ISSN 1846-6168 (Print), ISSN 1848-5588 (Online)
https://doi.org/10.31803/tg-20250430172617

Original scientific paper
Received: 2025-04-30, Accepted: 2025-05-21

Automation for Patient Medical Records in an Integrated Clinic Geographic Information System

Lely Prananingrum, Teuku Salman Farizi, Fajar Agus Dwi Rahmawan, Ilmiyati Sari*

Abstract: Majority clinics are dependent on traditional technique of managing medical records patients. This consequently makes the operations less efficient and patients wait longer. The objective of this study is the implementation of an electronic clinic management system with the introduction of the geographical information system (GIS) using the grounded theory, rapid application development (RAD) and agile approaches respectively. it includes GIS applications for clinic location finding, user account management processes such as login and registration, patient information management and reports, system support, and report generation. The RAD approach was utilized to fast-track development processes employing iterative patterns and the Agile approach ensured that the system suited the users' needs as they changed over time. The results have shown that this automated clinic information system is efficient in managing healthcare information by making more efficient use of existing information resources, balancing timeliness with processing time, and enhancing end-user satisfaction.

Keywords: agile; automation; GIS; grounded research; medical; record

1 INTRODUCTION

Improving the efficiency and quality of health services in clinics has become increasingly important along with the rapid development of information technology. One of the crucial elements in health services is the management of patient medical records, which act as a basis for making appropriate and accurate medical decisions. Medical records are important documents owned by health care facilities and function to record all information related to the patient's condition and treatment. Medical record documents must be managed properly, because health facilities are responsible for their integrity, security, and protection against loss, damage, forgery, or use by unauthorized parties. Although medical record documents are owned by health facilities, the contents of the medical records belong to the patient. The contents of these medical records consist of patient identity, results of physical and supporting examinations, diagnosis, treatment, follow-up plans, and signatures of health workers who provide further services. Information in medical records can be conveyed to patients, their families, or other parties who have received the patient's consent, except in certain conditions such as underage patients or in emergencies [1].

However, in practice, a number of clinics still use manual methods in managing medical records. This condition raises various problems, including difficulty in searching for data, duplication of information, and the risk of losing documents due to limited storage media. A clinic is a health service facility that provides basic or specialist medical services. Based on the type of service, clinics are divided into primary clinics, which provide basic medical services, and principal clinics, which provide specialist medical services or a combination of the two. These clinics can be owned by the government, local government, or the community, and can focus on a particular field based on a particular branch of science or organ system [2].

The manual recording system used by most clinics in recording medical records often causes problems such as data duplication, recording errors, and difficulties in searching for patient data. The service flow that includes registration, examination, and administration is often hampered by the slow and error-prone manual recording process. In its operations, the clinic must also document patient data and manage medical record files to provide administrative and medical information. Medical records are made in electronic form to prevent document loss, save storage space, and avoid inconsistencies in filling [3].

Therefore, automation of medical records through information technology is a relevant solution. Electronic medical records can overcome these problems by avoiding inconsistencies and loss of documents, as well as saving storage space. In addition, GIS, or Geographic Information Systems, is an application that processes spatial data using a computerized system, combining graphic data with object attributes through digital base maps that refer to the earth's surface [4]. GIS is designed to process information derived from various data sources, including geographic data related to the position of objects on the Earth's surface. GIS technology combines database-based data processing that can be accessed today with distinctive visualization through maps [5].

Integration of geographical information systems (GIS) with medical record automation provides a solution that improves the clinic's internal efficiency and makes it easier for the public to find health facilities that suit their needs. This study provides an innovative solution by developing a medical record automation system integrated with GIS. This system will help clinics manage patient data more effectively, reduce data redundancy, and speed up the service process.

In previous literature reviews, most studies have highlighted the importance of automating medical records and using GIS separately. Nevertheless, the novelty of this research is the combination of both aspects in one system designed using the Agile method and Grounded Research. Such an approach enables a more needs-driven system development with broader applicability across clinics and relevant scales. With this system, clinics are also expected to be more competitive and contribute to improving the quality of health services in Indonesia.

2 RESEARCH METHOD

Research methods are stages or steps researchers take to collect information or data and conduct investigations on the data obtained. The research methods in this study are:

2.1 Research Approach

The method used in this research is Grounded Research and Agile, two different approaches that support each other in the process of developing information technology-based systems. Grounded Research was originally developed by two sociologists, Barney Glaser and Anselm Strauss [6]. Grounded Theory is based on the principle that theories derived from field data are more accurate and relevant than theories based on hypotheses or speculation. In this method, experts begin with original pieces of data collected from interviews or observations, then, step by step, create theories [6]. The Grounded Theory approach works its way from practically the empirical stage to the conceptual-theoretical or in more precise terms aims to develop theories from the available data (Kesa & Sainuddin, 2020) [7]. The main goal of Grounded Theory is to expand the understanding of a phenomenon by identifying the key elements of that phenomenon, and then grouping the relationships between those elements within the context and process of the experiment. In other words, the goal is to move from the general to the more specific without neglecting the unique characteristics of the subject being studied [8].

Agile method is an approach in software development that is based on short development cycles. This approach provides flexibility for developers to quickly adapt to changes that occur [9]. The stages in the Agile method include: Plan, Design, Develop, Test, Deploy, Review, and Launch [10]. This study uses the Agile method because this method gives the development team the ability to adapt to changing needs and priorities during development, so that the team can respond quickly to changes or customer requests [11].

2.2 Method of Collecting Data

Data collection methods used in this study are:

1) Observation. The researcher made field observations offline with the aim of gathering data on clinical activities, notably at the clinic of Dr. Aris Rasidi Dahlan in Manggarai, South Jakarta and other clinics of Yogyakarta and Makassar. This observation was conducted to understand the workflow and management of the clinic directly, especially related to the administration and medical record processes that are still carried out manually. In addition, the observation aims to identify the potential for implementing an automation system that can reduce data redundancy and increase the efficiency of clinic administration processing. Additional data that was observed involved the actions related to documentation and retrieval of patient data, as well as the activities involving the clinic staff and the operation system in place.

2) Interviews. In order to gain deep insight into the requirements of the system which is under design, interviews were held not only with clinic employees and doctors but also with patients The focus of the interviews included collecting data regarding obstacles faced in the manual process, such as the length of patient waiting time and difficulties in searching for medical record data. Interviews were also conducted using focus group discussions (FGD) with the resource persons from the clinics located in three different places, namely Jakarta, Yogyakarta, and Makassar. The investigator inquired about the ways in which the use of computerized medical record systems can enhance overall clinical activities and assist in minimizing data errors and delays.

3) Literature Study. A literature study was conducted in the course of this writing by reviewing various literature and references, and also obtaining relevant books and articles from the internet.

2.3 System Development Methods

The approach used in the system development in this research is Rapid Application Development (RAD), a subclass in the System Development Life Cycle (SDLC) model [12]. Rapid application development is a software development method that allows for short development turnaround times by focusing on prototyping and quick building up, followed by the development process [13]. The information system design process usually takes an average of five months, but with this technique, the software system can only take three to four months [14].

The reason why implementing the Rapid Application Development (RAD) approach is essential is that this approach has some benefits, such as enabling a shorter development cycle, being more adaptive, increasing user involvement, decreasing the risks of making mistakes [12].

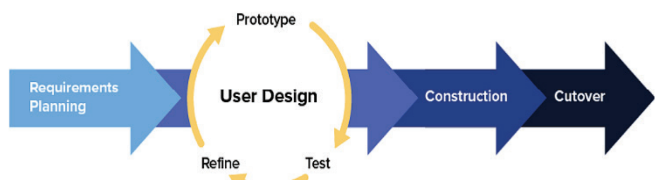

Figure 1 Rapid Application Development (RAD)

In this model, there are several stages of system development, namely:

1) Requirements Planning. At this stage, researchers and users hold discussions together to understand and analyze existing problems. Together, determine the needs that must be met in developing the application system. This stage is very important because it is the first step to ensure the success of the system to be developed, as well as preventing miscommunication between researchers and users.

2) User Design. At this stage, researchers design a system based on agreed needs, with the aim that the resulting solution is able to overcome the problems faced. In this study, system design is represented using a tool such as Unified Modeling Language (UML) and Figma in order to ensure that the presented design meets user requirements.

3) Construction (Development). At this stage, researchers start to realize the established design in a real life application. Researchers write program code, or what is often referred to as the coding process, to translate the system design into functional software according to the previously made plan.

4) Cutover (Implementation). In this final stage, the entire system that has been developed is thoroughly tested to ensure that all its components function properly. Researchers conduct testing using Black Box Testing, a testing technique that focuses on validating software functionality based on predetermined specifications, in order to reduce the risk of errors or defects in the system.

3 RESULTS AND DISCUSSION

3.1 Design System

In designing a practical and easy-to-understand information system, a tool that can visually represent various aspects of the system is needed. One approach used in this study is the Unified Modeling Language (UML). UML is a series of tools commonly used to represent an object-based system or software [15]. The author can define system requirements through UML, perform analysis and design, and describe the overall architecture. This tool provides various types of diagrams that help visualize multiple aspects of the system in several models. However, in this study, the author focuses the design on use case diagrams to describe user interactions with the system and sequence diagrams to show the process flow in the system in detail. Thus, these two diagrams are expected to provide a more structured and comprehensive view of the system being developed.

1) Functional Requirements Analysis. To ensure the system runs according to the expected goals, the author identified several functional requirements that need to be met. These requirements are summarized in Tab. 1.

Table 1 Functional Requirements

No.	Functional Requirements	What the Actor Did
1.	The system must first log in before it can be accessed	Officers and doctors log in first
2.	The system must be able to accept new patient registrations	Officers register new patients
3.	The system must be able to record patient medical records	Doctors record patient medical records
4.	The system must be able to print patient medical records	Officers or doctors print patient medical records
5.	The system must be able to display the location of the registered clinic	The person in charge of the clinic provides the location of the clinic during account registration

2) Use Case Diagram. The use case is one of the ways of putting together a series of activities or more than one which are related to each other and constitute an organized activity bearing out or being managed by an actor [16]. Use case diagrams (Fig. 2) do not explain in detail the use of use cases, but only provide a general description of the relationship between use cases, actors, and systems. Through this use case, the functions in the system can be identified [17].

3) Sequence Diagram. Sequence diagram (Fig. 3) is a diagram designed to study the flow of interactions between objects [18]. The main components of a sequence diagram include objects depicted in rectangular boxes with names, messages represented by lines with arrows, and time displayed in a vertical process [19].

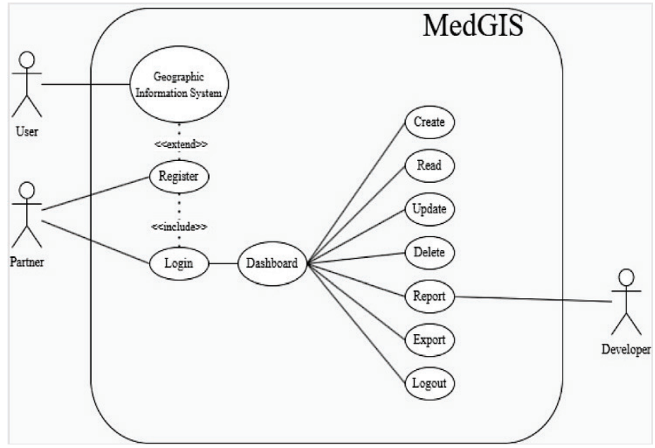

Figure 2 Use Case Diagram

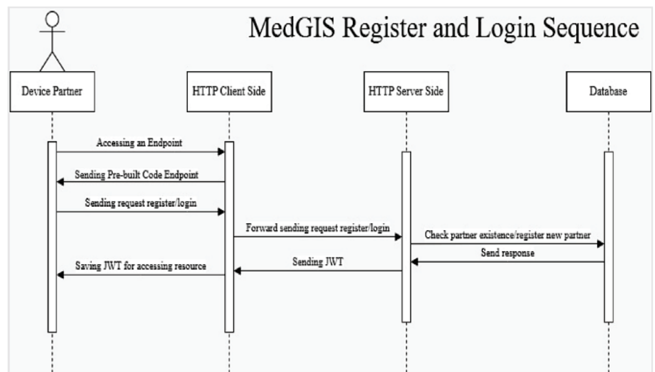

Figure 3 Sequence Diagram

3.2 Implementation

Fig. 4 is pages that are provided in the application.

Fig. 4(a) is used to show the location of clinics in an area. This map provides a geographic visualization with several clinic markers indicating the location of a particular clinic. In the middle of the map is pop-up information that presents detailed data on a clinic, such as the clinic name, address, and contact information. In addition, a "Join as a Partner" option at the top right of the page gives users the opportunity to join as a partner.

Fig. 4(b) is the registration page for clinics that want to join the platform. Users must fill in some information such as the clinic name, telephone number, name of the person in charge, email address, and password. Registering the current location as the clinic location is also available. After all the data is filled in, the user can press the "Register" button to complete the registration process. In addition, users can also access the login option if they already have an account. Fig. 4(c) shows the application screen, in which a user is required to log in by providing an email and password. Below the form, a provision to enroll or return to the home page is provided.

Figs. 4(a)–4(c) present the entry points to the system—clinic map, registration, and login—each framed by a persistent left-hand panel containing the project logo and tagline ("Terhubung dengan 300+ Klinik yang Telah Bergabung"). From the Folium-generated map embedded via iframe (Fig. 4(a)) to the Register (Fig. 4(b)) and Login (Fig. 4(c)) forms, the same purple accent color, rounded form fields, and primary-action button style are applied. Input labels are left-aligned, placeholders use consistent capitalization, and form controls maintain uniform padding and border radius, ensuring that users immediately recognize navigational and interactive elements regardless of context.

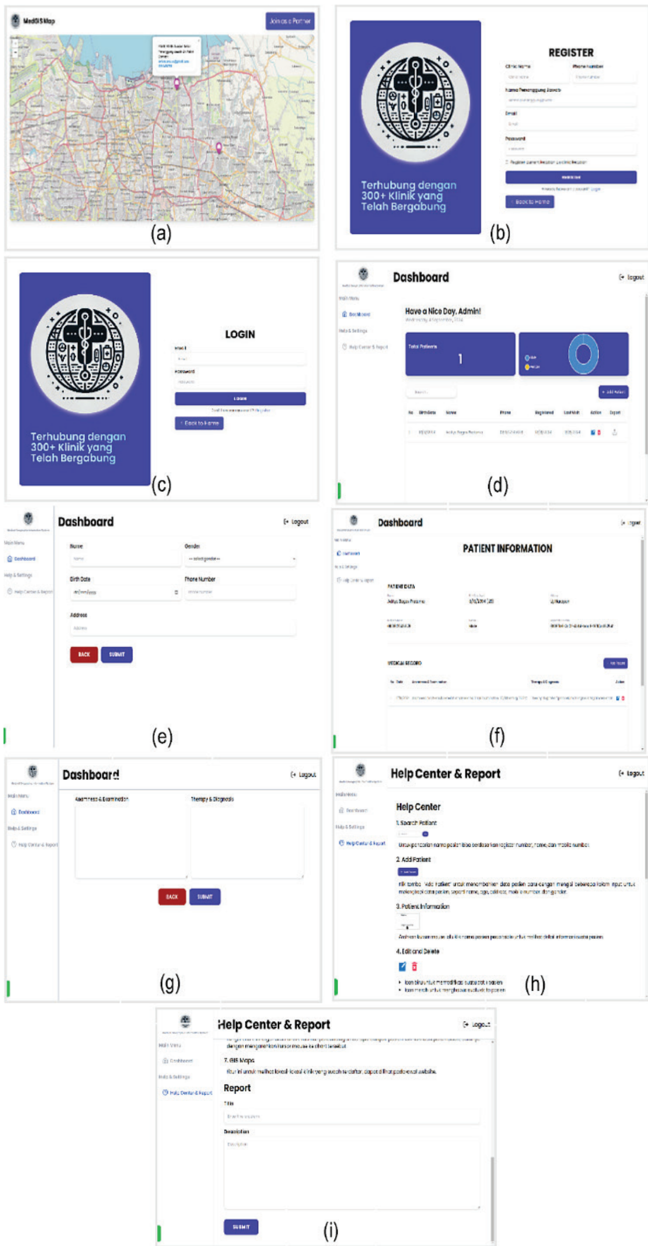

Figure 4 Clinic Medical Record System

Fig. 4(d) shows the post-login dashboard, where the top navigation bar retains the purple highlight and houses the greeting ("Have a Nice Day, Admin!") alongside the logout link. Key performance indicators are laid out in modular cards with matching drop shadows and corner rounding; the gender-split doughnut chart reuses the primary and secondary palette to reinforce visual coherence with earlier screens. The patient table below employs alternating row shading and action icons that follow the same color-on-white pattern established in the login and registration buttons.

Figs. 4(e) and 4(f) depict patient-management pages that continue the established grid layout: labels and inputs share spacing and alignment conventions from the registration form, while section headings and fieldsets use the same font size and weight. Buttons for "Back" and "Submit" consistently appear at the form's lower right, with the primary (Submit) button filled in purple and the secondary (Back) button outlined in the same hue.

Fig. 4(g) follows the identical card-and-form pattern when adding or updating medical record details. Text areas for anamnesis and diagnosis adopt the same line height and border styling as other multi-line inputs. Action controls for saving or discarding changes mirror the button hierarchy and placement seen throughout the application.

Figs. 4(h) and 4(i)—the Help Centre and Problem Report pages—extend the sidebar navigation style used on every page, with link items adopting the same hover and active states. Content panels continue to use white backgrounds, consistent typography, and iconography aligned to the left of each instruction, guaranteeing that guidance and feedback screens feel like an integral part of the same application environment.

3.3 System Architecture

Section 3.3 System Architecture is organized around three Django ORM models Clinic, Patient, and MedicalRecord exposed via Django REST Framework as RESTful API endpoints. The Clinic model uses a UUID primary key and stores clinic name, contact details (phone, email), a Person-In-Charge (PIC) field, authentication credentials, and geospatial coordinates (latitude, longitude). Adopting UUIDs ensures unique identification across Indonesia's highly distributed network of clinics, while the PIC attribute reflects the common operational practice of assigning a dedicated staff member per facility. Latitude and longitude are held as CharFields to feed directly into a custom Django GIS module built with Folium, enabling interactive mapping of clinics in both urban and remote areas.

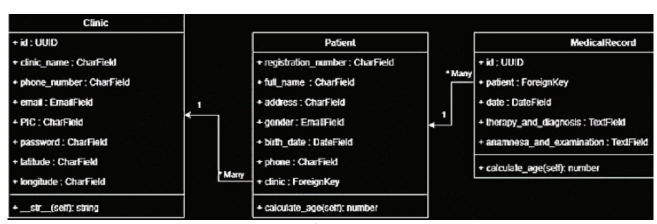

Figure 5 Class Diagram

The Patient model captures a unique registration number, full name, address, gender, birth date, and phone number, and links each patient to its clinic via a ForeignKey mirroring registration workflows puskesmas. The MedicalRecord model records its own UUID, references the Patient, logs the date of service, and stores clinical notes (therapy_and_diagnosis, anamnesa_and_examination). Both Patient and MedicalRecord classes implement a calculate_age() method to derive patient age dynamically from the stored dates.

A SQLite database serves as the relational backend, chosen for its simplicity in initial development, and all services are containerized to guarantee consistent deployment across varied infrastructure setups. The combination of Django, DRF, REST API, Folium-based GIS, and containerization directly addresses Indonesia's requirements for unique, location-aware clinic management and scalable service delivery.

Immediately following the registration of a new Clinic instance via the Django admin or registration endpoint, a bespoke GIS module reads the stored latitude and longitude values for all clinics. It then instantiates a Folium Map object, adds a marker for each clinic coordinate, and generates an HTML snippet representing the interactive map. Because Folium outputs self-contained HTML, this snippet is embedded into the web interface via an <iframe> element. The iframe approach ensures that the map renders consistently across browsers without requiring additional front-end dependencies—simply displaying the up-to-date positions of all registered clinics on the website.

3.4 Interoperability

Interoperability is achieved through a RESTful API layer secured by JSON Web Tokens (JWT), allowing each module—clinic registration, patient management, and medical records—to operate independently while sharing a common authentication mechanism. Upon successful login, the system issues a signed JWT that clients include as a Bearer token in the HTTP Authorization header for all subsequent requests. This stateless approach eliminates the need for a centralized session store, enabling microservice-style modularity: each service can validate request integrity and user identity locally without cross-module dependencies. Moreover, because JWT payloads can carry custom claims (e.g., clinic ID, user role), downstream systems and potential third-party integrations can enforce fine-grained access control and data partitioning without bespoke authentication code. By standardizing on JWT, the architecture supports seamless extension—whether integrating with external EHR platforms via secure API calls or deploying new front-end clients—while maintaining clear module boundaries and minimizing coupling.

3.5 Testing

In ensuring that the software meets the functions that have been set, testing is needed that focuses on validating the functionality of the system. One of the methods used in this study is black-box testing. Black-box testing is one way to test software that has been developed, both in small units and integrated systems to ensure the functionality of the software. This testing is carried out based on functional specifications without looking at the design or program code, with the aim of evaluating whether the functions, input and output of the software comply with the expected specifications [20]. The tester does not have to have knowledge of any particular programming language, as this method is designed for novice users [21]. Tab. 2 shows black box testing of application that is proposed in this research.

Table 2 Black Box Testing

No	Test Case	Expected Output	Status (Pass/Fail)
1.	Login	Successfully logged in, then switched to the dashboard page	Pass
2.	Register	Successfully registered, then switched to the dashboard page	Pass
3.	Add Patient Data	Patient data successfully added	Pass
4.	Search Patient Data	Relevant patient data displayed	Pass
5.	Delete Patient Data	Patient data successfully deleted	Pass
6.	Change Patient Data	Patient data successfully changed	Pass
7.	Add Patient Medical Record Data	Patient medical records successfully added	Pass
8.	Delete Patient Medical Record Data	Patient medical records successfully deleted	Pass
9.	Change Patient Medical Record Data	Patient medical records successfully changed	Pass
10.	Print Patient Medical Record Data	Patient medical records successfully displayed as a PDF file	Pass

Below is a summary of response sizes and execution times for three key endpoints, to be inserted immediately after the black-box testing section.

Table 3 Summary of response

Endpoint	Response Size	Execution Time
Login	901 B	631 ms
Map	10 KB	24 ms
Fetch single patient record	803 B	28 ms

Following the black-box testing results, Tab. 3 highlights the performance characteristics of representative API calls. The Login endpoint incurs a higher latency (631 ms) despite a modest payload (901 B) because it must perform cryptographic operations to generate and sign a JSON Web Token for secure, stateless authentication. In contrast, the Map endpoint returns roughly 10 KB in just 24 ms: this larger payload size reflects the self-contained HTML snippet produced by Folium for embedding the interactive clinic map via an <iframe> on the client. Finally, retrieving a full patient record yields an 803 B response in 28 ms, indicating efficient serialization of both patient fields and associated medical records under normal conditions.

4 CONCLUSIONS

This study successfully built a clinical medical record information system integrated with GIS using RAD and Agile methods. The implementation of this system is able to increase the efficiency of medical record maintenance by reducing data redundancy and accelerating the service process in the clinic. In addition, the integration of GIS in this system makes it easier for the public to access information related to the location of the clinic that suits their needs, thus expanding the reach of health services.

Testing conducted using the black-box method shows that this system is able to function well in running key features such as login, registration, patient data management, medical records, and GIS map visualization. This system is expected to be a solution for clinics in facing the challenges of increasing patient numbers and the need for more efficient and responsive health services.

Overall, this research provides a significant contribution to the development of clinical information system applications that not only focus on automating medical records but also maximizing the potential of GIS to improve the quality of health services.

For further development, it is recommended that this system integrate artificial intelligence (AI) and blockchain technology. AI integration can be used to improve accuracy and efficiency in managing medical records and provide smarter data-based medical recommendations. For example, AI can help doctors analyze a patient's medical history to provide health predictions or early diagnoses based on existing data patterns, thereby accelerating clinical decision-making.

In addition, the implementation of blockchain technology can provide an additional layer of security to ensure the integrity and safety of patient medical data. Blockchain, with its decentralized and transparent nature, can keep medical records from being altered without proper authorization. Every data change can be recorded in a distributed ledger system, ensuring that medical records remain secure and traceable.

However, in this development, the aspect of administrative processes must not be overlooked. The use of AI and blockchain technology must remain aligned with administrative needs such as patient registration management, data recording, and payment processes. The integration of electronic payment systems can also be enhanced to support the smoothness of administrative transactions. Thus, the system can provide comprehensive solutions, not only from a clinical perspective but also from an administrative one.

Further testing on a larger scale and involving more clinics in various locations is also recommended to ensure that the newly integrated technology can function effectively and provide real benefits in healthcare practice.

Acknowledgment

This research was funded by Regular-Fundamental 2024 Research Grant, Ministry of Education, Culture, Research and Technology, Indonesia to Gunadarma University No. 105/E5/PG.02.00.PL/2024, Tanggal 11 Juni 2024.

5 REFERENCES

[1] Permenkes No. 24. (2022). Peraturan Menteri Kesehatan RI No 24 tahun 2022 tentang Rekam Medis. *Peraturan Menteri Kesehatan Republik Indonesia Nomor 24 Tahun 2022*, *151*(2), 1–19. (in Indonesian)

[2] Raihan, F. M. (2021). Perancangan Sistem Informasi Rekam Medis Pada Klinik Saffira Sentra Medika Batam. *Jurnal Sains, Nalar, Dan Aplikasi Teknologi Informasi*, *1*(1). (in Indonesian) https://doi.org/10.20885/snati.v1i1.7

[3] Suryadi, A., Arif, Y. W. T., & Novitasari, N. S. (2022). Rancang Bangun Sistem Informasi Rekam Medis Klinik Rawat Jalan Berbasis Web. *Infokes: Jurnal Ilmiah Rekam Medis Dan Informatika Kesehatan*, *12*(1), 37–43. (in Indonesian) https://doi.org/10.47701/infokes.v12i1.1498

[4] Umar, T. L. (2021). Perancangan Sistem Informasi Geografis Tempat Bersalin Berbasis Mobile. *Jurnal Informatika Dan Rekayasa Perangkat Lunak (JATIKA)*, *2*(2), 221–229. (in Indonesian) http://jim.teknokrat.ac.id/index.php/informatika.

[5] Tinambunan, M., & Sintaro, S. (2021). Aplikasi Restfull Pada Sistem Informasi Geografis Pariwisata Kota Bandar Lampung. *Jurnal Informatika Dan Rekayasa Perangkat Lunak*, *2*(3), 312–323. (in Indonesian) https://doi.org/10.33365/jatika.v2i3.1230.

[6] Wardhana, A. (2023). Grounded Theory. *As-Shaff*. http://www.sciencedirect.com/science/article/pii/S0020748909003629

[7] Kesa, I., & Sainuddin, I. (2020). Pengoperasian Penelitian Grounded Theory. *Sociology the Journal of the British Sociological Association*, *1*, 14–23. http://www.msvu.ca/site/media/msvu/MixedMethodologyHandout.pdf

[8] Asbui, Risnita, M. Syahran Jailani, M. Husnullail, & Asrul. (2024). Metode Grounded Theory Dalam Pendekatan Praktis. *Jurnal Cahaya Mandalika ISSN 2721-4796 (Online)*, *5*(1), 47–58. (in Indonesian) https://doi.org/10.36312/jcm.v5i1.2298

[9] Nugraha, F. S., Syahidin, Y., Suryani, A. I., Kesehatan, M. I., Ganesha, P. P., Gatot, J., & No, S. (2024). *Penerapan Teknologi Sistem Informasi dalam Proses Pembuatan Surat Keterangan Kematian Berbasis Elektronik Menggunakan Metode Agile*. *7*(2), 483–493. (in Indonesian) https://doi.org/10.32493/jtsi.v7i2.39134.

[10] Rezy, A. F., Yuga Utama, M., & Ramadhan, N. R. (2023). Pengembangan Aplikasi Klinik Berbasis Web Untuk Pengelolaan Rekam Medis Menggunakan Metode Agile. *Jurnal Ilmu Komputer, Teknik Dan Multimedia*, *VOL 1*, *NO.*(2), 309–319. (in Indonesian)

[11] Atim, S. B. (2024). Permodelan Sistem Informasi Penjualan Barang Berbasis Website Menggunakan Metode Agile. *Journal of Data Science and Information ...*, *2*(1), 14–25. (in Indonesian) https://ejournal.techcart-press.com/index.php/dimis/article/view/97%0Ahttps://ejournal.techcart-press.com/index.php/dimis/article/download/97/92

[12] Nurman Hidayat, & Kusuma Hati. (2021). Penerapan Metode Rapid Application Development (RAD) dalam Rancang Bangun Sistem Informasi Rapor Online (SIRALINE). *Jurnal Sistem Informasi*, *10*(1), 8–17. https://doi.org/10.51998/jsi.v10i1.352

[13] Arrohimi, M. J., Windyasari, V. S., & Kurniasari, R. (2023). Perancangan Sistem Rekam Medis Dengan Metode RAD Pada Klinik Mumtaz. *Jurnal Ilmiah Fakultas Teknik*, *3*(1), 27–41. (in Indonesian) https://doi.org/10.33592/jimtek.v3i1.3783

[14] Wahid, B. A. (2022). Penerapan Metode Rapid Application Development Terhadap Penjualan Fashion Distro Secara Online. *Jurnal Esensi Infokom : Jurnal Esensi Sistem Informasi Dan Sistem Komputer*, *3*(1), 33–39. https://doi.org/10.55886/infokom.v3i1.345.

[15] Noviantoro, A., Silviana, A. B., Fitriani, R. R., & Permatasari, H. P. (2022). Rancangan Dan Implementasi Aplikasi Sewa Lapangan Badminton Wilayah Depok Berbasis Web. *Jurnal Teknik Dan Science*, *1*(2), 88–103. (in Indonesian) https://doi.org/10.56127/jts.v1i2.108.

[16] Tabrani, M., & Rezqy Aghniya, I. (2020). Implementasi Metode Waterfall Pada Program Simpan Pinjam Koperasi Subur Jaya Mandiri Subang. *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi Dan Komunikasi*, *14*(1), 44–53. (in Indonesian) https://doi.org/10.35969/interkom.v14i1.65

[17] Kurniawan, H., Apriliah, W., Kurnia, I., & Firmansyah, D. (2021). Penerapan Metode Waterfall Dalam Perancangan Sistem Informasi Penggajian Pada Smk Bina Karya Karawang. *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi Dan Komunikasi*, *14*(4), 13–23. (in Indonesian) https://doi.org/10.35969/interkom.v14i4.78

[18] Wulandari, T., & Nurmiati, S. (2022). Rancang Bangun Sistem Pemesanan Wedding Organizer Menggunakan Metode Rad di Shofia Ahmad Wedding. *Jurnal Rekasaya Informasi*, *11*(69), 79–85. (in Indonesian)

[19] Julianti, M. R., Dzulhaq, M. I., & Subroto, A. (2019). Sistem Informasi Pendataan Alat Tulis Kantor Berbasis Web pada PT Astari Niagara Internasional. *Jurnal Sisfotek Global*, *9*(2). (in Indonesian) https://doi.org/10.38101/sisfotek.v9i2.254

[20] Abdillah, M. T., Kurniastuti, I., Susanto, F. A., & Yudianto, F. (2023). Implementasi Black Box Testing dan Usability Testing pada Website Sekolah MI Miftahul Ulum Warugunung Surabaya. *Journal of Computer Science and Visual Communication Design*, *8*(1), 234–242. (in Indonesian) https://doi.org/10.55732/jikdiskomvis.v8i1.897

[21] Uminingsih, Ichsanudin, M. N., Yusuf, M., & Suraya. (2022). Perpustakaan Dengan Metode Black Box Testing Bagi Pemula. *STORAGE-Jurnal Ilmiah Teknik Dan Ilmu Komputer*, *1*(2), 1–8. (in Indonesian)

Authors' contacts:

Lely Prananingrum, Dr
Gunadarma University,
Margonda 100, Depok, West Java, Indonesia
(021) 29428935, lely_p@staff.gunadarma.ac.id

Teuku Salman Farizi
Gunadarma University,
Margonda 100, Depok, West Java, Indonesia
(021) 294228935, teukusalmanfarizi2003@gmail.com

Fajar Agus Dwi Rahmawan
Gunadarma University,
Margonda 100, Depok, West Java, Indonesia
(021) 294228935, fajarrahmawan0804@gmail.com

Ilmiyati Sari, Dr
(Corresponding author)
Gunadarma University,
Margonda 100, Depok, West Java, Indonesia
(021) 29428935, ilmiyati@staff.gunadarma.ac.id