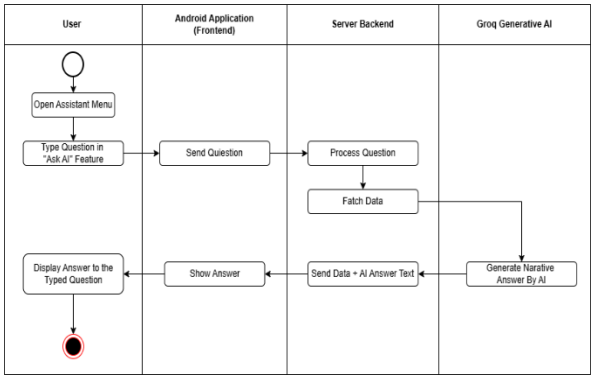


Figure 5. Activity Diagram Chatbot AI

The figure 5 illustrates the Activity Diagram of the Intelligent Assistant (AI Chatbot) feature, describing the interaction flow between the user (Commander, Pasipam, and warehouse personnel), the Android application, the backend server, and the Generative AI module. The process begins when the user accesses the Assistant menu and submits a question through the application. The server then retrieves relevant data from the database and forwards it to the Generative AI module for analysis. The AI generates a narrative response, which is sent back to the Android application and displayed to the user. This workflow is designed to operate automatically and in real time, enabling users to obtain warehouse condition information quickly without manual data searching (Agersi Diah Anggraini, 2024).

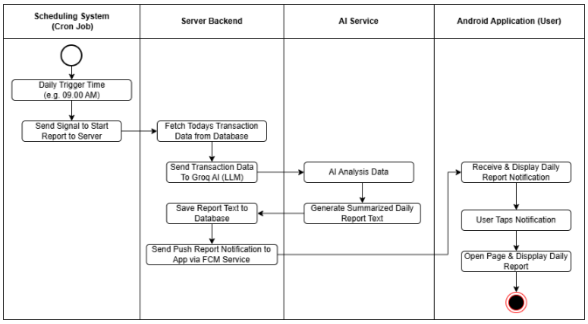


Figure 6. Activity Diagram Daily Report

The figure presents an Activity Diagram of the automated daily report generation process based on Generative AI, involving the scheduling system (cron job), backend server, AI service, and Android application. The process begins with a daily time trigger that instructs the server to retrieve transaction data from the database, which is then sent to Groq AI for analysis and summarization into a narrative report. The generated report is stored on the server and delivered as a push notification via Firebase Cloud Messaging (FCM) to users' devices. The Android application receives the notification and displays the full daily report when opened, enabling commanders and personnel to access real-time armory condition summaries without manual intervention (Tresnawati et al., 2024).

## RESEARCH RESULT

This chapter presents the results of the prototype development phase of the POLTEKAD weapons warehouse monitoring application. The discussion includes user

interface implementation, system functional testing results, and an analysis of the performance of the API integration and generative AI features.

**System Implementation**

Based on the system design, an Android application was developed as a monitoring client for the armory system, retrieving inventory and transaction data in real time from the central server via REST API in JSON format without local storage. All core processing is handled on the backend, including database management and integration with Generative AI through the Groq AI API (LLM) to automatically generate narrative reports from transaction data. The AI-generated results are returned to the server for storage and distribution to the application. With this architecture, the Android application serves as a lightweight interface displaying the monitoring dashboard, daily reports, FCM notifications, and AI chatbot responses, enabling armory information to be delivered in an integrated, automated, and real-time manner without manual record-keeping (Pambudi et al., 2025).

**Login (Authentication) Page**

This page serves as the main security gateway of the system. Due to the sensitivity of military data, the system is designed to grant access only to personnel registered in the established database. Users are required to enter a valid username and password to access the system.

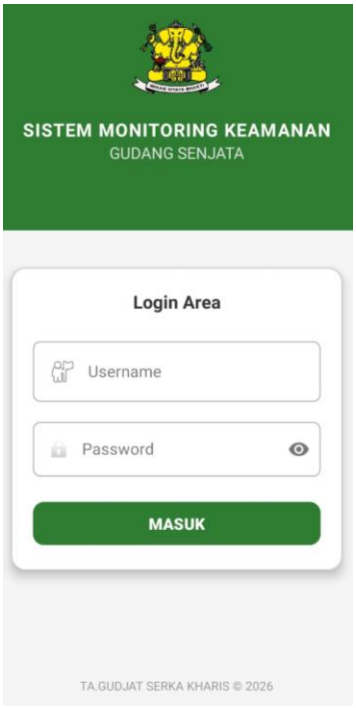


Figure 7. Application Login Page Display

**Main Monitoring Dashboard Page**

After login, users are directed to the Dashboard page as the central monitoring interface for the armory system. The system automatically retrieves JSON data from the server to display real-time inventory statistics in two main categories, Long-Barrel Rifles and Pistols, with indicators for Total, In Warehouse, and Checked Out that update dynamically during RFID scans. The dashboard also provides access to the Ask AI Warehouse Assistant feature, Daily Reports, Transaction History, and Weapon Rack Monitoring to view weapon positions in real time. Through this integrated interface, users

can monitor inventory and warehouse activities efficiently without manual record-keeping.


Figure 8. Monitoring Dashboard Display

**AI Assistant Feature Integration**

The flagship feature of this research is the integration of Generative AI, placed at the top of the dashboard in the form of a search bar labeled "Ask the AI Warehouse Assistant...". This feature allows officers or commanders to ask natural questions (for example, "Create a summary of weapons issued today") and receive narrative answers processed by AI based on recent transaction data.


Figure 9. AI Assistant Display

**Monitor Weapon Racks**

The figure 10 displays the Weapon Rack Map page, which visualizes the position of weapons on Weapon Rack 1 (slots 1–22) in real time. Each box represents a rack slot, with white indicating available weapons and serial numbers, while red indicating "Not Available" indicates weapons on loan or out of stock. This visualization allows officers and managers to directly monitor weapon availability and position without physical inspection, improving the speed of monitoring and accuracy of inventory data.

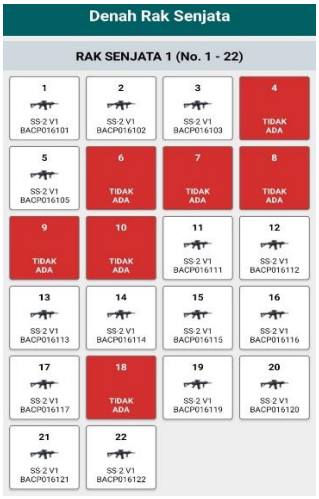


Figure 10. Weapon Racks

### Daily Report

The figure 11 displays the Daily Report page, which summarizes the weapons inventory status on a specific date. Information is displayed in four key indicators: Total Assets, the number of weapons issued today, the number returned, and the number not yet returned. Additionally, a Print Report (PDF) button allows users to download the daily report in official document format. This view helps leaders and officers obtain a quick overview of weapons status for daily operational evaluations.


Figure 11. Daily Report

### System Test Results

To ensure the application runs according to user specifications, testing is performed using the Black Box Testing method. This testing focuses on the application's input and output functionality without looking at its internal code. The following is a summary table of the main functional test results:

Table 1. Black Box Testing Result

| No | Tested Featur | Test Scenario | Input | Expected Output | Result |
|---|---|---|---|---|---|
| 1. | Login | User enters valid username & password | Correct username & Password | Successfully redirected to Dashboard page | Valid |
| 2. | Login | User enters invalid credentials | Incorrect username or password | Login failure message displayed | Valid |
| 3. | Dashboard | System loads inventory data | - | Weapon statistics displayed (Total, In Warehouse, Checked Out) | Valid |
| 4. | Weapon Rack Monitoring | User opens rack monitoring menu | - | Real-time rack visualization displayed | Valid |
| 5. | Daily Report | User opens daily report page | - | Daily summary displayed | Valid |
| 6. | AI Chatbot | User submits a question | Text query | Narrative response displayed | Valid |
| 7. | Push Notification | Transaction or report occurs | - | Notification appears automatically | Valid |

**DISCUSSION**

This research aims to improve the efficiency and accuracy of armory inventory management by developing an Android based monitoring system integrated with a backend server and Generative AI services. Based on the implementation and testing results black box Based on the work that has been done, this section discusses the functional implications of the system in answering the research problem. Testing was conducted using an input and output validation approach without reviewing the internal code structure, thus focusing on the suitability of the system's functions to operational needs.

Key findings were obtained from a series of tests on the application's core features, including login authentication, inventory dashboard loading, weapon rack monitoring, daily reports, AI chatbots, and automated notifications. All test scenarios yielded "Valid" results, meaning the system was able to respond to each input according to the expected output. The login feature successfully restricted access to only users with valid credentials, thus ensuring the system's initial security. The dashboard was proven capable of displaying inventory statistics (Total, In Warehouse, and Checked Out) based on the data in the database. The rack monitoring feature displayed real-time visualizations of actual conditions, while the daily report successfully presented a data summary without any functional errors.

Testing of the AI Chatbot feature demonstrated that the system was capable of accepting text-based queries and generating relevant narrative responses based on the available data. This confirmed that the Groq AI API (LLM) integration on the backend was running according to the architectural design. Furthermore, the push notification feature successfully displayed automatic notifications when transactions occurred or reports were generated, proving the integration with Firebase Cloud Messaging (FCM) was working properly.

Overall, the resultsblack box testingvalidated that the system architecture—which separates the roles of the Android application as a lightweight client and the backend server as the data processing center—functioned as designed. The system reduced reliance on manual record-keeping and provided inventory information quickly, integratedly, and responsively. The success of this testing demonstrated that the developed prototype met the functional operational needs of the armory and was suitable for use as an integrated technology-based monitoring solution.

## CLOSURE

Based on the research results, an Android-based armory monitoring system integrated with REST API, backend server, Firebase Cloud Messaging, and Generative AI Groq has been proven to improve inventory accuracy, speed of information presentation, and reporting efficiency. The separation of the application's role as a light client and the server as a processing center allows dashboards, shelf visualizations, real-time notifications, daily reports, and AI chatbots to run automatically without manual recording. The Generative AI integration successfully transforms transaction data into report narratives and contextual responses, thereby reducing personnel workload and accelerating leadership decision-making. Thus, the research objective of realizing a responsive, secure, and intelligent armory management system has been achieved, with opportunities for further development throughlocal caching, biometric authentication, and predictive analytics.

## BIBLIOGRAPHY

Agersi Diah Anggraini. (2024). Penerapan Artifical Intelligence Dalam Administrasi Pendidikan Untuk Meningkatkan Transparansi Dan Akuntabilitas Di Yayasan Pendidikan Garuda Khatulistiwa Kubu Raya. *Journal Islamic Global Network For Information Technology And Entrepreneurship*, *2*(3), 34–40. Https://Doi.Org/10.59841/Ignite.V2i3.1565

Alpiansyah, F., Abdillah, R., Himawan, I., Yulisa Geni, B., Bisri, M., Studi Teknologi Rekayasa Multimedia, P., Bhakti Kartini, P., & Artikel, R. (2023). Rancang Bangun Sistem Administrasi Perkantoran Berbasis Bahasa Pemrograman Java. *Jurnal Ilmiah Teknik Dan Ilmu Komputer*, *2*(1), 1–7. Https://Doi.Org/10.55123

Bintang Sadewa, R., Saragih, H., Lediwara, N., & Sadewa, R. B. (2024). Development Of Android-Based Weapon Monitoring Application In The Armory. In *Jurnal Mandiri It* (Vol. 13, Number 2).

Cusumano, M. A., Farias, V. F., & Ramakrishnan, R. (2024). *Generative Ai As A New Platform For Applications Development*. Https://Doi.Org/10.21428/E4baedd9.F189351f

Gunawan, D., Maulana, A., Alfarizi, S., Mulyawan, A. R., Ichsan, N., & Basri, H. (2024). *Implementasi Metode Prototype Dalam Perancangan Presensi Karyawan Berbasis Android Pada Pt Jedi Global Teknologi*. *4*. Https://Jurnal.Bsi.Ac.Id/Index.Php/Profitabilitas

Nasution, A., & Kamil Siregar, I. (2023). Monitoring Stok Barang Menggunakan Teknologi Push Notifikasi Android. *Journal Homepage*. Http://Jti.Aisyahuniversity.Ac.Id/Index.Php/Ajiee

Pambudi, A., Amelia, D., Komputer, J. I., Matematika, F., Ilmu, D., Alam, P., & Lampung, U. (2025). Agri Cuaca: Aplikasi Android Untuk Rekomendasi Kegiatan Pertanian Padi Berbasis Data Cuaca Dan Analisis Ai. *Https://Jurnal.Pustakagalerimandiri.Co.Id/Index.Php/Pustakadata*, *5*, 403–411. Https://Doi.Org/10.55382/Jurnalpustakadata.V5i2.1441

Paulose, R., & Neelanath, V. (2024). Generative Ai-Driven Automation Of Business Process Reimagination. *2024 Ieee Recent Advances In Intelligent Computational Systems (Raics)*, 1–6.

Https://Doi.Org/10.1109/Raics61201.2024.10689826

Putrie, *, Salsabilla, M., Salsabilla, P. M., Wolor, C. W., Negeri, U., Eka, J., & Utari, D. (2025). Analisis Pemanfaatan Artificial Intelligence (Ai) Dalam Otomasi Administrasi Perkantoran Di Pt Xyz. *Jurnal Ilmiah Ekonomi Manajemen Bisnis Dan Akuntansi*, *2*(4), 53–62. Https://Doi.Org/10.61722/Jemba.V2i4.1042

Rasyid Ridha, F., & Supriatna, I. (2023). Implementation Of Android-Based Sales Accounting Information System And Digital Cashier Application “Loyverse Pos” In Preparation Of Sales Reports (Case Study At Kramat Jaya Cimahi Shop). *Indonesian Accounting Literacy Journal*, *3*(3), 309–316.

Ryzki, *, Firmansyah, A., Sihaloho, O. W., Asdiana, F., Annas Amrullah, R., Arisusanty, D. J., & Turnip, G. I. P. (2024). Development Of Sps Inaportnet With Integration Web-Based Duty Officer Information At Tanjung Perak Surabaya Port. In *Times New Roman* (Vol. 5, Number 3). Https://Journal.Unhas.Ac.Id/Index.Php/Zonalautzonalaut

Sami, S. E. T., Rahmawati, S., Prasetyo, A., & Cahyono, C. (2024). Aplikasi Sistem Informasi Penjualan Pada Rumah Makan “Jeng Tin” Menggunakan Database Mysql. *Jurnal Janitra Informatika Dan Sistem Informasi*, *4*(1), 1–14. Https://Doi.Org/10.59395/Janitra.V4i1.178

Tresnawati, D., Fitriani, L., Kamal, M. A., & Setiawan, A. T. (2024). Pengembangan Desain Aplikasi Daily Report Dengan Pendekatan User Centered Design. *Jurnal Algoritma*, *21*(1), 141–151. Https://Doi.Org/10.33364/Algoritma/V.21-1.1525

Trisna Putra, A. (2025). Perancangan Sistem Monitoring Pengiriman Barang Berbasis Android Dan Website Pada Pt Sinergi Lintas Global. *Tekinfo*, *26*. Https://Doi.Org/10.37817/Tekinfo.V26i1

Wahyudi, R., Bush Henrydunan, J., Shaleh, A., Gaol, L., Amanah, F., Dly, R., & Perdana, A. (2025). Rancang Bangun Aplikasi Inventori Umkm Berbasis Flutter Dengan Integrasi Shared Preferences Dan Rest Api. In *Jurnal Mahasiswa Teknik Informatika)* (Vol. 9, Number 6).

Widodo, T., Santosa, H., & Styvani, M. (2025). Implementasi Push Notification Pada Aplikasi Antrean Pasien Berbasis Android. *Jsai : Journal Scientific And Applied Informatics*, *Vol. 08,No.3*, 644–649. Https://Doi.Org/0.36085