

Design of BPJS Patient Referral Information System Based on Python Tkinter at Mulia Medika Clinic

Piyyawati Dewi *

Health Information Management Study Program DIV, Politeknik Piksi Ganesha, Bandung City, West Java Province, Indonesia.

Corresponding Email: piyyawatidewi05@gmail.com.

Yuyun Yunengsih

Health Information Management Study Program DIV, Politeknik Piksi Ganesha, Bandung City, West Java Province, Indonesia.

Email: yoen1903@gmail.com.

Falaah Abdussalaam

Health Information Management Study Program DIV, Politeknik Piksi Ganesha, Bandung City, West Java Province, Indonesia.

Email: falaahabdussalaam@gmail.com.

Received: July 25, 2025; Accepted: September 5, 2025; Published: December 1, 2025.

Abstract: Digital transformation is still an ongoing process in health service delivery to improve operational performance and service quality. However, BPJS Kesehatan patient referrals are still trapped in administrative bottlenecks. At Mulia Medika Clinic, staff used to handle BPJS patient referrals manually, so that records were prone to errors and delays in obtaining information. We designed, implemented, and tested a desktop-based BPJS referral information system using Python and Tkinter for clinic operations. The development process followed the Waterfall methodology, which consisted of requirements analysis, system architecture design using Context Diagrams, Data Flow Diagrams, Entity-Relationship Diagrams as well as Flowcharts followed by implementation and black-box testing validation. The system will manage patient records, referral processing as well as user administration. Automated features include generating referral letters and producing reports. Testing has proven that this system is accurate and efficient—the manual workload has reduced, data traceability has improved, and continuity in the referral service has been maintained. Results prove operational readiness for clinic deployment to enhance administrative efficiency and precision of the referral data. Currently, it runs standalone without real-time database synchronization; hence workflow integration cannot be achieved. Future versions should have direct connections with both clinic management and BPJS databases to allow seamless data exchange without manual synchronization.

Keywords: Information System; BPJS Referral; Desktop Application; Python; Tkinter.

1. Introduction

Digital transformation in healthcare services is an effort that never ends, as organizations strive for ever-increasing levels of efficiency, accuracy, and quality of service. Referral processes in Indonesia's BPJS Kesehatan system are critical pathways within healthcare delivery yet frequently get mired in administrative hurdles and communication breakdowns between facilities. Integrated referral information systems answer these challenges by speeding up decision-making, improving traceability of patient data, and maintaining continuity of services between primary healthcare providers and hospitals to which they refer patients. After the launch of Jaminan Kesehatan Nasional (JKN), BPJS Kesehatan created a digital ecosystem that includes Primary Care applications as well as Mobile JKN. Research shows that technology adoption is impeded by many barriers including low data integration and interoperability issues between systems plus continuing dependency on manual management processes for referrals [1][2]. These problems lead to inefficient operations concerning referrals with high rates of recording errors, long queues, and waiting times for patients [3].

In private facilities such as Mulia Medika Clinic, the lack of a structured referral information system reduces the effectiveness of services provided and compromises the quality of data required for making sound medical decisions. Studies conducted recently have underscored designing information systems based on local needs with more emphasis on adaptive user interfaces as well as interoperability with BPJS platforms [4][5]. The Human-Centered Design methodology is becoming popular in research related to developing healthcare information systems because it helps create synergy between system developers and users who will use the system. Moreover, EMR integration with IHS has proven effective in enhancing the quality of primary care services at some community health centers [5]. Healthcare services across Indonesia have expanded significantly through the Jaminan Kesehatan Nasional (JKN) program run by BPJS Kesehatan. Patient referral systems from primary healthcare facilities Fasilitas Kesehatan Tingkat Pertama (FKTP) to Fasilitas Kesehatan Tingkat Lanjutan (FKTL) are basic components of JKN services. At Mulia Medika Clinic, the BPJS patient referral process is still semi-automated starting from data recording up to generating a referral letter which entails several challenges such as slow administration processing that may lead to data loss and inefficient recording and reporting activities.

As technology progresses, the need for information systems that support the processes of healthcare services becomes more and more pressing. Digitalizing referral systems can enhance workflow, increase the accuracy of data, and make documentation easier [6]. Previous studies have proven that creating referral information systems on desktop and web platforms can accelerate the process of referrals and reduce administrative burdens on medical personnel [7]. A study conducted by Sanjaya *et al.* (2019) revealed that a computer-based system could find the best referral paths for BPJS patients by analyzing hospital claim data [8]. His paper used rule-reasoning decision support integrated with an electronic referral system based on hospitals' coordinates and real bed availability information. Research in DI Yogyakarta province found that appropriate referral hospitals could be based on differential diagnosis and medical procedures needed. Historical medical diagnosis and procedure data from hospital claims were used as knowledge bases linking master data of hospitals so that primary care physicians could find the right place to refer patients before moving them to higher-level facilities. The approach using the Python programming language has also become more interesting since it is flexible in data processing and system integration [9][10]. With many libraries and frameworks owned by Python, application development can be done quickly while still maintaining code readability and ease of maintenance, making it a very suitable choice for developing healthcare information systems in conditions with limited resources.

Information security is the main concern in healthcare system development as discussed in a study conducted by Raharjo and Prayudi (2025) on cryptographic algorithm implementation for digital medical record documents [11]. Their study responded to an urgent need to protect patient confidentiality and data integrity using encryption mechanisms so that sensitive medical information remains secure when stored or transmitted. As healthcare facilities increasingly implement digital systems, security considerations are crucial for sustaining patient confidence and complying with regulations. Strong security measures—such as authentication protocols and data encryption—must be thoughtfully integrated to ensure they do not impede clinical workflows or discourage uptake among healthcare professionals.

This research will design a desktop-based BPJS patient referral information system using Python and the Tkinter graphical interface. The system is meant to help administrative staff in processing patient data, automatically print out referral letters, and keep patient referral histories in an ordered way. It has been shown that a desktop-based solution offers many benefits for small to medium-sized clinics over web-based solutions which need constant internet connectivity and server infrastructure. Desktop applications work without network availability, thus ensuring continuous service during periods of internet outages. They also provide rapid response times for data entry and retrieval because processing happens locally without any network latency. Furthermore, desktop systems allow easy deployment and maintenance for clinics having few technical support resources, reducing reliance on external IT infrastructure as well as the costs associated with it.

Therefore, the choice of Python and Tkinter is a practical choice for developing this system since these are open-source technologies that have low licensing fees but provide sufficient functionality for clinic operations. This study will answer the needs of practical low-cost healthcare information systems in developing countries characterized by limited resources and infrastructure that require implementation and maintenance with minimal technical expertise.

2. Related Work

The development of healthcare information systems, particularly for patient referral processes, has attracted considerable research attention in recent years. Sanjaya *et al.* (2019) conducted pioneering research on developing referral decision support systems using hospital claim data to improve patient flow from primary care facilities, implementing rule-reasoning mechanisms integrated with electronic referral systems that incorporated hospital coordinates and real-time bed availability data [8]. Their research demonstrated that appropriate referral hospitals could be determined based on differential diagnosis and required medical procedures, utilizing historical medical diagnosis and procedure data from hospital claims as knowledge bases to enable primary care physicians to identify suitable referral destinations before transferring patients to higher-level facilities. Building upon this foundation, Farhan *et al.* (2024) employed data mining techniques to identify referral patterns in Indonesian primary healthcare facilities, revealing systematic patterns in referral behavior that could inform the design of more efficient referral systems and highlighting opportunities for system optimization and process improvement [9]. Simanjuntak *et al.* (2023) further contributed to this area by implementing partition clustering algorithms on BPJS health insurance data to identify patient groups with similar characteristics, demonstrating that clustering techniques could support better resource planning, risk stratification, and personalized care delivery [10], suggesting that incorporating analytical capabilities into referral systems could enhance decision-making quality.

Several studies have explored different methodologies and implementations for designing healthcare information systems in the Indonesian context. Nugraha *et al.* (2020) designed a web-based planned referral information system connecting public health centers to hospitals [7], addressing the need for seamless information exchange between primary and secondary healthcare facilities while focusing on improving communication efficiency and reducing administrative delays, though their web-based approach required stable internet connectivity for optimal operation. Wijayanti *et al.* (2024) designed a BPJS participant patient referral information system that specifically addressed the unique requirements of Indonesia's national health insurance program [15], recognizing the complexity of BPJS referral procedures and developing system features tailored to these specific needs, including integration with BPJS databases and compliance with regulatory requirements. Abdussalaam and Oktaviani (2020) examined the prototyping method for developing web-based information systems, demonstrating its effectiveness in creating user-centered applications through iterative development cycles that allow for continuous user feedback and system refinement [12], which proves particularly valuable in healthcare settings where user requirements may evolve during the development process. Complementing this work, Abdussalaam and Badriansyah (2021) investigated the Structured Systems Analysis and Design (SSAD) method for web-based information system development [13], emphasizing the importance of systematic analysis and structured design approaches in creating robust and maintainable systems that provide a comprehensive framework for understanding system requirements and translating them into functional designs.

The implementation of electronic medical records and registration systems has been extensively studied in Indonesian healthcare contexts, revealing both successes and challenges in system adoption. Yunengsih (2024) analyzed the use of electronic medical records in supporting work effectiveness at medical record units in community health centers, finding that electronic systems significantly improved data accuracy, retrieval speed, and overall operational efficiency compared to manual record-keeping methods [3], though the research also identified challenges related to staff training, system maintenance, and initial implementation costs. Yunengsih *et al.* (2024) conducted an evaluation of electronic medical record implementation in the registration department at Klinik Pratama Madani Tasikmalaya, revealing both successes and ongoing challenges in system adoption and emphasizing the importance of continuous evaluation and system refinement [6], highlighting that successful implementation requires not only technical infrastructure but also organizational readiness and staff commitment. Aillin Elizabeth Joel *et al.* (2023) designed an outpatient registration information system using Visual Studio 2010 at RSUD Al-Ihsan, demonstrating that their desktop-based approach could effectively manage patient registration processes without requiring continuous internet connectivity, improving registration efficiency and reducing patient waiting times while validating the viability of desktop solutions for resource-constrained healthcare facilities [14]. Junaedi *et al.* (2024) explored the design of web-based electronic medical records integrated with the Indonesia Health Service (IHS) platform at Puskesmas Tarogong Garut [5], addressing interoperability challenges between local clinic systems and

national healthcare platforms and demonstrating that proper integration could enhance data sharing and service coordination across different healthcare levels while emphasizing the importance of standardized data formats and communication protocols in achieving seamless system integration.

Understanding factors that influence healthcare service quality, decision-making processes, and user experience provides essential context for system design, while security considerations have become increasingly critical as healthcare systems digitize sensitive patient information. Magdalena (2017) analyzed factors supporting decision-making in selecting referral hospitals in Bangka Belitung using the Analytical Hierarchy Process, identifying key factors that patients and healthcare providers consider when choosing referral destinations, including hospital reputation, distance, available facilities, and specialist availability, providing insights that inform system requirements for referral decision support features [1]. Puspitasari and Mawarni (2021) examined need, demand, and supply aspects of service quality at BPJS primary healthcare facilities during the COVID-19 pandemic in the Malang Raya region [2], revealing gaps between service capacity and patient needs and highlighting areas where information systems could help optimize resource allocation and improve service delivery, emphasizing that effective information systems must account for both supply-side constraints and demand-side requirements. Aly Baziyad *et al.* (2021) applied Human-Centered Design methodology to develop user experience for hospital queue applications at RSUD 45 Kuningan, underscoring the significance of involving end-users throughout the design process to ensure that systems meet actual user needs and preferences [4], revealing that systems designed with strong user involvement achieved higher adoption rates and user satisfaction levels compared to systems developed without user input. Raharjo and Prayudi (2025) addressed security concerns by implementing AES and LZMA algorithms to secure electronic medical documents, demonstrating that cryptographic approaches could effectively protect patient confidentiality and data integrity during storage and transmission while emphasizing that security measures must be integrated into system design from the outset rather than added as afterthoughts [11], ensuring that patient data remains protected throughout its lifecycle.

While existing research has made valuable contributions to healthcare information system development, several gaps remain that the current study addresses. Most studies have focused on web-based solutions that require continuous internet connectivity and server infrastructure, which may not be practical for small to medium-sized clinics with limited resources, and desktop-based approaches, though mentioned in some studies, have received less attention despite their potential advantages in resource-constrained settings. Furthermore, while various design methodologies have been explored, there is limited research specifically addressing the integration of referral management, automatic document generation, and historical data management in a unified desktop application tailored for BPJS patient referrals in private clinic settings. The current study addresses these gaps by designing a desktop-based BPJS patient referral information system using Python and Tkinter, specifically tailored for clinics like Mulia Medika that require practical, cost-effective solutions capable of operating independently of network infrastructure while maintaining comprehensive functionality for referral management, combining the advantages of desktop applications demonstrated by Aillin Elizabeth Joel *et al.* (2023) with the specific BPJS referral requirements identified by Wijayanti *et al.* (2024) and the user-centered design principles advocated by Aly Baziyad *et al.* (2021) [14][15][4].

3. Research Method

3.1 Waterfall Method Planning Design

This research employs the Waterfall method, a sequential and systematic software development model. The model consists of several stages carried out in order, where each stage must be completed before proceeding to the next. The stages of the Waterfall method used in designing the BPJS patient referral information system are as follows:

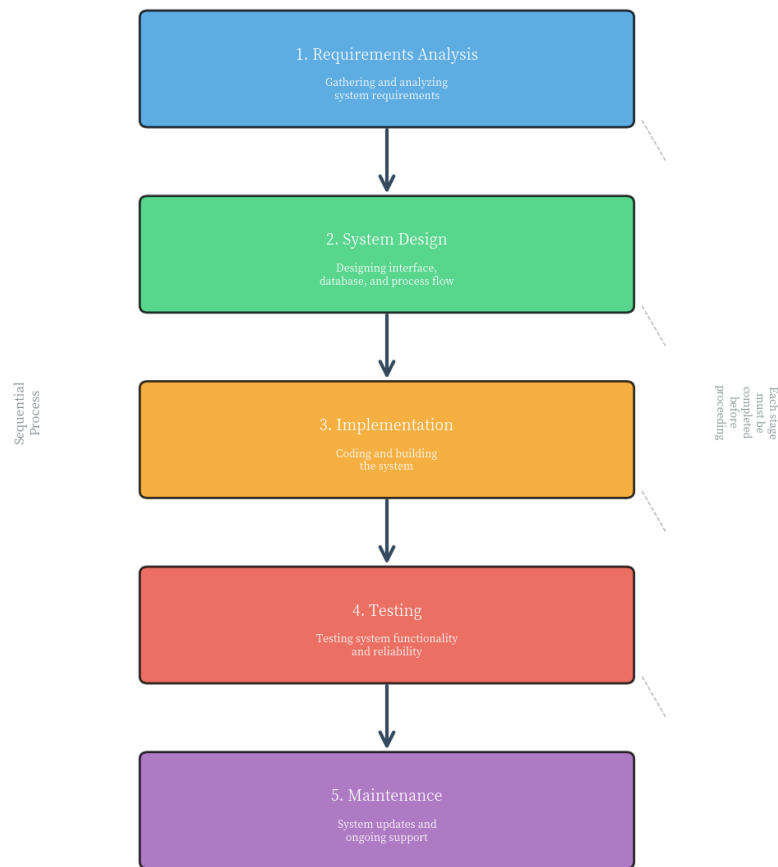


Figure 1. Waterfall Planning Design

- 1) **Requirements Analysis**
The first stage involves gathering and analyzing system requirements from users, specifically administrative staff and referral officers at Klinik Mulia Medika. Data collection was conducted through direct observation, interviews with clinic personnel, and document studies related to the BPJS patient referral process. The outcome of this stage is a system requirements document that serves as the foundation for system design.
- 2) **System Design**
After analyzing requirements, a thorough system design was developed. The design includes user interface planning, database structure, and referral process flow. System design was carried out by considering ease of use, data security, and process efficiency. Tools used include flowcharts, ERD (Entity Relationship Diagram), DFD, and context diagrams [12].
- 3) **Implementation**
This stage involves the system coding process based on the design that has been created. The programming language used is Python with the Tkinter library to build the graphical interface. Additionally, SQLite is used as a local database integrated into the application. Main functions such as patient data input, referral letter creation, medical record documentation, and referral status tracking are implemented at this stage.
- 4) **Testing**
After the system is fully implemented, testing is conducted to ensure that the system operates according to specifications and user requirements. Testing is performed using the black-box testing method, which examines system functions based on the output generated without viewing the source code. Testing was carried out by several users at the clinic to identify errors and ensure system reliability [13].
- 5) **Maintenance**
The final stage is system maintenance. After the system is deployed, new requirements or minor errors may be discovered that need to be corrected. Maintenance is performed to ensure the system continues to operate properly and can adapt to policy changes or requirements from the clinic.

3.2 Data Collection Techniques

In this research, data was collected through several techniques as follows:

- 1) Interviews
Conducted directly with relevant parties, such as administrative officers, clinic staff, and doctors involved in the referral process. Interviews aim to extract system requirements and understand the current workflow.
- 2) Observation
Researchers conducted direct observation of the patient referral process at Klinik Mulia Medika. Observation was useful for obtaining a realistic picture of the obstacles faced and the manual processes that have been used [14].
- 3) Documentation Study
Researchers examined related documents such as referral forms, patient data, and standard operating procedures (SOP) used in the BPJS patient referral process. The purpose is to ensure that the system created complies with applicable rules and policies.
- 4) Literature Review
Literature review was also conducted to serve as a reference for development, which is crucial for understanding workflow and system requirements, as demonstrated by Wijayanti *et al.* (2024) who used hospital claim data to design a BPJS patient referral system [15].

4. Result and Discussion

4.1 Results

The development of a Python Tkinter-based BPJS patient referral information system for Mulia Medika Clinic followed several stages: requirements analysis, design, implementation, and testing.

4.1.1 System Requirements Analysis

The initial stage identified functional and non-functional requirements to ensure the system operates according to clinic needs. Functional requirements include managing BPJS patient data, processing referrals, printing referral letters, and storing patient visit history. The system also features a login module to restrict access to authorized administrators or operators only. For non-functional requirements, the system runs efficiently on local clinic devices with a simple interface that users without technical backgrounds can understand easily. Staff members handle daily administrative tasks, so ease of use became a priority during development.

4.1.2 System Design

Various diagrams visualize the structure, functionality, and workflow of the information system. These diagrams help understand system complexity, identify entities and their relationships, and map involved business processes. The Context Diagram represents the highest level of a system, showing interactions between the system and external entities that communicate with it. The diagram focuses on main data flows entering and leaving the system without showing internal details. The Context Diagram provides a complete picture of system boundaries and interfaces.

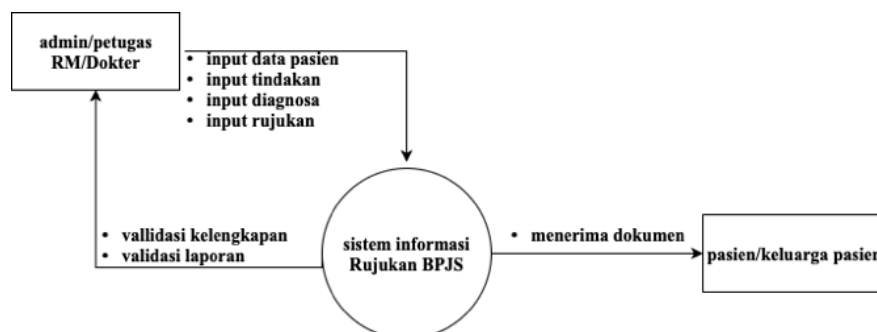


Figure 2. Context Diagram

Based on the context diagram, "BPJS Referral Information System" serves as the system core. External entities interacting with the system include "admin/medical records staff/Doctor" and "patient/patient's family". The "admin/medical records staff/Doctor" role provides various data inputs such as patient data, diagnosis, treatment, and referrals, while receiving completeness validation from the system. Meanwhile, "patient/patient's family" receives output documents from the system. The diagram illustrates the system's fundamental function: managing BPJS referral data and providing referral documents to patients or their families, with input and validation processes involving medical personnel. The Data Flow Diagram (DFD) breaks down the context diagram into more detail. The diagram visualizes main processes occurring within the system

and how data flows between processes, between processes and data storage (datastore), and between processes and external entities. DFD provides more detailed information about core system functions compared to the context diagram while maintaining a summary level.

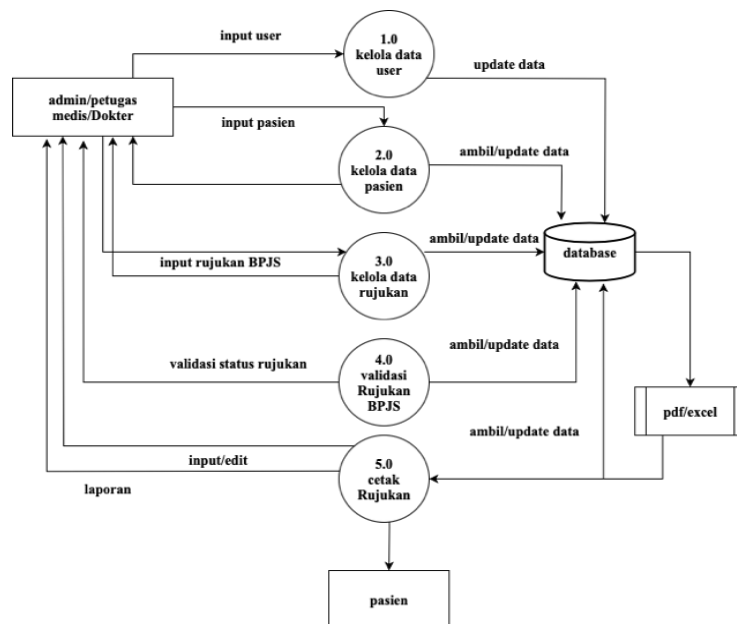


Figure 3. System DFD

The DFD analysis for the BPJS Referral Information System shows five interconnected main processes. Process 1.0 manages user data, handling information for admins, medical staff, or doctors, where data input comes from these parties and updates to the database. Process 2.0 manages patient data; patient data input comes from admin/medical staff/doctor, then retrieves or updates data in the database. Next, process 3.0 manages referral data, handling BPJS referral information with input from admin/medical staff/doctor, plus retrieval and update operations in the database. Process 4.0 validates BPJS Referrals, checking referral status based on input from admin/medical staff/doctor and performing retrieve/update operations from/to the database. Finally, process 5.0 prints medical referrals to produce referral documents. The process receives input or changes from admin/medical staff/doctor, retrieves/updates data from/to the database, and generates reports for admin/medical staff/doctor plus document output for patients. All processes interact with the database as the central data storage. Additionally, the system produces output in PDF/Excel format, likely results from the referral printing process or other reports originating from the database. The Entity Relationship Diagram (ERD) is a conceptual data model used to describe the logical structure of a database. ERD identifies entities (objects or concepts relevant for data storage), attributes (properties or characteristics of each entity), and relationships (connections or associations) between entities in a system. The diagram plays a crucial role in the database design phase to ensure optimal data consistency and integrity.

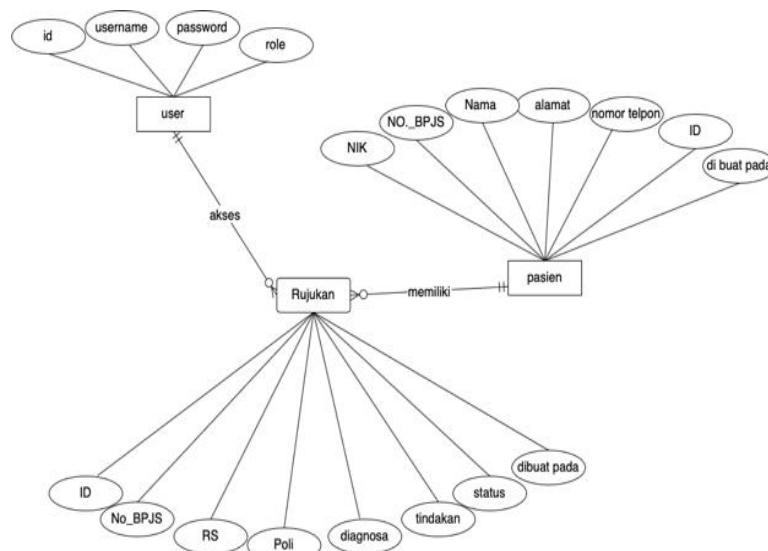


Figure 4. System ERD

The ERD visualizes the database structure for the BPJS Referral Information System through three main entities. The user entity represents system users such as admin, medical staff, or doctors, with attributes including id, username, password, and role. The patient entity stores patient information, represented by attributes ID, NIK, NO_BPJS, Name, address, phone number, and created at. Meanwhile, the Referral entity contains BPJS referral data, with attributes ID, No_BPJS, RS (Hospital), Poly, diagnosis, treatment, status, and created at. A Flowchart, or Flow Diagram, is a graphical representation of a process, algorithm, or workflow. The diagram uses various standard symbols to describe the sequence of steps, decision points, start and end points, and process flow direction. Flowcharts effectively visualize the operational logic of a system, help identify potential problems or obstacles, and communicate how a process should run.

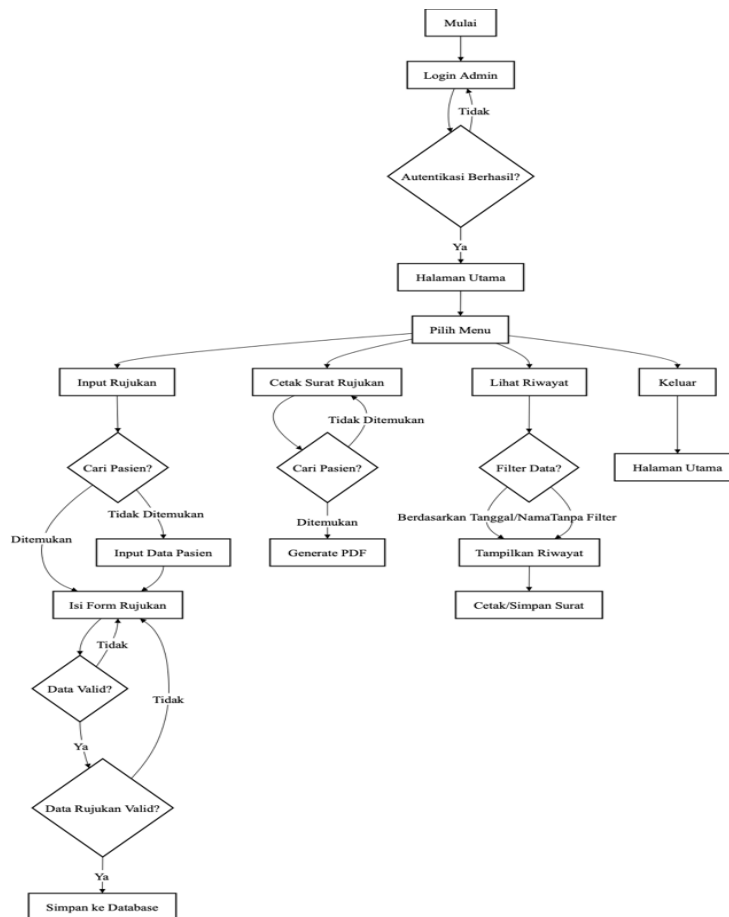


Figure 5. System Flowchart

The flowchart describes the main workflow in the system, most likely from the perspective of an "admin/medical staff". The process begins with Start, followed by the Admin Login phase. If authentication fails, the system returns the user to the login stage; however, if successful, the user proceeds to the Main Page. From the Main Page, users have several options, including Select Menu leading to various functionalities, or Exit to end the session.

4.1.3 System Implementation

The section explains the realization of BPJS referral information system functionality based on established design. Implementation covers the user interface (UI) and core functionality allowing users (admin/medical staff) to manage patient data, referrals, and users. The system features an intuitive and easy-to-use interface, as shown in the provided screenshots.



Figure 6. Login Page

Users start system interaction through the login page (Figure 6. Login Page), requiring Username and Password for authentication. The step ensures only authorized users can access the system.

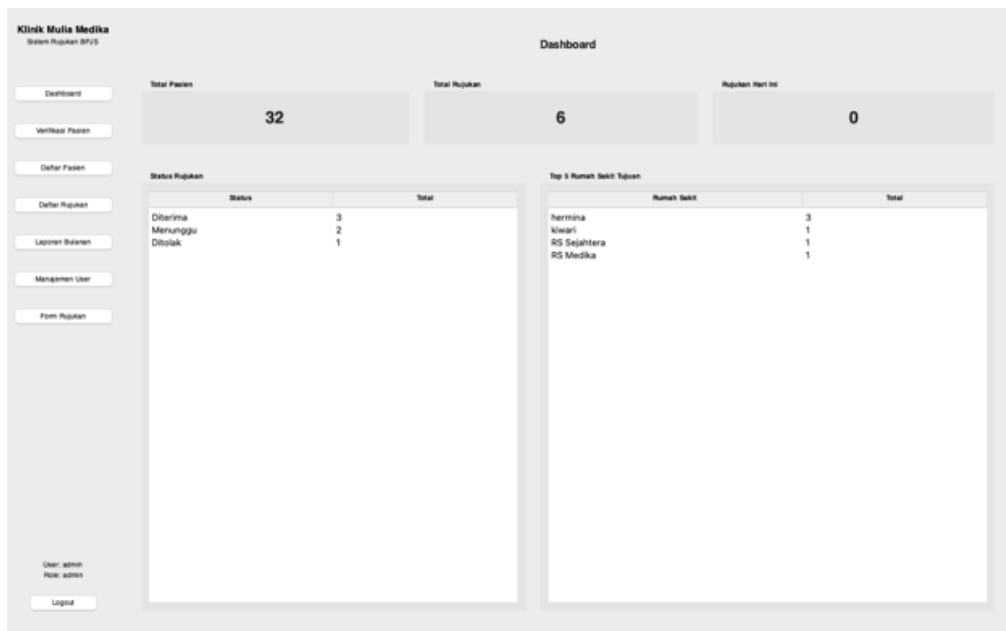


Figure 7. Dashboard

After successful login, users proceed to the dashboard (Figure 7. Dashboard). The page displays a summary of key information, including total patients, total referrals, and referrals for today. The dashboard also shows referral status statistics (accepted, pending, rejected) and the top five referral destination hospitals, giving an overview of system performance.

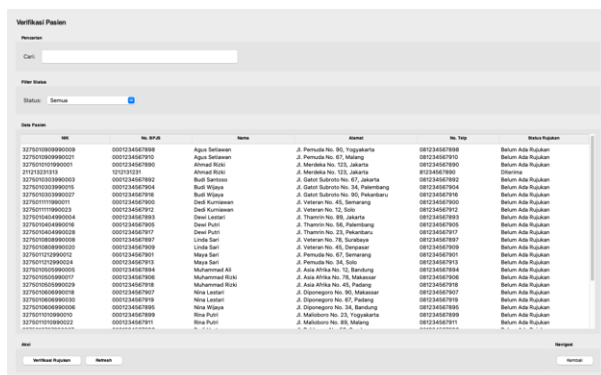


Figure 8. Patient Verification

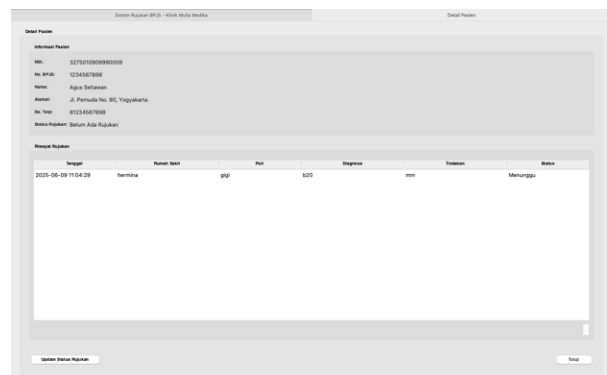


Figure 9. Patient Details

The module (Figure 8. Patient Verification) allows staff to view existing patient lists, search, and filter data based on referral status. The function verifies whether patients already have a referral history. When a patient is selected from the verification list, the system displays detailed patient information along with previously submitted referral history (Figure 9. Patient Details). From the page, referral status can be updated.

NIK	No. BPJS	Nama	Alamat	No. Telp	Tanggal Daftar
3275010101990001	0001234567890	Ahmad Rizki	Jl. Merdeka No. 123, Jakarta	081234567890	2025-06-09 09:16:16
3275010202990002	0001234567891	Siti Nurhaliza	Jl. Sudirman No. 45, Jakarta	081234567891	2025-06-09 09:16:16
3275010303990003	0001234567892	Budi Santoso	Jl. Gatot Subroto No. 67, Jakarta	081234567892	2025-06-09 09:16:16
3275010404990004	0001234567893	Dewi Lestari	Jl. Thamrin No. 89, Jakarta	081234567893	2025-06-09 09:16:16
3275010505990005	0001234567894	Muhammad Ali	Jl. Asia Afrika No. 12, Bandung	081234567894	2025-06-09 09:16:16
3275010606990006	0001234567895	Nina Wijaya	Jl. Diponegoro No. 34, Bandung	081234567895	2025-06-09 09:16:16
3275010707990007	0001234567896	Rudi Hartono	Jl. Pahlawan No. 56, Surabaya	081234567896	2025-06-09 09:16:16
3275010808990008	0001234567897	Linda Sari	Jl. Veteran No. 78, Surabaya	081234567897	2025-06-09 09:16:16
3275010909990009	0001234567898	Agus Setiawan	Jl. Pemuda No. 90, Yogyakarta	081234567898	2025-06-09 09:16:16
3275011010990010	0001234567899	Rina Putri	Jl. Maliboro No. 23, Yogyakarta	081234567899	2025-06-09 09:16:16
3275011111990011	0001234567900	Dedi Kurniawan	Jl. Veteran No. 45, Semarang	081234567900	2025-06-09 09:16:16
3275011212990012	0001234567901	Maya Sari	Jl. Pemuda No. 67, Semarang	081234567901	2025-06-09 09:16:16
3275010101990013	0001234567902	Rudi Santoso	Jl. Merdeka No. 89, Medan	081234567902	2025-06-09 09:16:16
3275010202990014	0001234567903	Siti Aminah	Jl. Sudirman No. 12, Medan	081234567903	2025-06-09 09:16:16
3275010303990015	0001234567904	Budi Wijaya	Jl. Gatot Subroto No. 34, Palembang	081234567904	2025-06-09 09:16:16
3275010404990016	0001234567905	Dewi Putri	Jl. Veteran No. 56, Palembang	081234567905	2025-06-09 09:16:16
3275010505990017	0001234567906	Muhammad Rizki	Jl. Asia Afrika No. 78, Makassar	081234567906	2025-06-09 09:16:16
3275010606990018	0001234567907	Nina Lestari	Jl. Diponegoro No. 90, Makassar	081234567907	2025-06-09 09:16:16
3275010707990019	0001234567908	Rudi Hartono	Jl. Pahlawan No. 23, Denpasar	081234567908	2025-06-09 09:16:16
3275010808990020	0001234567909	Linda Sari	Jl. Veteran No. 45, Denpasar	081234567909	2025-06-09 09:16:16
3275010909990021	0001234567910	Agus Setiawan	Jl. Pemuda No. 67, Malang	081234567910	2025-06-09 09:16:16
3275011010990022	0001234567911	Rina Putri	Jl. Maliboro No. 89, Malang	081234567911	2025-06-09 09:16:16
3275011111990023	0001234567912	Dedi Kurniawan	Jl. Veteran No. 12, Solo	081234567912	2025-06-09 09:16:16
3275011212990024	0001234567913	Maya Sari	Jl. Pemuda No. 34, Solo	081234567913	2025-06-09 09:16:16
3275010101990025	0001234567914	Rudi Santoso	Jl. Merdeka No. 56, Surakarta	081234567914	2025-06-09 09:16:16
3275010202990026	0001234567915	Siti Aminah	Jl. Sudirman No. 78, Surakarta	081234567915	2025-06-09 09:16:16
3275010303990027	0001234567916	Budi Wijaya	Jl. Gatot Subroto No. 90, Pekanbaru	081234567916	2025-06-09 09:16:16
3275010404990028	0001234567917	Dewi Putri	Jl. Thamrin No. 23, Pekanbaru	081234567917	2025-06-09 09:16:16
3275010505990029	0001234567918	Muhammad Rizki	Jl. Asia Afrika No. 45, Padang	081234567918	2025-06-09 09:16:16
3275010606990030	0001234567919	Nina Lestari	Jl. Diponegoro No. 67, Padang	081234567919	2025-06-09 09:16:16

Figure 10. Patient List

The module (Figure 10. Patient List) displays all registered patients in the system with basic information such as NIK, BPJS No., Name, Address, Phone No., and Registration Date. There is an option to add new referrals directly from the list.

Figure 11. Referral Form

Tanggal	No. BPJS	Nama	RS Tujuan	Poli	Poli	Poli	Poli	Poli	Poli	Poli
09/06/2025	121212121	Ahmad Rizki	RS Sulthama	Poli Umum	mm	mm	mm	mm	mm	mm
09/06/2025			Kardi	gpr	gpr	gpr	gpr	gpr	gpr	gpr
09/06/2025			RS Medika	Poli Umum	gpr	gpr	gpr	gpr	gpr	gpr
09/06/2025	121212	Jamal	RS Medika	Poli Umum	gpr	gpr	gpr	gpr	gpr	gpr
09/06/2025			RS Medika	Poli Umum	gpr	gpr	gpr	gpr	gpr	gpr
09/06/2025			RS Medika	Poli Umum	gpr	gpr	gpr	gpr	gpr	gpr

Figure 12. Referral List

The "Referral Form" module (Figure 11. Referral Form) fills in new referral data. The form includes information about the referred patient (Name, BPJS No.) and referral details such as Destination Hospital, Poly, Diagnosis, Treatment, and Referral Status. The "Referral List" page (Figure 12. Referral List) displays all created referral data. Users can filter data by status and perform searches. There are options to update referral status and generate PDF documents from selected referrals.

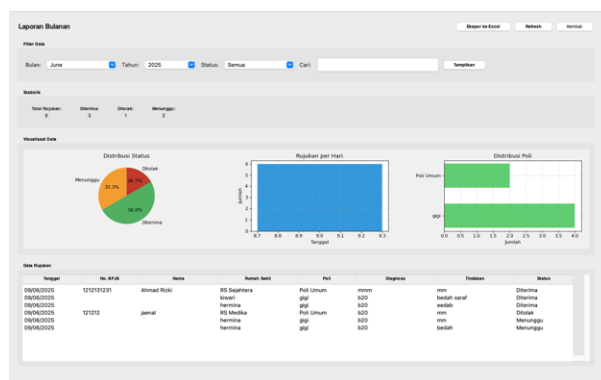


Figure 13. Monthly Report

Username	Password	Role	ID User
admin	admin	admin	admin123
admin	admin	admin	admin123
admin	admin	admin	admin123
admin	admin	admin	admin123

Figure 14. User Management

The "Monthly Report" module (Figure 13. Monthly Report) provides features to view referral statistics in monthly periods. The report includes data visualization such as referral status distribution (pie chart), referrals

per day (bar chart), and poly distribution (bar chart). Displayed referral data can be filtered by month, year, and status, and exported to Excel format. The "User Management" module (Figure 14. User Management) allows administrators to manage system user accounts. Functionality includes adding, changing passwords, and deleting users, as well as displaying existing user lists along with their username, Role, and LOGIN ID.

Figure 15. Patient & Referral Management Form

The module (Figure 15. Patient & Referral Management Form) is a page that integrates patient search, referral data entry, and patient list display in one interface. The design speeds up the process of creating new referrals quickly. The system implementation successfully provides a platform for managing BPJS referrals at Mulia Medika Clinic, with features supporting business processes from user authentication, patient and referral data management, to reporting.

Tanggal	No. BPJS	Nama	Rumah Sakit	Poli	Diagnosa	Tindakan	Status
09/06/2025	1212131231	Ahmad Rizki	RS Sejahtera	Poli Umum	mmm	mm	Diterima
09/06/2025	1212131231	daniele	kiwari	gigi	b20	bedah saraf	Diterima
09/06/2025	1212131231	Ahmad jae	hermina	gigi	b20	eedab	Diterima
09/06/2025	1212131231	jaenal	RS Medika	Poli Umum	b20	mm	ditolak
09/06/2025	1212131231	kevin	hermina	gigi	b20	mm	Menunggu
09/06/2025	1212131231	herman	hermina	gigi	b20	bedah	Menunggu

Figure 16. Generated PDF Result

Export PDF File (Referral Letter): Testing on the "Referral List" module proves the system accurately generates referral letter documents in PDF format. After activating the "Generate PDF" command for selected referrals, the system successfully downloads a PDF file containing all relevant information about the referral, including patient data, referral destination, diagnosis, and treatment. Data consistency between system preview and generated PDF file content confirms the reliability of the feature for official documentation.

SURAT RUJUKAN PASIEN BPJS

Yang bertanda tangan di bawah ini:
 Nama : dr. ahmad
 Jabatan : Dokter Pemeriksa
 Alamat : Klinik Mulya Medika

Dengan ini menerangkan bahwa:

Nama Pasien	:	Ahmad Rizki
No. BPJS	:	1212131231
Alamat	:	Alamat Pasien
Poli	:	Poli Umum
Diagnosa	:	mmm
Tindakan	:	mm

Berdasarkan pemeriksaan yang telah dilakukan, pasien tersebut di atas memerlukan penanganan lebih lanjut di:

Rumah Sakit Tujuan	:	RS Sejahtera
Tanggal Rujukan	:	09/06/2025
Status	:	Diterima

Demikian surat rujukan ini dibuat untuk dipergunakan sebagaimana mestinya.

Hormat kami,

dr. ahmad
Dokter Pemeriksa
SIP: 1234567890

Catatan:
 1. Surat rujukan ini berlaku selama 3 (tiga) hari sejak tanggal diterbitkan
 2. Pasien diharapkan membawa surat rujukan ini saat mendaftar di rumah sakit tujuan
 3. Untuk informasi lebih lanjut, silakan hubungi kami di nomor yang tertera di atas

Figure 17. Generated Excel Result

In the "Monthly Report" module, the data export functionality to Excel format has been successfully tested. After users set report filters and click the "Export to Excel" button, the system successfully downloads a file with .xlsx extension. Verification of the downloaded Excel file shows that all referral data displayed in the monthly report loads completely and arranges neatly in a consistent table format. The result confirms the accuracy and efficiency of the report export feature for further data analysis.

4.1.4 System Testing (Black-box Testing)

System testing ensures implemented functionality runs according to requirement specifications. The black-box testing approach was chosen because it focuses on testing system functionality from the user's perspective without considering internal structure or program code. Testing involves providing input and observing generated output, comparing it with expectations.

Table 1. Black-box Test Results

Module	Test Case	Input	Result
Login	Login With Valid Data	Correct Username & Password	Valid
Login	Wrong Username	Wrong Username	Valid
Login	Wrong Password	Wrong Password	Valid
Login	Empty Column	No Username/Password Filled	Valid
Dashboard	Initial Display	Successful Login	Valid
Patient Verification	Display List	Access Module	Valid
Patient Verification	Filter By Status	Select "Accepted"	Valid
Patient Verification	Search By Keyword	Patient Name	Valid
Patient Details	View Details & History	Click Patient	Valid
Patient Details	Change Referral Status	Select "Accepted" Status	Valid
Patient List	View All Patients	Access Module	Valid
Patient List	Add New Referral	Click Add Button	Valid
Referral Form	Save Complete Data	Fill All Fields	Valid
Referral Form	Validate Empty Data	Empty 1 Required Field	Valid
Referral List	Display All Referrals	Access Module	Valid
Referral List	Filter Status	Select "Pending"	Valid
Referral List	Search Referral	Patient Name	Valid
Referral List	Change Referral Status	Select "Rejected"	Valid
Referral List	Print PDF	Click "Print PDF"	Valid
Monthly Report	Display Default	Access Module	Valid
Monthly Report	Filter Report	Select Month, Year, Status	Valid
Monthly Report	Export Excel	Click "Export"	Valid
User Management	View User List	Access Module	Valid

User Management	Add New User	Fill Valid Form	Valid
User Management	Change User Password	Change Password Data	Valid
User Management	Delete User	Confirm Delete	Valid
Integrated Form	Display Patient & Referral Integration	Access Module	Valid
Integrated Form	Fill Referral From Patient List	Select Patient & Fill Data	Valid

4.2 Discussion

The BPJS patient referral information system that has been developed is able to function according to the functional and non-functional requirements that have been described in the previous chapter. The functional requirements include patient data management, referral processing, printing of referral letters, and storage of visit histories; all these have been implemented. The login module has been applied correctly, which means only authorized users can access this system so that it can protect the patient's data. Non-functionally, this system works well on the local clinic machines. It has a simple and user-friendly interface design so that clinic administrative staff may be able to use it without having to go through a long course of technical training. This result is in line with the objective set at the beginning of development: increasing operational efficiency for clinics in managing BPJS patient referrals. Wijayanti *et al.* (2024) found similar results [15]; they stated that if a referral information system is designed well, it would be easy for an administration process and would shorten its processing time at a healthcare facility. The use of various diagrams at the system design stage was very useful for visualizing structure and workflow. The Context Diagram provides a brief summary of the interaction between the system and external entities: admin/medical staff/doctor and patient/patient's family. The DFD describes five main processes that are interconnected: user data management, patient data management, referral data management, validation of referrals, and printing medical referrals. An ERD with three main entities (user, patient, and referral) indicates a good database structure. Relationships among entities have been designed to keep data consistent and ensure integrity. A flowchart for the system shows logic in an easy-to-understand way from login through all possible functions in the main menu. Abdussalaam & Badriansyah (2021) support structured system analysis and design diagrams because they help make a clear understanding of requirements as well as facilitate early identification of potential problems during development [13].

User interface implementation has a design that is attractive and responsive. The dashboard contains some summary information, such as total patients, total referrals, and referral status statistics, which help in monitoring system performance at runtime. Data visualization takes the form of graphs (pie chart and bar chart) in the Monthly Report module to help analyze referral data. The developed modules are Patient Verification, Patient Detail, Patient List, Referral Form, Referral List, Monthly Report, and User Management that integrate well and support each other in the patient referral business process. Search and filter features in several modules increase efficiency when accessing data needed by users. Aly Baziyad *et al.* (2021) proved that user-centered design approaches improve significantly both user satisfaction and operational efficiency for non-technical staff in healthcare applications [4]. Such an approach was implemented in this application development process as well as supporting features such as document generation in PDF and Excel formats which add more value to the system. The PDF export feature for the referral letter ensures that an official document can be printed with a uniform professional look while the Excel export feature for monthly reports allows further analysis on data using spreadsheet applications which are generally used for reporting purposes and management decision-making activities. Junaedi *et al.* (2024) stated that document generation features in electronic medical record systems are very important to keep standardized documentation support administrative workflows within healthcare facilities [5]. This feature also allows better data archiving and retrieval processes which are very important for audit purposes quality assurance activities. Black-box testing results indicate that all system functionalities are working properly and according to specifications. All 29 test cases executed on different modules resulted in "Valid" status. The results show that the system is compliant with the requirement specifications and is ready to be used in Mulia Medika Clinic's operational environment. Testing has been conducted for different scenarios, including normal cases (valid data) as well as error handling cases (invalid data or empty fields). The system has proven capable of managing all input conditions properly, including performing validation on data in referral forms and authentication in the login module. Nugraha *et al.*, (2020) stated that rigorous testing procedures are required for the reliability of systems and accuracy of data in healthcare information systems [7], especially those dealing with sensitive patient information and critical referral processes. The system has some strengths that make it feasible to be implemented at Mulia Medika Clinic. First, its user-friendly interface will allow non-technical users to operate the system without long training periods, thus reducing both cost and time of implementation. Second, good integration among various modules of the system ensures smooth flow of data and minimizes repetitive entries which may lead to errors. Thirdly, visualization features help staff analyze data easily by presenting them in digestible formats. Fourthly, exportable documents available in PDF format as well as Excel provide flexibility for diverse reporting and documentation needs. Fifthly, a secure login module with user management

safeguards sensitive patient information from unauthorized access to the system. Finally, the fact that this application can work locally without any internet connection makes it a reliable solution in places where network connectivity is unstable. Sanjaya *et al.*, (2019) stated that locally hosted healthcare information systems have an advantage in terms of data control and availability of the system for small healthcare facilities having limited IT infrastructure [8].

There are several areas that need to be improved in the future. The system is standalone, meaning it cannot be accessed simultaneously from different locations, which restricts collaboration among clinic branches or remote access scenarios. The system has yet to integrate with the BPJS online system for real-time patient data verification and requires manual verification processes that may take time. Data backup still relies on manual processes, thus increasing the possibility of losing data in case of hardware failure. The lack of automatic notification features for referral status requires staff to check manually for updates, which may delay coordination in patient care. Farhan *et al.* (2024) stated that integration with national health insurance systems and automated notification mechanisms would improve referral management efficiency significantly and reduce administrative burden on healthcare staff [9]. Future enhancements could address these issues by creating a web-based version that supports multi-location access with data security, integration with the BPJS online verification system enabling real-time eligibility checks reducing manual verification workload, implementing automated backup systems ensuring data safety business continuity, and adding notification features via email or SMS keeping parties informed about changes in referral status promptly. Simanjuntak *et al.* (2023) advanced data analytics and clustering techniques applied to BPJS health insurance data provide valuable information for improving referral patterns and resource allocation [10]. These enhancements will upgrade the current system into a more powerful and scalable solution that can support increasing clinic operations and changing healthcare needs.

5. Conclusion

Based on the design, implementation, and testing stages that have been carried out, the development of the BPJS Referral Information System at Mulia Medika Clinic has been successfully completed. The system design involving standard diagrams such as Context Diagram, DFD, ERD, and Flowchart has provided a solid foundation for the development process. The implemented system successfully delivers an intuitive user interface that functions properly and includes essential modules such as patient management, referral management, user management, and reporting features. Testing results using the black-box testing method demonstrate that all main system functions, including authentication processes, data management, and document export features, operate according to specifications. These achievements indicate that the system is ready for deployment in clinic operations and is expected to enhance efficiency and accuracy in the BPJS patient referral process.

References

- [1] Magdalena, H. (2017). Analisis Faktor–Faktor Pendukung Pengambilan Keputusan Memilih Rumah Sakit Rujukan Di Bangka Belitung Dengan Analytical Hierarchy Process. *Fountain Informatics J*, 2(2), 10. <https://doi.org/10.21111/fij.v2i2.1196>
- [2] Puspitasari, S. T., & Mawarni, D. (2021). Need, demand dan supply pada kualitas layanan fasilitas kesehatan tingkat I BPJS Kesehatan era pandemi Covid-19 di wilayah Malang Raya. *Preventia: The Indonesian Journal of Public Health*, 6(1), 13-22. <https://doi.org/10.17977/um044v6i1p13-22>
- [3] Yunengsih, Y. (2024, November). Analysis Of The Use Electronic Medical Record In Supporting Work Effectiveness Of Medical Record Unit at Puskesmas Tegalbugub. In *Proceedings* (Vol. 4, No. 1, pp. 71-80).
- [4] Aly Baziyad, H., Tolle, H., & Rokhmawati, R. I. (2021). Perancangan pengalaman pengguna pada aplikasi antrean rumah sakit menggunakan metode Human-Centered Design (Studi kasus: RSUD 45 Kuningan). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, 5(6), 2152–2162. <http://j-ptiik.ub.ac.id>

- [5] Junaedi, F. A., Suryani, D. L., & Fadly, F. (2024). Perancangan rekam medis elektronik berbasis web dengan platform Indonesia Health Service (IHC) di Puskesmas Tarogong Garut. *Jurnal Kesehatan Bakti Tunas Husada: Jurnal Ilmu-Ilmu Keperawatan, Analis Kesehatan dan Farmasi*, 24(1), 84–96. <https://doi.org/10.36465/jkbth.v24i1.1297>
- [6] Yunengsih, Y., & Syahidin, Y. (2024). Evaluasi Penerapan Rekam Medis Elektronik Bagian Pendaftaran di Klinik Pratama Madani Tasikmalaya. *Media Bina Ilmiah*, 18(10), 2663–2670. <https://doi.org/10.33758/mbi.v18i10.804>.
- [7] Nugraha, E., Arso, S. P., & Prasetyo, A. B. (2020). Design of web-based planned referral information system from public health centers to hospital. *Unnes Journal of Public Health*, 9(1), 11–19.
- [8] Sanjaya, G. Y., Lazuardi, L., Hasanbasri, M., & Kusnanto, H. (2019). Using hospital claim data to develop referral decision support systems: Improving patient flow from the primary care. *Procedia Computer Science*, 161, 441–448. <https://doi.org/10.1016/j.procs.2019.11.143>
- [9] Farhan, M., Santosa, B., & Sholihah, M. (2024). Identification of referral pattern in Indonesian primary healthcare facilities using data mining techniques. In *2024 IEEE Technology & Engineering Management Conference-Asia Pacific (TEMSCON-ASPAC)* (pp. 1–6). IEEE.
- [10] Simanjuntak, H. T. A., Simanungkalit, H. M., Tampubolon, E. R. A., & Natalia, T. D. (2023). Patients clustering on BPJS health insurance data using partition clustering algorithm. In *2023 International Conference of Computer Science and Information Technology (ICOSNIKOM)* (pp. 1–8). IEEE. <https://doi.org/10.1109/ICOSNIKOM60230.2023.10364367>
- [11] Raharjo, T., & Prayudi, Y. (2025). Securing electronic medical documents using AES and LZMA. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 9(2), 374–384.
- [12] Abdussalaam, F., & Oktaviani, I. (2020). Perancangan sistem informasi nilai berbasis web menggunakan metode prototyping. *Jurnal E-Komtek (Elektro-Komputer-Teknik)*, 4(1), 16–29. <https://doi.org/10.37339/e-komtek.v4i1.213>
- [13] Abdussalaam, F., & Badriansyah, B. (2021). Perancangan sistem informasi pemeriksaan barang berbasis web menggunakan metode SSAD. *EXPERT: Jurnal Manajemen Sistem Informasi dan Teknologi*, 11(2), 174–180. <https://doi.org/10.36448/expert.v11i2.2167>
- [14] Joel, A. E., Yunengsih, Y., & Abdussalaam, F. (2023). Perancangan sistem informasi pendaftaran pasien rawat jalan menggunakan Visual Studio 2010 di RSUD Al-Ihsan. *Jurnal Ilmiah Perkam dan Informasi Kesehatan Imelda (JIPIKI)*, 8(2), 143–155. <https://doi.org/10.52943/jipiki.v8i2.1292>
- [15] Wijayanti, T. C., Salim, B. S., Sucahyo, N., Widianoro, A. D., & Wibowo, E. W. (2024). Design of BPJS participant patient referral information system. *International Journal of Artificial Intelligence Research*, 6(1.1).