

Design and Development of a Fixed Asset Inventory System At The Holy Family Parish Church Atmodirono

Robertus Firenza Nugroho¹, Albertus Dwiyoga Widianoro², Andre Kurniawan Pamudji³

¹²³Department of Information Systems

¹²³Soegijapranata Catholic University, Semarang, Indonesia

¹firenzanugroho@gmail.com, ²yoga@unika.ac.id, ³andre.kurniawan@unika.ac.id

Abstract- The management of fixed assets at The Holy Family Parish Church Atmodirono is still carried out manually using Microsoft Excel, which causes various problems such as limited multi-user access, the risk of data duplication, recording errors, and difficulties in asset tracking and reporting. This study aims to design and build an integrated web-based fixed asset inventory system. The system development method used is the Waterfall method, which includes the stages of requirements analysis, system design, implementation, and testing. The system is built using the Laravel framework and MySQL database. The results of the study show that the developed system is able to improve the efficiency of asset management through centralized recording, asset tracking using QR Codes, and automatic reporting features.

Keywords: inventory system, fixed assets, laravel, waterfall, church.

I. INTRODUCTION

The Holy Family Parish Church Atmodirono is a religious institution that owns a large number of fixed assets. Asset management is an important function required by an organization or institution to determine the status and condition of its owned assets [1]. So far, the asset inventory process still relies on Microsoft Excel, although it is easy to access and use, it is still not efficient due to the large and complex number of parish assets. This condition causes several issues such as Excel files that can only be opened by one user at a time, which hinders collaboration among administrators from different teams, data

duplication, the risk of data loss if the device is damaged or the file is not saved properly, input errors, and difficulties in tracking the book value of assets over time.

A study conducted at the Salulemo Village Office showed that replacing Microsoft Excel with a dedicated inventory application resulted in more effective and accurate data processing, as well as improved document administration that previously often experienced errors [2]. Another example is the design of a web-based inventory system at ITSK Soepraen, where the new system successfully overcame the inefficiency of manual Excel recording, increased transparency, and facilitated data management with a user acceptance rate reaching 92.75% [3]. The implementation of a web-based inventory information system at CV Tunggal Putra Jaya proved an increase in operational efficiency of up to 92.73%, marked by a drastic reduction in stock monitoring time from 30 minutes to only 1 minute, as well as the acceleration of financial report generation from 10 minutes to 1 minute [4]. In addition, at CV Aqualux Duspha Abadi, it was shown that the implementation of an Android-based inventory system was able to overcome the problem of unsynchronization between the physical quantity of goods and manual paper-based recording data, with user satisfaction test results (User Acceptance Test) reaching above 90% [5]. A study shows that by integrating web technology, churches can improve communication, streamline operations, and strengthen relationships with congregations [6].

Based on this background, the problem formulation in this study is how to design a fixed asset inventory system that can replace

the role of Microsoft Excel at The Holy Family Parish Church Atmodirono and how the acceptance of the system by the Inventory Team and the Service Team. The objective of this study is to design a website-based fixed asset inventory system that replaces the role of Microsoft Excel in recording and managing assets at The Holy Family Parish Church Atmodirono and to evaluate it.

II. METHOD

The developed system functions as a centralized asset data management system to support the process of recording, tracking, and reporting fixed assets more effectively. Because an information system is a tool or system used to present data into information that is valuable to its users [7], and is a main component that is interrelated to create a system that can work together to transform data into useful information [8].

A. Data Collection Method

Data collection was carried out using three main techniques, namely literature study, interviews, and observation. The literature study was conducted by searching for various scientific journal references and references related to information systems, fixed asset management, and software development as the basis for system design.

Structured interviews were conducted with parties directly involved in the management of fixed asset inventory, namely the Inventory Team and the Service Team of The Holy Family Parish Church Atmodirono. The interviews were carried out in two stages, namely the initial stage to identify user requirements and the final stage after the system was fully developed to evaluate system acceptance. In addition, direct observation was conducted on the ongoing fixed asset management process, including the asset recording process and the implementation of stock opname. SO or Stock Opname is an important method in internal inventory control that aims to verify the quantity and value of inventory physically [9].

This observation aims to understand the existing workflow and identify problems and inefficiencies that can be addressed through the developed system.

B. System Development Method

The system development method used in this study is the waterfall method. This method was chosen because it is suitable for system development with requirements that have been clearly defined from the beginning. The waterfall method consists of the stages of software requirements analysis, system design, program code development, and testing [10].

At the software requirements analysis stage, system requirements are formulated based on the results of interviews and observations to ensure system conformity with user needs. The system design stage is carried out using Unified Modeling Language (UML), because Unified Modeling Language (UML) is a graphical-based visual language used to present specifications, design, and documentation in object-oriented system development [11]. The UML diagrams used consist of use case diagrams and activity diagrams which are used to provide an overview of system processes and user interactions.

Next is the implementation stage, where in this stage the system is built using the PHP programming language and the Laravel framework. PHP (PHP Hypertext Preprocessor) is a programming language that can be inserted into HTML. PHP is a server-side scripting programming language designed for web development [12], and Laravel was chosen because it is an open-source framework [13] that supports secure and structured web application development [14]. This system uses a MySQL database, which is a relational database management system and is capable of supporting data processing operations effectively [15].

C. System Testing and Acceptance Evaluation

After the implementation stage, the next step is the system testing stage which functions to ensure that all functions and the User Interface (UI) can function normally and in accordance with the initial design as well as to ensure that there are no bugs in the system.

After the system is tested, the system is then evaluated using a qualitative approach through interviews with six users who are directly involved in the operational inventory of parish assets, consisting of the Inventory Team and representatives of the Service Team. The interview results are then analyzed qualitatively to provide an overview of the user experience, system usability, and the benefits of the system in improving the efficiency of the fixed asset inventory process at The Holy Family Parish Church Atmodirono.

III. RESULTS AND DISCUSSION

The implementation of this system includes the design of the system flow, namely the flowchart for the main feature of adding item data starting from the request for adding items until the items are registered by the inventory team, use case diagrams, and the user interface (UI), which are then implemented using the Laravel framework and MySQL database.

A. Flowchart

Figure 1 below is a flowchart that illustrates the system procedure starting from the submission of requests by the Service Team to the verification and final recording process by the Inventory Team into the database. After the system logic flow has been created, the implementation results are then created into the application interface. Figure 1 below shows the flow of adding items, which begins when the Service Team selects the item request menu and clicks the “Submit New Request” button. The Service Team fills in request details such as item name, reason,

and estimated price, then submits them to the system. The system then validates the data, stores it in the item_requests table with a “Pending” status, and sends a notification to the Inventory Team. After that, the Inventory Team opens the request menu to review the submission details. If the request is rejected, the Inventory Team fills in the rejection reason and the system changes the status to “Rejected”. If approved, the Inventory Team clicks the “Approve” button and the status changes to “Approved”.

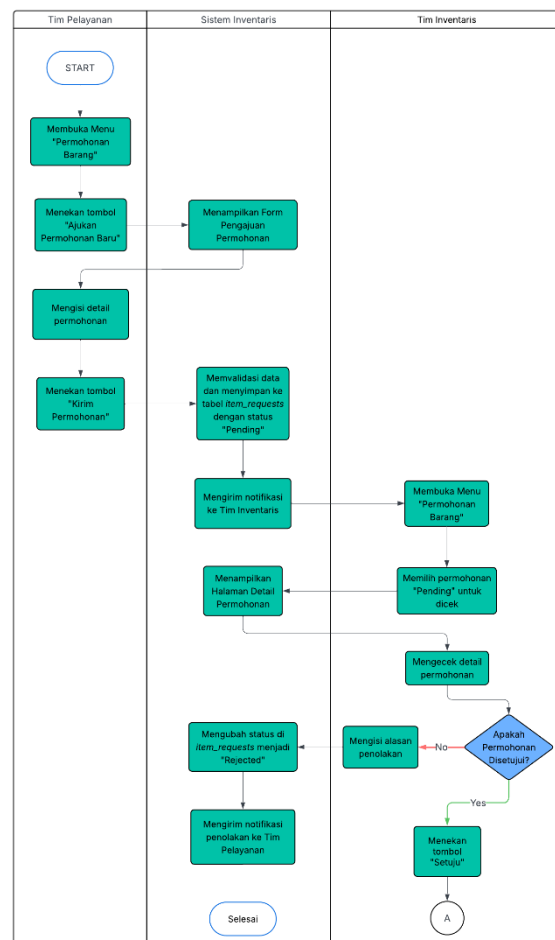


Figure 1. General Flowchart of the Fixed Asset Inventory System

Figure 2 shows the continuation of the process after Figure 1, which begins when the request has been approved. The Inventory Team opens the item data menu and clicks the “Add Data” button. The team fills in the complete form and clicks the “Save” button. The system then validates the data, stores the new asset data in the commodities table with an “Active” status, and records the data in the commodity_depreciations table. The process

is completed after the system displays the notification “New Asset Successfully Added”.

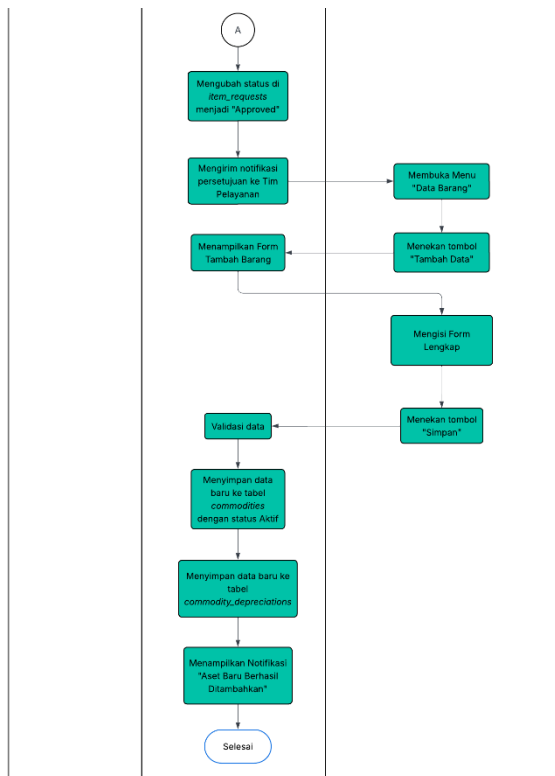


Figure 2. General Flowchart of the Fixed Asset Inventory System

B. Use Case Diagram

The use case diagram provides a visualization that connects components such as actors, use cases, and the relationships between them [16]. Figure 3 below shows the use case diagram involving two main actors, namely the Service Team and the Inventory Team. The Service Team submits item requests and monitors the status of submitted requests through the system, while the Inventory Team verifies requests, manages fixed asset data, carries out the stock opname process, and generates inventory reports.

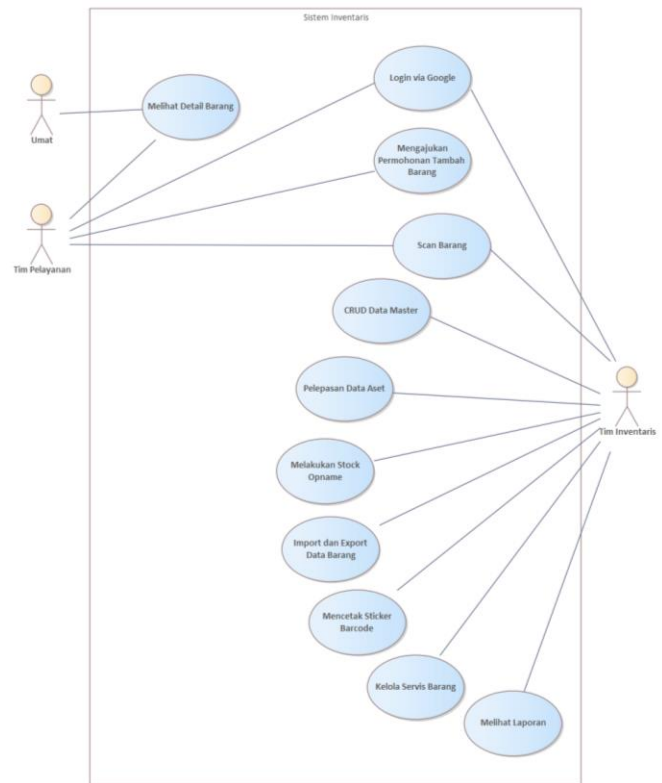


Figure 3. Use Case Diagram

C. ERD Diagram

Figure 4 below shows the ERD diagram for storing important data such as the commodities table, which stores the main data of each owned asset. In inventory control, there is a database capable of storing item data correctly and managing relationships between item data so that the resulting information is useful and beneficial in company work processes [17]. The commodities table is related to the commodity_locations table to determine asset locations or rooms, and the commodity_locations table is related to the gedung table for building locations. The commodities table is also related to the kategori_barang and jenis_barang tables.

There is also the commodity_depreciations table used to store depreciation calculation data, and the asset_disposals table to record data on disposed assets. The commodity_services table stores service history, and the commodity_stock_opname table records SO or Stock Opname results. The item request submission flow is stored in the item_requests table, which is connected to the users table.

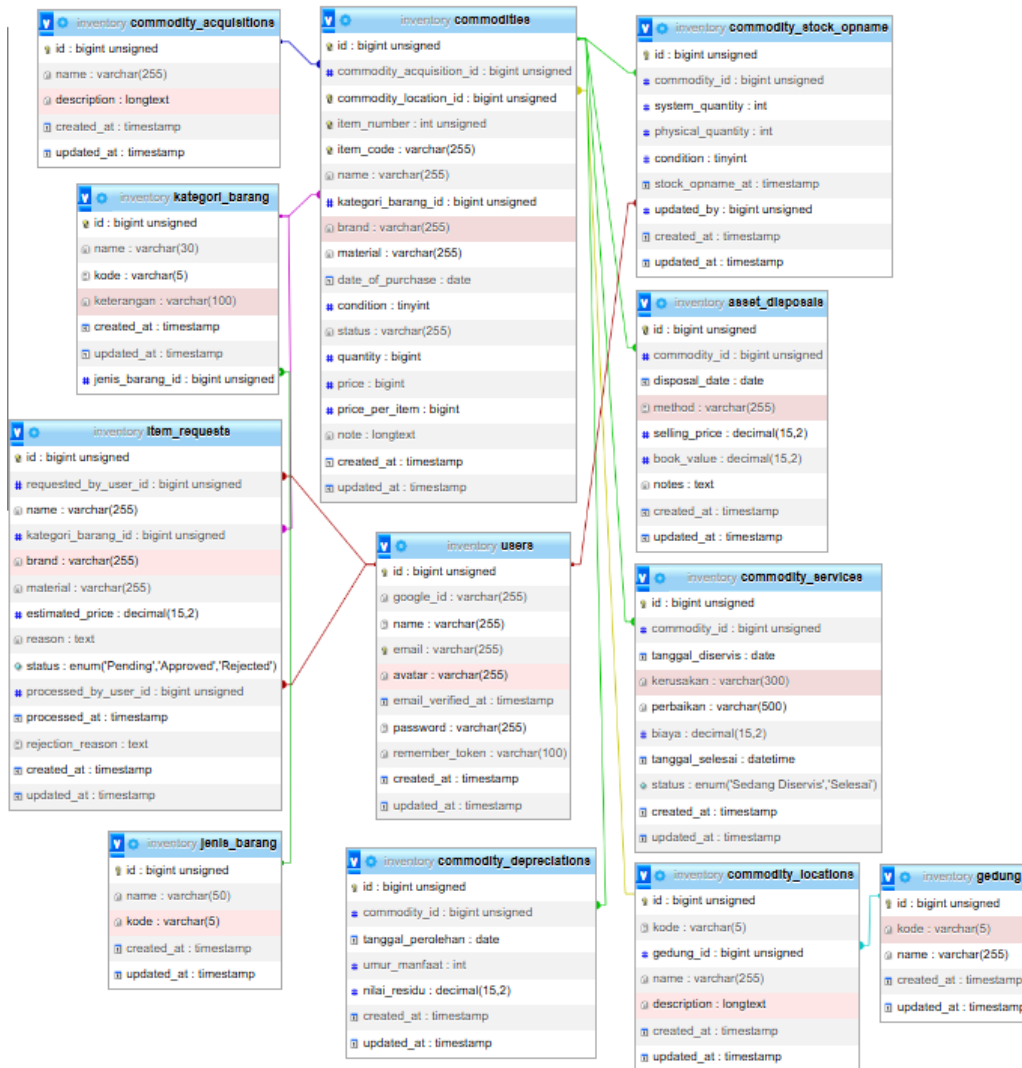


Figure 4. ERD Diagram of the Fixed Asset Inventory System

D. User Interface Display

a. Login Page Display

Figure 5 shows the login page used as the initial page to access the fixed asset inventory system. On this page, users can log in in two ways, namely by entering an email and password or by using a Google account that has been registered by the administrators.

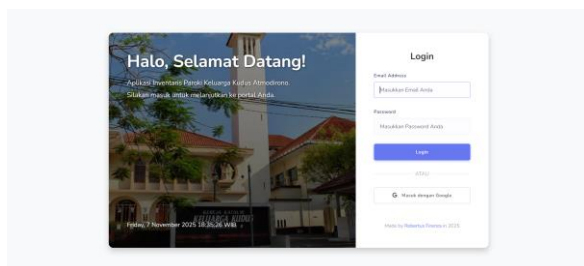


Figure 5. Login Page Display

b. Main Page Display

Figure 6 shows the main page displayed after the user successfully logs in. This page is a summary of fixed asset inventory information that can help users determine the overall inventory condition.

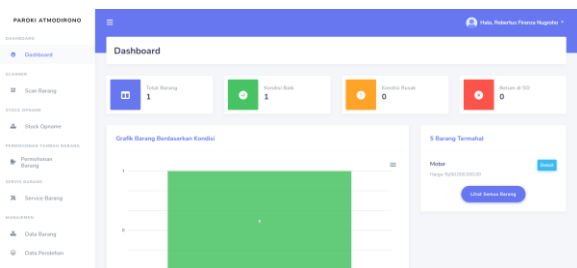


Figure 6. Main Page Display

c. Item Scan Page Display

Figure 7 is the item scanning page used to track fixed assets using QR codes attached to each asset. Users can scan the QR code, and the detailed information of the scanned item will be displayed.

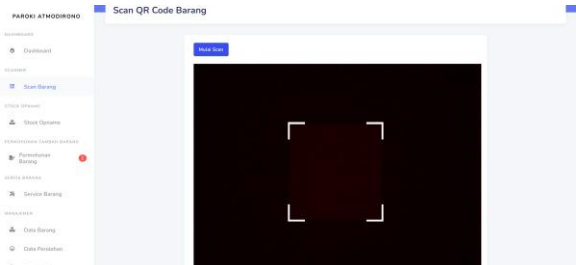


Figure 7. Item Scan Page Display

d. Item List Page Display

Figure 8 shows the item list page used to display all fixed asset data stored in the system. On this page, there are features to display the list of fixed assets, search asset data, filter asset data, add asset data, update asset data, and view asset details.

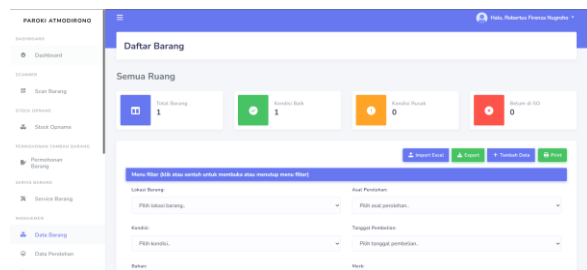


Figure 8. Item List Page Display

e. Item Request Submission Page Display

Figure 9 shows the item request submission page used to manage the submission and verification process of item requests. The Service Team can submit item requests through the system, while the Inventory Team checks and provides approval or rejection for the submitted requests.

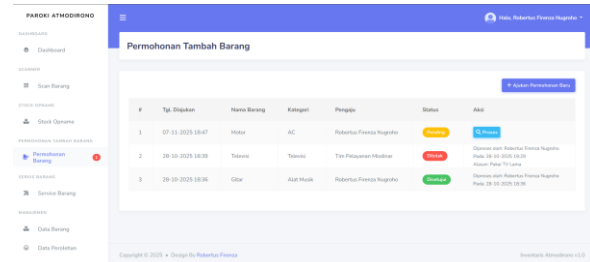


Figure 9. Item Request Submission Page Display

f. Fixed Asset Depreciation Report Page Display

Figure 10 shows one of the report pages, namely the fixed asset depreciation report page, which is used to display information on asset value depreciation. This report helps the Inventory Team and the treasurer monitor the book value of fixed assets, especially assets that are due for depreciation.

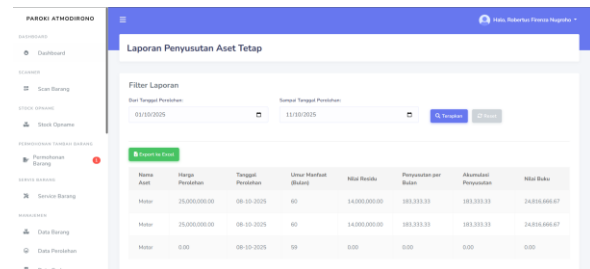


Figure 10. Fixed Asset Depreciation Report Page Display

E. Interview Results

For the Service Team, the web-based fixed asset inventory system has a positive impact because the application interface is considered easy to understand, simple, and not complicated, including for elderly users. The process of filling out item request forms becomes clearer and less confusing. In addition, this system makes the procedure for submitting item requests more efficient compared to the previous offline method, which required more time. With the web-based system, submissions can be made online via mobile devices anytime and anywhere. Overall, the Service Team feels comfortable using the system because it provides a solution to the constraints of the previous procedure.

For the Inventory Team, the system also provides significant benefits in asset management. Statistical information on the

number of items, asset conditions, and graphs displayed on the main page help the inventory monitoring process. Compared to using Microsoft Excel, the processes of data entry, verification, and request approval become easier and more structured, and can be carried out without checking physical documents. The reporting process to the treasurer becomes more practical with the system. In addition, the QR Code feature is considered very helpful in carrying out stock opname because it accelerates the physical verification process of large numbers of assets. The system is also capable of storing service history and continues to record assets that are no longer in use. Overall, the system is considered a solution that improves the efficiency and orderliness of fixed asset inventory processes.

IV. CONCLUSION

This study aims to design and develop a web-based fixed asset inventory system as a replacement for the use of Microsoft Excel at The Holy Family Parish Church Atmodirono. Based on the results of the design, implementation, and discussion, the developed system is able to provide centralized fixed asset data management so that the processes of recording, tracking, and reporting assets become more efficient than before. The interview results show that the fixed asset inventory system provides a positive impact for both the Service Team and the Inventory Team. This system simplifies the process of submitting and verifying item requests and assists in the implementation of stock opname through the QR Code scanning feature. In addition, this system also assists in the reporting process, such as reporting the depreciation of fixed asset values. Based on these results, it can be concluded that the developed web-based fixed asset inventory system has achieved the research objectives and is able to serve as a solution to the problems of fixed asset management that were previously carried out manually. For future development, this system can add automatic notification features via WhatsApp

or email that are used to provide alerts to the Inventory Team regarding routine item service schedules or notifications when new item requests are submitted, so that the item request process can be carried out without having to manually check the system first.

REFERENCES

- [1] A. Khoirunisa, A. Sidik, and L. Darmarjati, "Perancangan Sistem Informasi Manajemen Aset Berbasis Web (Studi Kasus pada: PT Onesia Nusantara Evolusioner)," *ICIT J.*, vol. 10, no. 2, pp. 214–222, 2024, doi: 10.33050/icit.v10i2.3208.
- [2] Hasmia, Nirsal, and A. Jumardi, "Rancang Bangun Aplikasi Inventaris Pada Kantor Desa Salulemo Kecamatan Baebunta Kabupaten Luwu Utara," *J. Ilm. Inf. Technol.*, vol. 12, pp. 25–32, 2022.
- [3] N. T. Yuono, M. S. Haris, and R. S. Pradini, "Designing Inventory Information System at ITSK Soepraoen Using the Waterfall Method," *J. Enhanc. Stud. Informatics Comput. Appl.*, vol. 1, no. 2, pp. 40–46, 2024, doi: 10.47794/jesica.v1i2.9.
- [4] N. Widaningsih, N. Windiyanti, and N. Rukhviyanti, "Web-Based Inventory Information System using Agile Scrum Method at CV Tunggal Putra Jaya," *Sist. J. Sist. Inf.*, vol. 14, no. 3, pp. 1471–1488, 2025, doi: 10.32520/stmsi.v14i3.5253.
- [5] E. Listiyan and E. R. Subhiyakto, "Rancang Bangun Sistem Inventory Gudang Menggunakan Metode Waterfall (Studi Kasus Di CV. Aqualux Duspha Abadi Kudus Jawa Tengah)," *Konvergensi Teknol. dan Sist. Inf.*, vol. 1, no. 1, pp. 74–82, 2021, doi: 10.24002/konstelasi.v1i1.4272.
- [6] A. C. Sutanto, B. Harnadi, and T. B. Chandrawati, "Designing a Web-Based St. Mikael Church Information System," *J. Bus. Technol.*, vol. 5, no. 2,

- pp. 78–85, 2025, doi: 10.24167/jbt.v5i2.12257.
- [7] M. H. Jusuf and A. Prabowo, “Perancangan Sistem Informasi Pendataan Posyandu Pada Kelurahan Tanjung Duren Berbasis Web,” *J. Inform. Dan Komputasi*, vol. 17, no. 2, pp. 60–67, 2023.
- [8] N. Widyani, E. Andayanie, and Nurbaeti, “Implementasi Sistem Informasi Kesehatan Terhadap Kualitas Pelayanan Administrasi Puskesmas Biru Kabupaten Bone,” *Wind. Public Heal. J.*, vol. 5, no. 6, pp. 797–806, 2024.
- [9] N. Shakira and T. D. Widajantie, “Analisis Penerapan Stock Opname Terhadap Pengendalian Internal Persediaan : Pt Bub,” *J. Ekon. Dan Bisnis*, vol. 5, no. 5, 2025.
- [10] A. Mulyanto and S. Aulia Fathi Salam, “Penerapan Metode Waterfall Pada Aplikasi Toko Online Bima Kirana Cibitung,” *J. Ilm. Teknol. Inf. Dan Komun.*, vol. 12, no. 2, pp. 34–41, 2021.
- [11] D. Saputra, W. S. Dharmawan, M. Syarif, and D. Risdiansyah, *Analisis & Perancangan Sistem Informasi*. 2023.
- [12] Steven and Irwan, “Perancangan Sistem Informasi Pengelolaan Dokumen Monitoring Pada Lembaga Pengembangan Pembelajaran dan Penjaminan Mutu Berbasis Web,” *J. Mhs. Apl. Teknol. Komput. dan Inf.*, vol. 4, no. 1, pp. 8–13, 2022.
- [13] Fauziah, A. Fatih, and A. Z. Rahanyamtel, “Rancang Bangun Sistem Informasi Penjualan Pada Toko Anisa Menggunakan PHP Native,” *J. Tek. Inform.*, vol. 3, no. 1, pp. 30–39, 2025, doi: 10.34005/insit.v3i01.4504.
- [14] A. C. Nawalagatti, D. P. R. Kolhe, and D. A. Sharma, “a Brief Study of Laravel Framework With PHP,” *J. Emerg. Technol. Innov. Res.*, vol. 8, no. 7, pp. c656–c661, 2021, [Online]. Available: www.jetir.org
- [15] J. V. Dachi and J. Suhada, “Implementasi Basis Data Sederhana Menggunakan MySQL / PostgreSQL,” *J. Tek. Inform. dan Terap.*, vol. 3, no. 2, pp. 197–207, 2025, doi: 10.62951/router.v3i2.621.
- [16] F. N. Hasanah and R. S. Untari, *Rekayasa Perangkat Lunak*. 2020.
- [17] B. Widodo, F. H. Prasetya, and A. C. Nugroho, “Design of Information Systems for Supplies on A Web-Based Company,” *J. Bus. Technol.*, vol. 5, no. 2, pp. 47–53, 2025, doi: 10.24167/jbt.v5i2.10593.