Patient regristration information system with auto reminder feature using the prototype method

Rizki Kurniawan*, Danur Wijayanto, Arizona Firdonsyah.

Information Technology Study Program, Faculty of Science and Technology, Universitas Aisyiyah Yogyakarta, Indonesia
*Email: rizkikur7676@gmail.com, danurwijayanto@unisayogya.ac.id, arizona@unisayogya.ac.id

Abstract
This research was conducted at Prima Dental Clinic, which still uses a manual patient registration process. The existing system causes long queues, potential data recording errors, and delays in service. Patients also often miss scheduled visits due to the lack of reminders from the clinic. To address these issues, this research aims to develop a web-based patient registration information system with an Auto Reminder feature that can automate the registration process and send appointment reminders automatically. This system was built using a prototype method that allows for intense interaction between developers and users, allowing the system to be continuously adjusted to meet needs. This system was built using the PHP programming language and MySQL database, and utilizes the Laravel framework, including the Laravel Scheduler feature to manage automatic reminder tasks and the WhatsApp API for the automatic reminder feature. System testing was conducted using Black Box Testing on 23 test scenarios, and the results showed that all features functioned properly with a 100% success rate. The 100% success rate proves that all system functionality has run well according to user expectations and needs.

Keywords: Auto Reminder; Information System; Laravel, Prototype; WhatsApp API

1. Introduction

The rapid development of information and communication technology has had a significant impact on various aspects of life, including healthcare services. Patient registration processes that are still carried out manually often result in long queues, data entry errors, and suboptimal service times (Asworowati et al., 2023; Goda, 2020). Timeliness in medical consultations or examinations is also critical to the success of patient care (Sreeramoju et al., 2020). In practice, however, many patients miss their scheduled visits due to the absence of a reminder system from the clinic.

To address these problems, this research aims to develop a web-based patient registration information system equipped with an auto-reminder feature. This feature is designed to automate the registration process and send visit reminders automatically to patients. The web-based approach was chosen because it can improve efficiency, reduce resource usage, and expand service coverage (Andriyan et al., 2020).

The system was developed using the prototype method, which allows intensive interaction between developers and users (Syarif & Risdiansyah, 2024). Through this approach, users can provide direct feedback, ensuring that the system meets their needs and expectations (Bacsafra et al., 2022). The iterative nature of the prototype method also supports the continuous adjustment of essential features, such as the auto-reminder, to optimize functionality. Consequently, the proposed system is expected to increase registration efficiency, minimize data errors, and enhance the overall quality of healthcare services.

An information system is a human-designed framework consisting of various organizational components with the primary goal of providing the required information (Pratama et al., 2021). It integrates information technology with the role of users who operate it, meaning that its success depends heavily on the synergy between the two (Mukhsin, 2020). Proper implementation enables organizations to present useful information, enhance knowledge, and minimize data processing errors (Putra et al., 2020).

Patient registration is the first step in the healthcare service process, particularly in medical records, with the purpose of organizing services and simplifying access to information (Handayuni, 2021). This procedure serves as a reference for registration officers to handle patients accurately, quickly, and efficiently (Silitonga & Purba, 2021). Moreover, patients' experiences during registration often influence their perception of the overall service quality of hospitals or clinics. Collecting demographic

and social information also plays an important role in supporting smooth healthcare delivery and achieving patient safety goals (Rohman, 2023).

A reminder is a tool that helps individuals recall important information and supports time management through time-based notifications or contextual notes (Amri, 2020). In healthcare, reminders serve as appointment alerts to ensure that patients do not miss their scheduled visits (Edde & Budayawan, 2021). An auto-reminder expands this concept by enabling the system to send notifications automatically, without manual intervention. By leveraging technology and system integration, auto-reminders can be scheduled to provide alerts at specific times or conditions, thereby improving patient attendance and the quality of care.

The prototype method is a system development approach that emphasizes building an initial model or temporary version of the system to give users an overview of how it works (Ulama et al., 2022). This process is carried out iteratively, involving users through direct feedback at each prototype version, allowing the system to be refined continuously (Siswidiyanto et al., 2020). The advantages of this method include its iterative nature, accelerated development, active user involvement, and reduced risk of mismatch between the final system and actual needs (Alda, 2023). In general, the prototype method includes need analysis, rapid design, prototype construction, user evaluation, and revisions until the system meets expectations (Syachroni & Mulyanto, 2022a).

Hypertext Preprocessor (PHP) is a widely used server-side programming language for web-based application development. As an open-source technology, PHP is freely accessible and directly integrates with HTML (Sari et al., 2022). Known for its flexibility, ease of learning, and broad community support, PHP continues to evolve to meet system development demands (Siahaan & Sianipar, 2020). It also provides extensive libraries and supports connections with popular databases such as MySQL and MariaDB. With proven performance and stability, PHP is used in systems ranging from small to large scale. One of the most popular PHP frameworks is Laravel, which offers a structured architecture to simplify and accelerate web application development (Supandi, 2022).

Black box testing is a software testing method focused on the functionality of an application without examining its internal code (Utomo et al., 2020). This approach is important because it simulates end-user interactions, making it easier to identify structural, visual, and functional errors (Parlika dkk., 2020). Its advantages include efficiency, cost-effectiveness, and the fact that it does not always require deep programming expertise (Trengginaz et al., 2020).

In this research, testing was carried out using the Guttman Scale, an instrument that employs binary 'Yes' or 'No' responses (Ulum, 2021). This scale is effective for evaluating functionality because it clearly indicates whether a feature works as expected. The testing results demonstrated a 100% success rate, proving that all system functions performed optimally without errors (Triyanto, 2020).

2. Method

The prototype method is a software development model that emphasizes building an initial design (prototype) to visualize the system being developed. The process begins with gathering user requirements, followed by creating an initial design that can be immediately evaluated by users. In this way, user feedback can be quickly accommodated until a system that meets their needs is achieved (Siswidiyanto et al., 2020).

This method was chosen in this research because it is considered effective for rapidly and iteratively adapting to user requirements. The stages include requirement analysis, rapid design, prototype construction, user evaluation, and testing. Thus, the prototype method allows the developed system to be more flexible and aligned with the needs of end-users (Syachroni & Mulyanto, 2022).

The stages of the prototype method are illustrated in Figure 1.

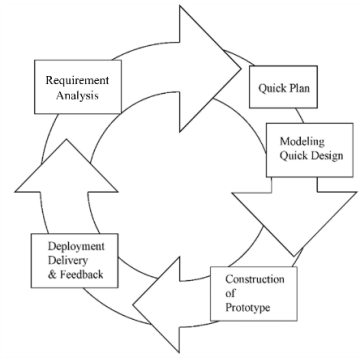

Figure 1. Prototype Development Stages

The manual patient registration process at Prima Dental Clinic presented several problems, including long queues, error-prone data recording, and delays in service. Moreover, the absence of a reminder system often caused patients to miss scheduled visits or consultations, leading to irregularities in medical services. These issues not only disadvantaged patients but also disrupted the effectiveness of healthcare workers in providing optimal services.

To address these problems, requirements were gathered through interviews, direct observation, and literature studies. A direct interview was conducted on Sunday, May 5, 2024, with drg. Prima Kusuma Wardhany, the dentist and owner of the clinic, to obtain detailed information about the registration workflow and patient data management. In addition, direct observations of the registration process and a review of relevant literature were carried out to strengthen the foundation of the system requirements. The results of these activities became the basis for formulating the functional and non-functional specifications of the system.

The functional requirements describe the core services that the system must provide. Based on the interview and observations, these include: (a) a homepage display; (b) a patient registration form with data storage capability; (c) an administrator login page; (d) a patient data display with create, read, update, and delete (CRUD) functions; (e) development using PHP and MySQL; and (f) an auto-reminder feature integrated with the WhatsApp API.

Meanwhile, the non-functional requirements specify the necessary hardware and software for system development. These requirements are summarized in Table 1 and Table 2.

Table 1. Hardware Requirements

No	Device Name	Details and Specifications
1	Laptop	Acer Aspire E5-575
2	Storage	SSD NVMe 256GB
3	RAM	4 GB
4	Processor	Intel Core i3

Table 2. Software Requirements

No	Software Name	Description
1	Sistem Operasi	Windows 10, basic platform for application development
2	Visual Studio Code	Main code editor for software development
3	Figma	Used for designing the user interface
4	Xampp	Used as a local server to test web-based applications
5	WhatsApp API	Used to integrate the auto-reminder feature

3. Results and Discussion

This section presents the design and implementation results of the patient registration information system with an auto-reminder feature developed using the prototype method. The discussion includes system modeling with Unified Modeling Language (UML), database design, user interface implementation, and system testing to ensure compliance with user requirements.

3.1. System Design

The system design was carried out to illustrate the process flow, user interactions, and data structures that support the patient registration information system with the auto-reminder feature. System modeling was performed using a Use Case Diagram to represent the main functionalities and an Activity Diagram to describe the process flow of each activity within the system.

3.1.1. Use Case Diagram

The Use Case Diagram serves as a guide in the system development process, ensuring that the construction and implementation of the patient registration system are more structured and minimize potential errors. This diagram illustrates the interactions between actors and the main functions of the system. The designed Use Case Diagram is shown in Figure 2.

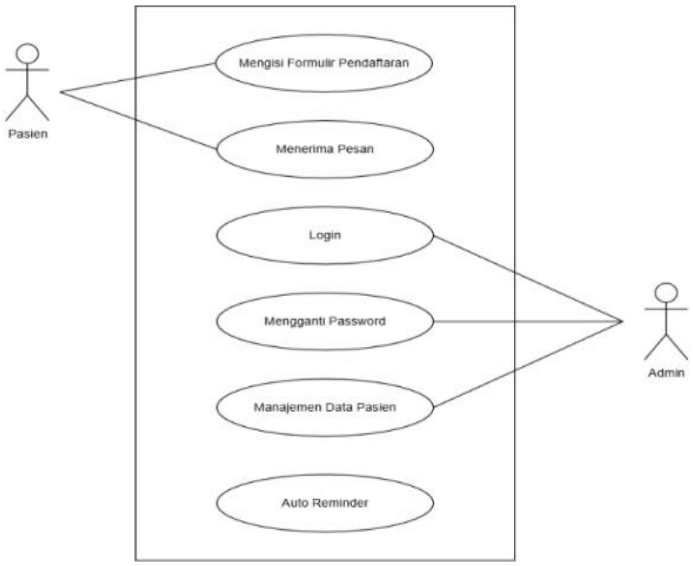

Figure 2. Use Case Diagram

3.1.2. Activity Diagram

a. Patient Registration

The activity diagram for patient registration illustrates the interactive workflow between the patient and the system. The process begins when the patient accesses the website, selects the registration option, and fills out the registration form. Once the data is submitted, the system automatically validates the input. If the data is valid, it is stored in the database, and the system provides confirmation through a pop-up notification and a WhatsApp message. If the data is invalid, the system directs the patient to correct the form entries, ensuring that all information is accurate before the registration process is considered complete. The activity diagram for patient registration is shown in Figure 3.

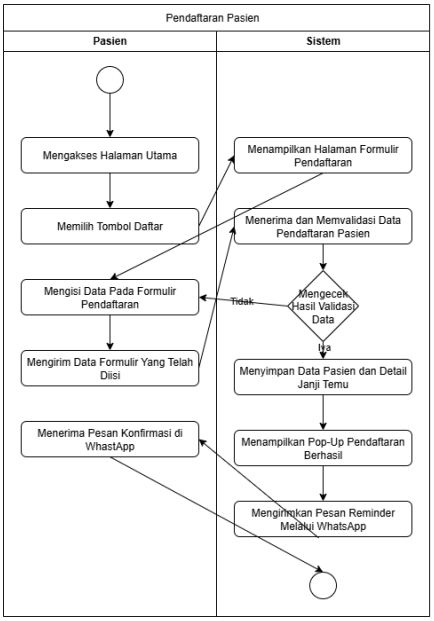

Figure 3. Activity Diagram of Patient Registration

b. Administrator Login and Password Recovery

This activity diagram explains the administrator authentication process, which is divided into two main flows: login and password recovery. The first flow describes the standard login procedure, where the administrator enters an email and password, after which the system verifies the credentials. If the login information is correct, the administrator is directed to the dashboard page. If the credentials are invalid, the system displays an error notification and provides the opportunity to retry. The second flow is initiated when the administrator selects the "Forgot Password" option. In this process, the system guides the administrator through a secure account recovery procedure, starting with email verification, sending a reset link, and setting a new password. Once the password has been successfully updated, the flow directs the administrator back to the login page to access the system using the new credentials. The login activity diagram is presented in Figure 4, while the password recovery activity diagram is shown in Figure 5.

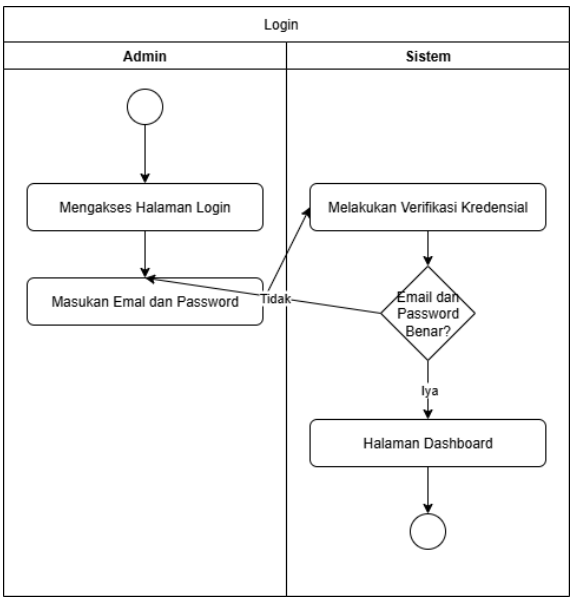

Figure 4. Activity Diagram of Administrator Login

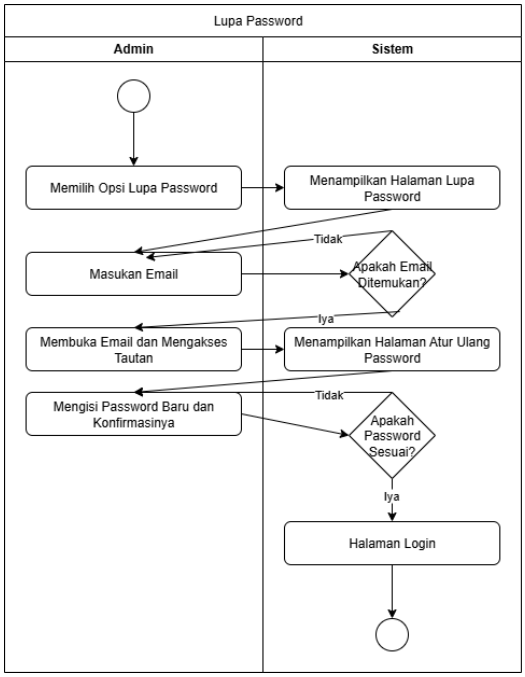

Figure 5. Activity Diagram of Password Recovery

c. Patient Data Management

The activity diagram for patient data management describes the series of actions that can be performed by the administrator after logging in to manage patient information through the dashboard. This process covers the core functionalities, including adding new patient data, updating existing information, deleting records, and searching data based on specific criteria. Each action selected by the administrator is processed by the system, which responds by displaying the appropriate form, performing data validation, updating the database, and presenting the updated information in the dashboard table. This diagram visually maps the dynamic workflow that enables administrators to efficiently manage the entire patient data lifecycle through a centralized interface. The activity diagram for patient data management is presented in Figure 6.

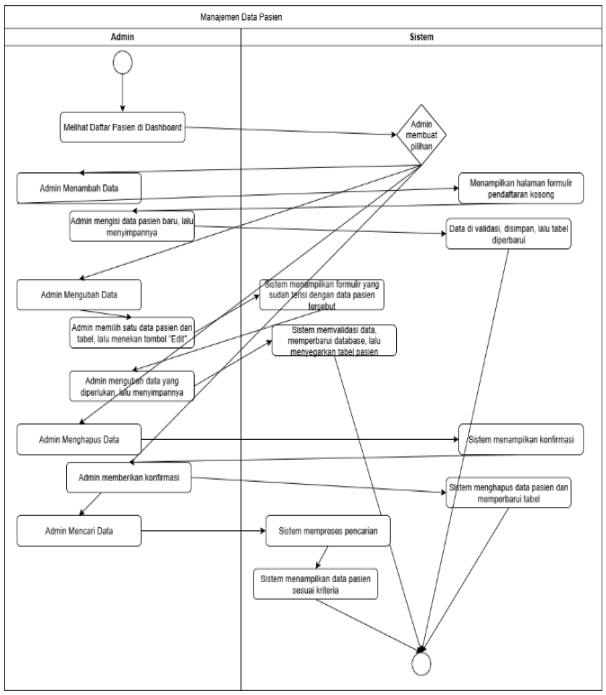

Figure 6. Activity Diagram of Patient Data Management

d. Auto Reminder

The activity diagram for the auto-reminder models a fully automated process executed by the system. The process is triggered by a scheduler that periodically checks for upcoming appointments. The system first searches for appointments scheduled five days in advance (H-5). If any are found, it iterates through them, sending reminder messages via the WhatsApp API and updating their status in the database. After completing this step, the process continues with a similar check for appointments scheduled one day in advance (H-1). This diagram illustrates a proactive and autonomous mechanism designed to reduce patient absenteeism by providing timely reminders without requiring manual intervention. The activity diagram for the auto-reminder is presented in Figure 7.

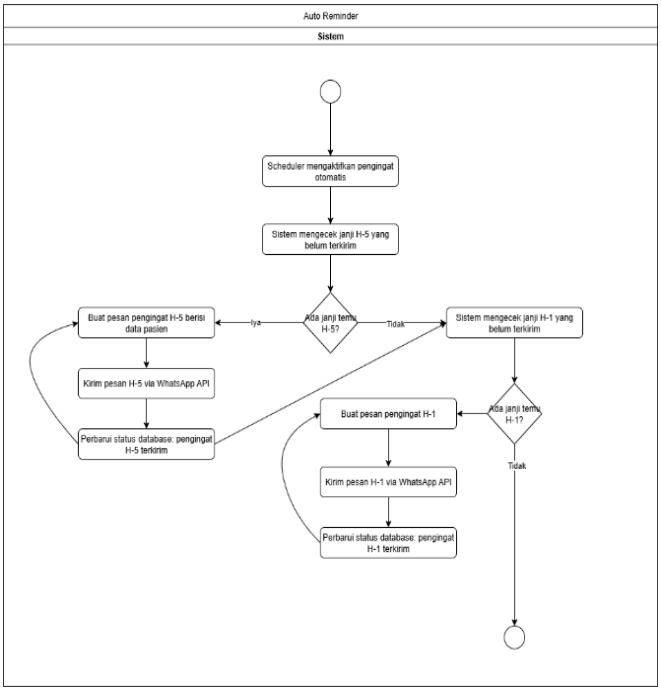

Figure 7. Activity Diagram of Auto Reminder

3.1.3. Entity Relationship Diagram

The Entity Relationship Diagram (ERD) was designed to illustrate the database structure used in the patient registration information system. The ERD consists of three main entities: patients, appointments, and administrators. The patient entity stores the basic information of registered patients, while the appointment entity records details of patient visits along with the automatic reminders. The administrator entity manages both patient and appointment data. The relationships between entities are clearly defined, where one patient can have more than one appointment (one-to-many relationship). Thus, the ERD provides a structured representation of data relationships and serves as the foundation for database design. The ERD is presented in Figure 8.

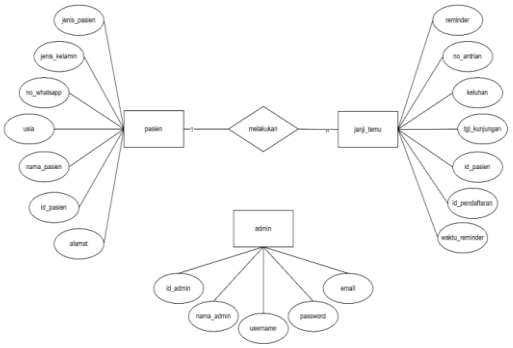

Figure 8. Entity Relationship Diagram

3.2. User Interface Implementation

The user interface implementation was carried out to transform the system design into a web-based application display. The interface was designed to be user-friendly and aligned with the needs of both patients and administrators. This section presents the main pages of the system, including the login page, patient registration form, administrator dashboard, and features for managing patient and appointment data.

3.2.1. Patient Registration Form Implementation

The patient registration form is the core component of the system used to register patients and schedule appointments with the dentist. On this page, patients are required to provide essential information such as name, address, WhatsApp number, visit date, age, gender, and medical complaint, while the queue number is generated automatically by the system. The interface of the patient registration form is presented in Figure 9.

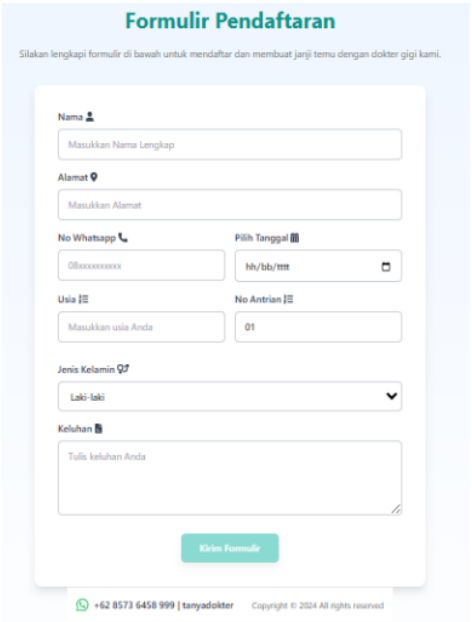

Figure 9. Patient Registration Form Interface

3.2.2. Administrator Login Page Implementation

The administrator login page serves as the entry point to access the system. On this page, administrators are required to enter their registered email and password to log in. The login form consists of two input fields Email and Password along with a Login button for authentication. If an error occurs, such as an incorrect password, the system displays an error message to prompt the administrator to correct the input. In addition, a Forgot Password feature is available, allowing administrators to recover their accounts in case of login issues. The interface of the administrator login page is shown in Figure 10.

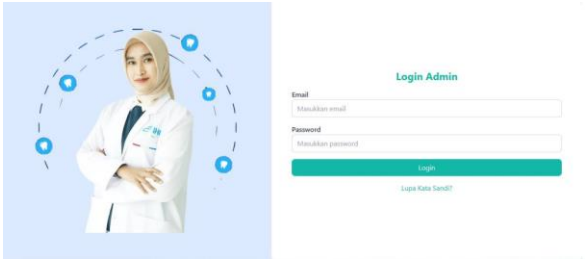

Figure 10. Administrator Login Page Interface

3.2.3. Forgot Password Page Implementation

The forgot password page functions as an account recovery tool for administrators who experience difficulties accessing the system. On this page, administrators are required to enter their registered email address. After clicking the Submit button, the system verifies the email. If the email is valid and registered, the system automatically sends a password reset link to the provided address. This link directs the administrator to the password reset page, where a new password can be created. With this feature, the account recovery process becomes more convenient while maintaining secure administrator access to the system. The interface of the forgot password page is shown in Figure 11.

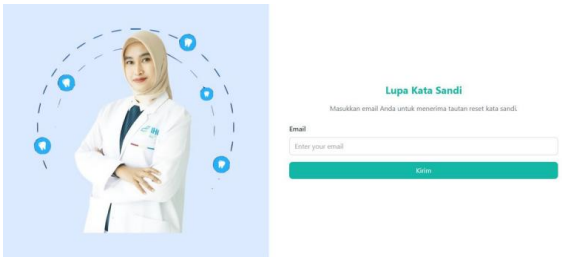

Figure 11. Forgot Password Page Interface

3.2.4. Reset Password Page Implementation

The reset password page is designed for administrators to change their password after receiving a reset link from the system. On this page, administrators are required to enter a new password in the Password field and confirm it in the Confirm New Password field. The system verifies the consistency of both inputs before updating the password data in the database. The confirmation process is completed by clicking the Reset Password button. Once successful, the system saves the new password and redirects the administrator to the login page, enabling access with the updated credentials. The interface of the reset password page is shown in Figure 12.

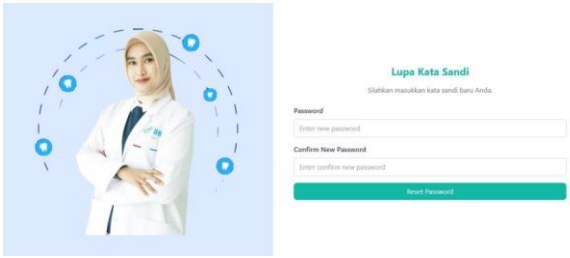

Figure 12. Reset Password Page Interface

3.2.5. Administrator Dashboard Implementation

The administrator dashboard functions as the central control panel for managing patient data registered in the system. On this page, patient data is displayed in an interactive table equipped with various management features. These include a Delete button (trash bin icon) to remove patient records, an Edit button (pencil icon) to update information, a Show Data button (eye icon) to view patient details, a Filter Data button (funnel icon) to filter records by visit date, and a Cancel Visit button (X icon) to cancel a patient's scheduled appointment. In addition, the dashboard provides an Add Data button for registering new patients and a Search feature that allows administrators to quickly find patient records based on name. With these features, the administrator dashboard ensures that patient data management can be carried out efficiently, quickly, and effectively. The interface of the administrator dashboard is shown in Figure 13.

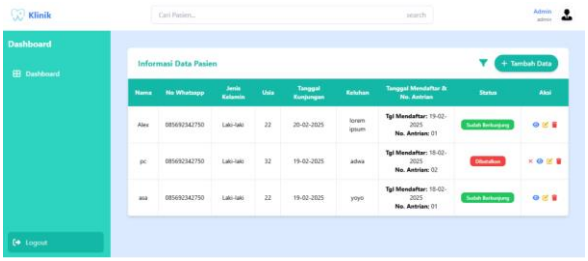

Figure 13. Administrator Dashboard Interface

3.2.6. WhatsApp API Implementation

The WhatsApp API integration feature functions to send automatic reminder messages to patients prior to their scheduled visits. The system automatically delivers reminders five days (H-5) and one day (H-1) before the appointment date selected by the patient. Each reminder message contains essential information, including the visit date and the queue number generated by the system. With this feature, patients receive direct reminders via WhatsApp, thereby minimizing the risk of forgetting their appointment schedule. The implementation of the WhatsApp API is shown in Figure 14.

Figure 14. WhatsApp API Implementation Interface

3.2.7. Logout Page Implementation

The logout page functions to confirm whether the administrator truly intends to exit the system. When the administrator selects the logout option, the system displays a confirmation pop-up with the message *"Are you sure?"* before ending the user session. This pop-up provides two buttons: Cancel, to abort the logout process, and Logout, to confirm exiting the system. In addition, an X button in the top-right corner can be used as an alternative to close the pop-up without logging out.

By including this confirmation feature, the system helps prevent accidental logouts and ensures secure use of the application. The logout page interface is shown in Figure 15.

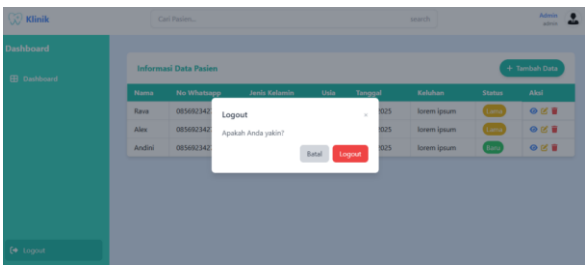

Figure 15. Logout Page Interface

3.3. System Testing

System testing was conducted using the black box testing method, a software testing approach that focuses on system functionality based on input and output without examining the source code. The purpose of this testing was to ensure that each feature in the Patient Registration Information System with Auto-Reminder Feature Using the Prototype Method operated in accordance with the specified requirements. A total of 23 test scenarios were executed, covering the main functionalities of the system, including patient registration, login, auto-reminder, administrator dashboard, and logout. Each test scenario involved the use of both valid and invalid inputs to observe whether the system responded according to the expected results. The testing scenarios are presented in Table 3.

Table 3. System Testing Results

No	Feature Tested	Input	Expected Output	Test Result
1	Appointment Registration	Complete and valid data	Registration successful	Passed
2	Appointment Registration	Empty name	Error message displayed	Passed
3	Appointment Registration	Invalid WhatsApp number	Error message displayed	Passed
4	Appointment Registration	Empty visit date	Error message displayed	Passed
5	Appointment Registration	Future visit date	Status set to “Not yet visited”	Passed
6	Login	Correct email and password	Login successful	Passed
7	Login	Incorrect email, correct password	Error message displayed	Passed
8	Login	Correct email, incorrect password	Error message displayed	Passed
9	Forgot Password	Valid email entered	Reset link sent	Passed
10	Forgot Password	Unregistered email	Error message displayed	Passed
11	Reset Password	Matching password and confirmation	Password successfully updated	Passed
12	Reset Password	Non-matching confirmation	Error message displayed	Passed
13	Admin Dashboard	Add patient data	Data successfully added	Passed
14	Admin Dashboard	Edit patient data	Data successfully updated	Passed
15	Admin Dashboard	Delete patient data	Data successfully deleted	Passed
16	Admin Dashboard	Search valid patient name	Data found	Passed
17	Admin Dashboard	Search invalid patient name	Data not found	Passed
18	Admin Dashboard	View patient details	Patient data displayed	Passed
19	Admin Dashboard	Click logout button	System exits and returns to login page	Passed
20	Reminder H-5	Patient has appointment	Notification sent	Passed
21	Reminder H-5	Invalid WhatsApp number	Notification failed	Passed
22	Reminder H-1	Patient has appointment	Notification sent	Passed
23	Reminder H-1	No scheduled appointment	No notification sent	Passed

Based on the testing results, all system features were successfully evaluated and demonstrated performance in accordance with expectations. The success rate was calculated using the following formula:

$$\text{Success Rate} = \frac{23}{23} \times 100\% = 100\%$$

Thus, it can be concluded that the system has met the expected functionality standards. A success rate of 100% falls into the "Very High" category according to the Guttman scale, indicating that all test scenarios were executed without any errors or functional failures. This demonstrates that the system is capable of operating optimally in accordance with the specified requirements.

4. Conclusion

The construction of a web-based patient registration information system with an auto-reminder feature was successfully carried out using the prototype method. The construction process included requirement analysis, design, and implementation utilizing PHP, MySQL, the Laravel framework, as well as Laravel Scheduler and the WhatsApp API for automated reminders. System testing using the Black Box Testing method across 23 scenarios demonstrated a 100% success rate, indicating that all functionalities operated optimally in accordance with user requirements.

For future construction, the system can be improved by adding a rescheduling feature that allows patients to modify their appointment dates without re-registering, and by integrating with an electronic medical record system so that patient data is directly linked to medical notes. The auto-reminder feature may also be refined by including alternative reminder channels such as email or SMS, as well as an administrative dashboard with statistical insights, including daily, weekly, and monthly patient counts and visit trends. Another potential refinement is the addition of a patient attendance confirmation feature, enabling administrators to track appointment attendance and reschedule efficiently when necessary.

5. Acknowledgements

The author would like to express sincere gratitude to the supervisors at Universitas Aisyiyah Yogyakarta for their valuable guidance and support throughout this research. Appreciation is also extended to Klinik Gigi Prima as the research partner for providing access and practical insights essential to this research.

References

Alda, M. (2023). Pengembangan Aplikasi Pengolahan Data Siswa Berbasis Android Menggunakan Metode Prototyping. Jurnal Manajemen Informatika (JAMIKA), 13(1), 11–23.

Amri, I. (2020). Sistem E-Reminder Service Pada Aplikasi Manajemen Keuangan Pondok Pesantren Miftahussalam Menggunakan Whatsapp Gateway. University of Technology Yogyakarta.

Andriyan, W., Septiawan, S. S., & Aulya, A. (2020). Perancangan Website sebagai Media Informasi dan Peningkatan Citra Pada SMK Dewi Sartika. Jurnal Teknologi Terpadu, 6(2), 79–88. https://doi.org/10.54914/jtt.v6i2.289

Asworowati, R. D., Mustomi, D., Adawia, P. R., Suhendra, A. D., Natong, A., & Ningrum, M. C. (2023). Perancangan Sistem Informasi Pendaftaran Pasien Rawat Jalan Pada Rumah Sakit Ibu dan Anak Gizar Berbasis Mobile. Jurnal Teknologi dan Sistem Informasi Bisnis, 5(4), 542–549.

Bacsafra, M. A., Kusumawardani, D. M., & Darmansah, D. (2022). Pengembangan Sistem Informasi Badan Pusat Statistik Kabupaten Kuningan Berbasis Android Dengan Metode Prototype. J-SAKTI (Jurnal Sains Komputer Dan Informatika), 6(1), 379–390.

Edde, G. P., & Budayawan, K. (2021). Pembuatan Aplikasi Reminder Jadwal Perkuliahan di Jurusan Teknik Elektronika Berbasis Android. Voteteknika (Vocational Teknik Elektronika Dan Informatika), 9(4), 1–11.

Goda, M. M. (2020). Perancangan Sistem Informasi Pendaftaran Pasien Berbasis Web Dengan Fingerprint Di Puskesmas. Jurnal Fasilkom, 10(3), 199–208.

Handayuni, L. (2021). Faktor-Faktor yang Mempengaruhi Terhambatnya Proses Pendaftaran Pasien Berdasarkan E-Puskesmas. Jurnal Manajemen Informasi Kesehatan Indonesia, 9(2), 129. https://doi.org/10.33560/jmiki.v9i2.326

Mukhsin, M. (2020). Peranan teknologi informasi dan komunikasi menerapkan sistem informasi desa dalam publikasi informasi desa di era globalisasi. Teknokom, 3(1), 7–15.

Parlika, R., Nisaaâ, T. A., Ningrum, S. M., & Haque, B. A. (2020). Studi Literatur Kekurangan dan Kelebihan Pengujian Black Box. Teknomatika, 10(2), 131–140.

Pratama, A., Fachrurrazi, S., & Ula, M. (t.t.). Perancangan Dan Aplikasi Model Sistem Informasi Sekolah. Sisfo: Jurnal Ilmiah Sistem Informasi, 5(1), 2021.

Putra, D. M., Yasli, D. Z., Leonard, D., & Yulia, Y. (2020). Penerapan Sistem Informasi Manajemen Puskesmas (SIM-PUS) Pada Unit Rekam Medis Dan Informasi Kesehatan Di Puskesmas Lubuk Buaya Kota Padang. J. Abdimas Saintika, 2(2), 67–72.

Rohman, R. (2023). PMIK Berani Bertransformasi. Penerbit NEM.

Sari, I. P., Jannah, A., Meuraxa, A. M., Syahfitri, A., & Omar, R. (2022). Perancangan Sistem Informasi Penginputan Database Mahasiswa Berbasis Web. Hello World Jurnal Ilmu Komputer, 1(2), 106–110.

Siahaan, V., & Sianipar, R. H. (2020). TWO BOOKS IN ONE: PHP/MYSQL Untuk Pemula dan Programmer. SPARTA PUBLISHING. https://books.google.co.id/books?id=mz7ODwAAQBAJ

Silitonga, P. D. P., & Purba, D. E. R. (2021). Implementasi System Development Life Cycle Pada Rancang Bangun Sistem Pendaftaran Pasien Berbasis Web. Jurnal Sistem Informasi Kaputama (JSIK), 5(2), 196–203.

Siswidiyanto, S., Munif, A., Wijayanti, D., & Haryadi, E. (2020). Sistem Informasi Penyewaan Rumah Kontrakan Berbasis Web Dengan Menggunakan Metode Prototype. Jurnal Interkom, 15(1), 18–25. https://doi.org/10.35969/interkom.v15i1.64

Siswidiyanto, S., Wijayanti, D., & Haryadi, E. (2020). Sistem Informasi Penyewaan Rumah Kontrakan Berbasis Web Dengan Menggunakan Metode Prototype. Jurnal Interkom, 15(1), 16–23.

Sreeramoju, P. V., Weber, S. G., Snyder, A. A., Kirk, L. M., Reed, W. G., & Hardy-Decuir, B. A. (2020). The patient and health care system: Perspectives on high-quality care. Springer.

Supandi, R. R. (2022). BISA PHP DALAM 1 HARI (Issue v. 1). Ronald Robi Supandi. https://books.google.co.id/books?id=rG-HEAAAQBAJ

Syachroni, W., & Mulyanto, A. (2022a). Penerapan Metode Prototype Dalam Perancangan Sistem Administrasi Tpu Desa Karang Setia Berbasis Web. Jurnal Informatika SIMANTIK, 7(2), 63–66.

Syachroni, W., & Mulyanto, A. (2022b). Penerapan Metode Prototype Dalam Perancangan Sistem Administrasi Tpu Desa Karang Setia Berbasis Web. Jurnal Informatika SIMANTIK, 7(2), 63–66.

Syarif, M., & Risdiansyah, D. (2024). Pemanfaatan Metode Prototype dalam Perancangan Sistem Informasi Penjualan Berbasis Website. JATI (Jurnal Mahasiswa Teknik Informatika), 8(4), 7945–7952.

Trengginaz, R. B., Yusup, A., Jihad, M. R., Sunyoto, D. S., & Yulianti, Y. (2020). Pengujian aplikasi pemesanan tiket kereta berbasis website menggunakan metode black box dengan teknik equivalence partitioning. J. Teknol. Sist. Inf. Dan Apl., 2654(3788).

Triyanto. (2020). Implementasi Metode Black Box Testing pada Aplikasi Manajemen Proyek berbasis Web. Jurnal Teknologi dan Sistem Informasi Bisnis, 2(1), 12–18.

Ulama, E. K., Priandika, A. T., & Ariany, F. (2022). Sistem Pendukung Keputusan Pemilihan Sapi Siap Jual (Ternak Sapi Lembu Jaya Lestari Lampung Tengah) Menggunakan Metode Saw. Jurnal Informatika Dan Rekayasa Perangkat Lunak, 3(2), 138–144.

Ulum, M. (2021). Aplikasi Berbasis Web untuk Pengujian Kecepatan Membaca Qur'an Metode Tartiil. Jurnal Ilmiah Teknologi Informasi Terapan, 8(2), 11–18.

Utomo, A., Sutanto, Y., Tiningrum, E., & Susilowati, E. M. (2020). Pengujian Aplikasi Transaksi Perdagangan Menggunakan Black Box Testing Boundary Value Analysis. Jurnal Bisnis Terapan, 4(2), 133–140.